Desarrollo de técnicas de aumento de datos para la aplicación de aprendizaje profundo en problemas de bioinformática

UNIVERSIDAD DE MÁLAGA

TESIS DOCTORAL

Francisco Javier Moreno Barea

Programa de Doctorado en Tecnologías Informáticas
Departamento de Lenguajes y Ciencias de la Computación
Escuela Técnica Superior de Ingeniería Informática
Universidad de Málaga

Septiembre 2022

UNIVERSIDAD DE MÁLAGA

AUTOR: Francisco Javier Moreno Barea

https://orcid.org/0000-0002-3887-9095

EDITA: Publicaciones y Divulgación Científica. Universidad de Málaga

Esta obra está bajo una licencia de Creative Commons Reconocimiento-NoComercial-SinObraDerivada 4.0 Internacional:
http://creativecommons.org/licenses/by-nc-nd/4.0/legalcode
Cualquier parte de esta obra se puede reproducir sin autorización
pero con el reconocimiento y atribución de los autores.
No se puede hacer uso comercial de la obra y no se puede alterar, transformar o hacer obras derivadas.

Esta Tesis Doctoral está depositada en el Repositorio Institucional de la Universidad de Málaga (RIUMA): riuma.uma.es

Desarrollo de técnicas de aumento de datos para la aplicación de aprendizaje profundo en problemas de bioinformática

Memoria para optar al título de Doctor por la Universidad de Málaga

Francisco Javier Moreno Barea

Dirigida por los doctores

Leonardo Franco
José Manuel Jerez Aragonés

Programa de Doctorado en Tecnologías Informáticas
Departamento de Lenguajes y Ciencias de la Computación
Escuela Técnica Superior de Ingeniería Informática
Universidad de Málaga

Septiembre 2022

UNIVERSIDAD DE MÁLAGA

UNIVERSIDAD
DE MÁLAGA

DECLARACIÓN DE AUTORÍA Y ORIGINALIDAD DE LA TESIS PRESENTADA PARA OBTENER EL TÍTULO DE DOCTOR

D./Dña FRANCISCO JAVIER MORENO BAREA
Estudiante del programa de doctorado TECNOLOGÍAS INFORMÁTICAS de la Universidad de Málaga, autor/a de la tesis, presentada para la obtención del título de doctor por la Universidad de Málaga, titulada: DESARROLLO DE TÉCNICAS DE AUMENTO DE DATOS PARA LA APLICACIÓN DE APRENDIZAJE PROFUNDO EN PROBLEMAS DE BIOINFORMÁTICA

Realizada bajo la tutorización de LEONARDO FRANCO y dirección de JOSÉ MANUEL JEREZ ARAGONÉS, Y LEONARDO FRANCO (si tuviera varios directores deberá hacer constar el nombre de todos)

DECLARO QUE:

La tesis presentada es una obra original que no infringe los derechos de propiedad intelectual ni los derechos de propiedad industrial u otros, conforme al ordenamiento jurídico vigente (Real Decreto Legislativo 1/1996, de 12 de abril, por el que se aprueba el texto refundido de la Ley de Propiedad Intelectual, regularizando, aclarando y armonizando las disposiciones legales vigentes sobre la materia), modificado por la Ley 2/2019, de 1 de marzo.

Igualmente asumo, ante a la Universidad de Málaga y ante cualquier otra instancia, la responsabilidad que pudiera derivarse en caso de plagio de contenidos en la tesis presentada, conforme al ordenamiento jurídico vigente.

En Málaga, a 28 de SEPTIEMBRE de 2022

Fdo.: FRANCISCO JAVIER MORENO BAREA Doctorando/a	Fdo.: LEONARDO FRANCO Tutor/a
Fdo.: LEONARDO FRANCO Director/es de tesis	JOSÉ MANUEL JEREZ ARAGONÉS

Edificio Pabellón de Gobierno. Campus El Ejido.
29071
Tel.: 952 13 10 28 / 952 13 14 61 / 952 13 71 10
E-mail: doctorado@uma.es

UNIVERSIDAD
DE MÁLAGA

UNIVERSIDAD DE MÁLAGA

Departamento de Lenguajes y Ciencias de la Computación
Escuela Técnica Superior de Ingeniería Informática
Universidad de Málaga

El Dr. D. Leonardo Franco, Catedrático de Universidad perteneciente al área de Ciencia de la Computación e Inteligencia Artificial de la E.T.S. de Ingeniería Informática de la Universidad de Málaga, y el Dr. D. José Manuel Jerez Aragonés, Catedrático de Universidad perteneciente al área de Lenguajes y Sistemas Informáticos de la E.T.S. de Ingeniería Informática de la Universidad de Málaga,

Certifican que,

D. Francisco Javier Moreno Barea, Graduado en Ingeniería Informática con Mención en Computación, ha realizado en el Departamento de Lenguajes y Ciencias de la Computación de la Universidad de Málaga, bajo su dirección, el trabajo de investigación correspondiente a su Tesis Doctoral titulada:

Desarrollo de técnicas de aumento de datos para la aplicación de aprendizaje profundo en problemas de bioinformática

Revisado el presente trabajo, estimamos que puede ser presentado al tribunal que ha de juzgarlo. Y para que conste a efectos de lo establecido en la legislación vigente, autorizamos la presentación de este trabajo en la Universidad de Málaga.

Málaga, Septiembre de 2022

Fdo.: Dr. Leonardo Franco Fdo.: Dr. José Manuel Jerez Aragonés

UNIVERSIDAD DE MÁLAGA

UNIVERSIDAD
DE MÁLAGA

A mis padres
por su amor, paciencia,
comprensión y apoyo infinito.
Gracias y mil gracias.

UNIVERSIDAD
DE MÁLAGA

UNIVERSIDAD
DE MÁLAGA

Agradecimientos

Con esta tesis doctoral concluyen mis últimos años de vida académica en mi camino para convertirme en Doctor. En este proceso de investigación he recorrido ese sendero a veces tortuoso y otras amable, y que requiere esfuerzo, constancia y dedicación. Al final hay una meta de satisfacción, aunque lo realmente importante son las experiencias vividas por el camino, que muchas veces tiene profundos hoyos de los que es difícil escapar si no tienes el apoyo de las personas que te rodean.

Por ello, me gustaría mostrar en primer lugar mi agradecimiento a mis directores de tesis, D. José Manuel Jerez Aragonés y D. Leonardo Franco. Varias veces nos hemos visto en un hoyo en el camino, mucho tiempo dedicado a una cuestión que al final no ha arrojado resultados suficientes para ser meritorios. Pero ellos han sabido ayudarme a salir de él y emprender el camino hacia una nueva dirección. Les agradezco su amabilidad, dedicación, confianza y paciencia que me han mostrado y otorgado para completar este trabajo. Ha sido, y espero que siga siendo, un privilegio trabajar con vosotros. El mismo pensamiento se lo dedico a mis compañeros del grupo ICB y de investigación, con especial agradecimiento a D. Julio Montes, D. Héctor Mesa y D. Karl Thurnhofer, a los que he recurrido tantas veces, y siempre he obtenido de ellos amabilidad y apoyo. También quiero dar las gracias a D. David Elizondo y D. Martin Grootveld por dejarme colaborar con ellos en una situación tan compleja como la vivida en estos últimos años, y por su confianza e interés mostrado en mi investigación. En el ámbito más personal, no puedo olvidarme del apoyo y cariño recibido de toda mi familia, incluyendo los que no están, pues seguro que se alegrarían mucho por mí. Y también de mis amigos, unos perjudicados por los sacrificios que a veces ha conllevado este proceso.

Por último, mi más profundo agradecimiento quiero hacérselo a mis padres. Han sido los que más han sufrido y los que más se han alegrado, incluso más que yo. Su apoyo absoluto y amor incondicional ha hecho que esto fuera posible, que llegara a este final sin dejarlo todo por el camino. Y en este momento, yo soy lo que soy gracias a que ellos siempre han estado a mi lado. Estoy muy orgulloso de vosotros y espero que también lo estéis de mí.

UNIVERSIDAD DE MÁLAGA

Apoyo Financiero

Este trabajo de tesis doctoral cuenta con el apoyo del Ministerio de Economía y Competitividad de España con las subvenciones TIN2017-88728-C2-1-R, PPIT.UMA.B1.2017 y PID2020-116898RB-I00. También cuenta con el apoyo del Ministerio de Ciencia, Innovación y Universidades de España con la subvención RTI2018-094645-B-I00, y de la Junta de Andalucía (España) con las subvenciones TIC-6213, UMA18-FEDERJA-084 y UMA20-FEDERJA-045. Todos ellos incluyen fondos del Fondo Europeo de Desarrollo Regional (FEDER). Agradezco el apoyo del Instituto de Investigaciones Biomédicas de Málaga (IBIMA) y la Universidad de Málaga. Finalmente, el agradecimiento por el apoyo de NVIDIA Corporation con la donación de una GPU Titan X utilizada para esta investigación. Francisco Javier Moreno Barea fue apoyado por un contrato predoctoral como parte del Plan Propio de Investigación y Transferencia de la Universidad de Málaga, y también por una beca predoctoral (FPU18/02574) del Ministerio de Educación, Cultura y Deporte de España en el marco del programa de Formación de Profesorado Universitario (FPU).

UNIVERSIDAD DE MÁLAGA

UNIVERSIDAD
DE MÁLAGA

UNIVERSIDAD
DE MÁLAGA

Resumen

Esta tesis doctoral trata sobre el desarrollo de métodos de aumento de datos para mejorar la aplicación del aprendizaje profundo en problemas de bioinformática. En la última década, el aprendizaje profundo (DL) se ha impuesto como el enfoque de inteligencia artificial con mayor progresión y éxito. Las técnicas de DL conforman actualmente el estado del arte en visión por computador y procesamiento del lenguaje natural, mostrando además un potencial prometedor en un campo de gran impacto económico y social como es la bioinformática.

Sin embargo, estos modelos presentan una importante desventaja, dado que requieren el análisis de conjuntos de datos compuestos de miles de instancias para lograr un buen nivel de éxito. Actualmente en bioinformática, la adquisición de datos sigue siendo un proceso difícil y costoso, especialmente cuando se trata de conjuntos de exploración de microarrays de expresión génica, expresión molecular o metabolómica. Estos datos son significativamente difíciles de obtener, su tratamiento conlleva una mayor dificultad, y su escasez es acuciante en estudios de enfermedades raras o regiones geográficas concretas.

Para resolver esta limitación se puede emplear el aumento de datos (DA), es decir, el incremento del número de muestras disponibles aplicando transformaciones sobre el conjunto o generando muestras totalmente nuevas. En los últimos años, modelos de DA enmarcados dentro del DL han obtenido un rendimiento asombroso en generación de imágenes sintéticas y ya se utilizan en series secuenciales y procesamiento lenguaje natural. Sin embargo, aplicar estos modelos a conjuntos bioinformáticos que no presentan información espacial o temporal es una tarea desafiante.

A este respecto, la investigación realizada en esta tesis describe en primer lugar la motivación, los objetivos y la metodología empleada. Nuestro objetivo es el desarrollo de métodos de DA y su aplicación en problemas con conjuntos biomédicos no estructurados. Posteriormente se presenta el estado del arte del DA, el DL y la aplicación de modelos generativos profundos en problemas de bioinformática. Entre los modelos generativos profundos, se presta especial atención a las *Generative Adversarial Networks* (GAN), que han mostrado un gran rendimiento en la generación de imágenes falsas.

UNIVERSIDAD DE MÁLAGA

A continuación, se describen los métodos de DA desarrollados y se presentan los resultados de su aplicación en problemas habituales de aprendizaje automático con un número de instancias reducido. Estos métodos se basan en adición de ruido gaussiano y un esquema específico de ajuste de niveles de ruido, obteniendo con este último una ligera mejora en la métrica F_1, pero requiriendo un tiempo de computación elevado. También se presenta el modelo adaptable al uso de modelos generativos profundos, con funcionalidades añadidas para controlar el balanceo de los conjuntos generados. Este modelo constituye un enfoque útil para utilizar cualquier modelo generativo en una tarea supervisada y aumentar la capacidad de predicción.

Se aplican los algoritmos, modelos y métodos de DA desarrollados sobre tres problemas del campo de la bioinformática: predicción de eventos en cáncer mediante el análisis de muestras RNA-Seq; predicción de la enfermedad rara Niemann-Pick Tipo C a partir de datos metabolómicos; y clasificación del subtipo molecular de cáncer de mama utilizando la expresión génica del sistema inmune. El rendimiento se analiza mediante métricas comúnmente usadas en la literatura en tareas de predicción como la capacidad predictiva, especificidad, sensibilidad, *F-measure* y AUC-ROC; y mediante métodos y métricas de análisis de las características de los datos generados.

De las investigaciones realizadas con datos ómicos se concluye el potencial del DA para generar muestras que replican información biomédica y conducen a un aumento en el rendimiento de predicción, con independencia de la técnica utilizada como método de clasificación. También se presenta un esquema meta-clasificador diseñado para poder aprovechar los efectos de la aplicación simple de DA, habiéndose observado un aumento claro en la sensibilidad obtenida acompañada de una reducción en la precisión.

Por último, se presentan y discuten las conclusiones generales de la investigación y se describen posibles líneas de trabajo futuro que puedan hacer uso de los resultados obtenidos en esta tesis doctoral. En el apéndice B hay un resumen más extenso de la tesis doctoral en inglés.

Palabras Clave:

Aprendizaje Profundo • Aumento de Datos • Bioinformática • GAN • SMOTE • Adición de Ruido • Datos Reducidos • Conjuntos Desbalanceados • Aprendizaje Automático • Inteligencia Artificial

UNIVERSIDAD DE MÁLAGA

Abstract

This Ph.D. thesis is about development of data augmentation methods to improve the application of deep learning in bioinformatics problems. In the last decade, deep learning (DL) has established as the most successful artificial intelligence approach. These techniques are currently the state of the art in computer vision and natural language processing and, in a field with great social impact such as bioinformatics, DL has also shown promising potential.

However, these models present an important disadvantage, they require the analysis of datasets composed of thousands of instances to achieve a good level of success. Currently in bioinformatics, data acquisition remains a difficult and expensive process, especially when dealing with gene expression microarray, molecular expression or metabolomics datasets. These are significantly difficult to collect, their treatment entails greater difficulty, and their scarcity is frequent in rare diseases studies or research in specific geographical regions.

The data augmentation (DA) strategy can be employed to avoid this limitation. This strategy involves increasing the number of available samples by applying transformations to the dataset or generating new synthetic samples. In recent years, DA models from the DL framework have achieved amazing performance in synthetic image generation, and are already used in sequential series and natural language processing. Nevertheless, applying these DL models to bioinformatic datasets without spatial or temporal information is a challenging task.

In this regard, the research carried out in this Ph.D. thesis first describes the motivation, the objectives and the methodology employed. The development of DA methods and their application in problems with 'non-structured' biomedical datasets will be our goal. After that, the state of the art of DA, DL and application studies of deep generative models in bioinformatics problems are presented. The Generative Adversarial Networks (GAN) draw special attention among the deep generative models, since they show a great performance in the generation of fake images.

Thereafter, the development of DA methods is described and the application results in common problem datasets of machine learning with a

UNIVERSIDAD DE MÁLAGA

reduced number of instances are presented. These methods are based on the addition of Gaussian noise and a specific noise level adjustment scheme, obtaining with the latter a slight improvement in F_1 score, but requiring a heavy computation time. The adaptive model for the use of deep generative models is also presented, with added functionality to control the balance of the augmented sets. This model is a useful approach to use any generative model in a supervised task and increase prediction power.

The application of developed DA methods, models and algorithms on three problems in the bioinformatics field is addressed: cancer events prediction through RNA-Seq samples analysis; prediction of rare Niemann-Pick Type C disease from metabolomics data; and breast cancer molecular subtype classification using the gene expression in the immune system. Performance is analysed by metrics commonly used in prediction tasks such as accuracy, specificity, sensitivity, *F-measure* and AUC-ROC; and by methods and metrics for analysing the features of the data generated.

From this research with omics data, the potential of DA to generate samples that replicate biomedical information is concluded. This leads to an increase in prediction performance, regardless of the technique used as classifier. A meta-classifier scheme designed to take advantage of the effects of the simple DA application is also presented, as a clear increase in sensitivity was observed along with a reduction in accuracy.

Finally, the general conclusions of the research are presented and discussed, and possible future work lines that can make use of the results obtained in this Ph.D thesis are described. In the appendix B there is a more extensive summary of the Ph.D thesis in English.

Keywords:

Deep Learning • Data Augmentation • Bioinformatics • GAN • SMOTE • Noise Injection • Small Data • Unbalanced Sets • Machine Learning • Artificial Intelligence

UNIVERSIDAD DE MÁLAGA

UNIVERSIDAD
DE MÁLAGA

UNIVERSIDAD
DE MÁLAGA

Índice

Agradecimientos xi

Resumen xv

Abstract xvii

I Contexto y Fundamentos 1

1. Introducción 3
1.1. Motivación 4
1.2. Objetivos 11
1.3. Metodología 12
1.4. Contribuciones de la Tesis 14
1.5. Estructura de la Tesis 14

2. Revisión del Estado del Arte 17
2.1. Aumento de Datos 17
2.1.1. Métodos basados en transformaciones 18
2.1.2. Métodos de remuestreo 23
2.1.3. Métodos basados en SMOTE 25
2.2. Aprendizaje profundo 30
2.2.1. Redes neuronales feedforward profundas 31
2.2.2. Redes neuronales convolucionales 33
2.2.3. Redes neuronales recurrentes 35
2.2.4. Función de activación 38
2.2.5. Regularización 41
2.2.6. Optimización del gradiente 45
2.3. Modelos generativos profundos 48
2.3.1. Máquinas de Boltzmann 49
2.3.2. Autoencoders 50
2.3.3. Variational Autoencoders 52

UNIVERSIDAD DE MÁLAGA

2.3.4. Generative Adversarial Networks 56
2.3.5. Aplicaciones de GAN en DA y biomedicina 62

3. Técnicas de Aprendizaje y Evaluación 69
3.1. Métodos de selección de características 69
3.1.1. Modelo LASSO 70
3.1.2. Random Forest 71
3.2. Medidas de evaluación y validación 73
3.2.1. Métricas de clasificación 73
3.2.2. Área bajo la curva ROC 76
3.2.3. Análisis de componentes principales 77
3.2.4. Validación Cruzada 78
3.3. Modelos clasificadores auxiliares 80
3.3.1. Regresión logística 80
3.3.2. Máquina de vectores de soporte 82

II Métodos de Aumento de Datos 85

4. Métodos de Adición de Ruido 87
4.1. Método de adición de ruido variable FNA 87
4.2. Métodos de adición de ruido gaussiano 91
4.3. Resultados de aplicación con el algoritmo FNA 93
4.3.1. Arquitectura del modelo de clasificación 93
4.3.2. Resultados experimentales 94
4.4. Conclusiones 100

5. Adaptación de Modelos Generativos Profundos 103
5.1. Modelo clasificador generativo 104
5.2. Modificaciones para tratar distribuciones desbalanceadas . . . 108
5.3. Modelos generativos profundos estudiados 110
5.3.1. Implementaciones de los modelos 110
5.4. Resultados experimentales 114
5.5. Conclusiones 125

III Aplicación en Bioinformática 129

6. Predicción de eventos en cáncer 131
6.1. Introducción al Problema 131
6.2. Implementación de los modelos 136
6.3. Resultados Experimentales 139

UNIVERSIDAD DE MÁLAGA

6.3.1. Rendimiento en predicción 140
6.3.2. Análisis del conjunto generado 151
6.4. Conclusiones 159

7. Predicción Niemann-Pick C1 163
7.1. Introducción al Problema 163
7.2. Implementación de los modelos 166
7.3. Resultados Experimentales 168
7.3.1. Rendimiento en predicción 170
7.3.2. Análisis del conjunto generado 178
7.3.3. Replicación de importancia de los metabolitos 185
7.4. Conclusiones 190

8. Predicción del Subtipo Molecular de Cáncer 193
8.1. Introducción al Problema 193
8.2. Implementación de los modelos 196
8.3. Resultados Experimentales 200
8.3.1. Rendimiento en predicción 202
8.3.2. Análisis del conjunto generado 211
8.4. Conclusiones 216

IV Método Meta Clasificador 219

9. Meta-Clasificador para Conjuntos Desbalanceados 221
9.1. Enfoques para el desequilibrio de clases 222
9.2. Trabajos relacionados 223
9.3. Modelo Meta-Clasificador 224
9.4. Implementación de los modelos 227
9.5. Resultados Experimentales 230
9.6. Conclusiones 240

V Conclusiones 243

10. Conclusiones y Trabajos Futuros 245
10.1. Discusiones 245
10.2. Conclusiones 248
10.3. Trabajo Futuro 250

UNIVERSIDAD DE MÁLAGA

VI Apéndices **253**

A. Problemas estudiados **255**
A.1. Capítulo 4: FNA 255
A.2. Capítulo 5: Modelo clasificador generativo 257
A.3. Capítulo 6: Predicción de eventos en cáncer 262
A.4. Capítulo 7: Niemman-Pick C1 264
A.5. Capítulo 8: Subtipo molecular del cáncer 266
A.6. Capítulo 9: Esquema Meta Clasificador 271

B. Thesis Summary in English **275**
B.1. Introduction 275
B.1.1. Motivation and Objective 276
B.1.2. Structure of the Summary 279
B.2. State of the Art 279
B.2.1. Transformation-based Methods and SMOTE 279
B.2.2. Deep Generative Models 281
B.3. Designed Data Augmentation Methods 285
B.3.1. Noise Addition Methods 285
B.3.2. Adaptation of Deep Generative Models 288
B.4. DA Application in Bioinformatics 294
B.4.1. Events Prediction in Cancer 294
B.4.2. Niemann–Pick C1 Prediction 300
B.4.3. Cancer Molecular Subtype Prediction 306
B.5. Meta-Classifier Method 311
B.6. Conclusions 315
B.7. Future Works 317

C. Publicaciones que avalan la tesis **319**
C.1. Revistas indexadas JCR 319
C.2. Conferencias Internacionales 320

Bibliografía **323**

UNIVERSIDAD DE MÁLAGA

Índice de figuras

1.1. Número de artículos científicos publicados relacionados con aprendizaje profundo desde 2010. 6
1.2. Ejemplo del proceso de aumento de datos aplicado a una imagen del conjunto de datos CIFAR-10. Izquierda: imagen de muestra original. Derecha: imagen creada a través de una rotación de 15°. 9

2.1. Ejemplos de transformaciones geométricas y del espacio de color[1]. 20
2.2. Ejemplos de filtros de kernel[2] (izquierda) y Random erasing[3] (derecha). 21
2.3. Ejemplo de combinación de instancias para procesamiento del lenguaje natural[4]. 22
2.4. Ejemplo de aplicación sobre una oración del método de traducción inversa[5]. 23
2.5. Generación de muestras sintéticas con la técnica de sobremuestreo SMOTE. 26
2.6. Generación con Borderline-SMOTE[6]: a) sobre el conjunto sintético Circle; b) instancias "peligrosas"; c) instancias generadas. 29
2.7. Redes Neuronales *Feedforward* profunda y "no-profunda"[7]. . . 31
2.8. Red Neuronal Convolucional[8]. 35
2.9. Una red neuronal recurrente y el desarrollo en el tiempo del proceso de *unfolding*[9]. 36
2.10. Representación de las funciones de activación sigmoid, tanh, softplus y softsign (izquierda) y las funciones ReLU, Leaky ReLU, PReLu y ELU (derecha). 40
2.11. Ejemplo de aplicación del método dropout en una red neuronal. 43
2.12. Gráfica ejemplificando la minimización del gradiente[10]. 46
2.13. Ejemplo de *Denoising Autoencoder* 52
2.14. Diagrama esquemático del funcionamiento del modelo VAE. . 53
2.15. Diagrama esquemático del funcionamiento del modelo *Denoising* VAE. 54

UNIVERSIDAD DE MÁLAGA

2.16. Diagrama esquemático del funcionamiento del modelo *Conditional* VAE. 55

2.17. Aplicación del modelo *Grammar* VAE en la replicación de estructuras moleculares[11]. 56

2.18. Diagrama esquemático del funcionamiento del modelo GAN. . 57

2.19. Diagrama esquemático del funcionamiento del modelo Wasserstein GAN con penalización del gradiente. 60

2.20. Diagrama esquemático del funcionamiento del modelo *Conditional* GAN. 61

2.21. Aplicación del modelo CycleGAN en la traducción de imágenes entre varios dominios[12]. 62

2.22. Generación de tomografías computarizadas cerebrales sin contraste a partir de imágenes con contraste[13]. 64

2.23. Visualización t-SNE de las células reales (a). Expresión del gen NKG7 en células reales (b) y generadas por scGAN (c)[14]. 65

2.24. Visualización t-SNE de las muestras generadas mediante diverson modelos generativos en el estudio del diagnóstico de cáncer[15]. 67

3.1. Representación gráfica del método Random Forest[16]. 72

3.2. Validación cruzada con 10-folds. 79

4.1. Esquema del procedimiento FNA propuesto para la estimación del nivel óptimo de ruido. 90

4.2. Esquema de la arquitectura del perceptrón multicapa utilizado en la experimentación con el procedimiento FNA. Presenta 3 capas ocultas con 32, 64 y 32 neuronas respectivamente, utilizando Leaky ReLU (LR) como función de activación. Las cajas azules representan el uso de dropout con 0.4 como valor de tasa de abandono. 94

4.3. Relación entre la medida F_1 para los conjuntos de datos de entrenamiento aumentados (500 % DA) y los valores de referencia para los conjuntos de test (la línea continua es la función de identidad). 98

4.4. Medida relativa de F_1 en función del número de características (gráfico superior), número de instancias (gráfico medio) y grado de desbalanceo (gráfico inferior) obtenido para cada conjunto de datos de referencia. 99

UNIVERSIDAD DE MÁLAGA

4.5. Rendimiento en entrenamiento del modelo DL respecto al número de épocas para los conjuntos '*card*'y '*wdbc*'. Las líneas muestran la pérdida obtenida cuando no se aplica ningún aumento de datos, y con el proceso de aumento de datos teniendo en cuenta el número de patrones generado según un porcentaje de 100 %, 200 % y 500 %. 100

5.1. Ejemplo de generación de muestras sintéticas con un modelo GAN cuando se aplica con un conjunto de imágenes (MNIST) y con un conjunto no estructurado. Existe un impedimento a la hora de realizar una evaluación visual de la corrección de las muestras generadas. 104

5.2. Proceso del Modelo clasificador generativo para la creación de datos sintéticos. El modelo generativo G tiene como entrada el conjunto train. Cuando se entrena, se generan las muestras sintéticas z y el clasificador generativo C predice las etiquetas apropiadas y. Si la etiqueta label_i representa ruido, sample_i se descarta y G genera otra nueva muestra. 106

5.3. El proceso de entrenamiento del clasificador generativo (C) implica el uso de entrenamiento (Tr_i) y conjuntos de validaciones (V_i) de los datos de entrada, junto con el uso de fuentes de ruido. El objetivo del entrenamiento es que C discrimine entre las muestras originales y las ruidosas. 107

5.4. Esquema de las modificaciones introducidas para abordar el problema del aumento de datos. 109

5.5. Esquema de la arquitectura del perceptrón multicapa con 3 capas ocultas con N, M y K neuronas respectivamente, utilizando Leaky ReLU (LR) como función de activación. Las cajas azules representan el uso de normalización por lotes y dropout. 113

5.6. Diagrama de flujo del proceso desarrollado para realizar el DA y evaluar los resultados en un conjunto de problemas de referencia. 114

5.7. Proceso de clasificación para la realización de los experimentos. Los datos reales x se utilizan para realizar un proceso de validación cruzada de 10-folds. Los conjuntos de entrenamiento se incrementan agregando muestras sintéticas, implementando el aumento de datos. Se realiza la predicción con el conjunto Test y se mide la capacidad de predicción. 115

5.8. Diferencia relativa (RD) de capacidad de predicción frente al logaritmo del número de instancias utilizando los valores de la Tabla 5.2. La línea continua es una regresión lineal ajustada a los datos. 118

UNIVERSIDAD DE MÁLAGA

5.9. Tiempo de cálculo utilizado en los diferentes experimentos. La línea discontinua representa el proceso de generación usando el modelo WGAN-GP, y la línea continua el modelo WGAN-GP con el esquema 'Balanced Multiclass' con 100 % DA. El tiempo empleado en el proceso de clasificación se representa por una línea de guiones y puntos. 124

5.10. Capacidad predictiva utilizando los datos originales, la mejor opción para DA, el modelo WGAN-GP con 'Balanced Multiclass' 100 % DA, el modelo VAE con 'Multiclass' 100 % DA y el modelo CGAN con 100 % DA. Los resultados se indican para cuatro grupos: todos los conjuntos; agrupados según el número de instancias; y del ámbito médico. 126

6.1. Relación entre la capacidad predictiva obtenida con los conjuntos procesados y la obtenida con el conjunto de referencia original. Las cruces representan la selección de características y las aspas la incorporación del DA. La línea continua es la función de identidad. 142

6.2. Diferencia relativa (RD) en capacidad de predicción frente al logaritmo del número de instancias. Las líneas son una regresión lineal ajustada a los datos. Las aspas y la línea continua representan los resultados obtenidos con CGAN, los puntos y la línea discontinua los obtenidos con NOISE, y las cruces y la línea punteada los obtenidos con SMOTE. 144

6.3. Comparación de la distribución de capacidad predictiva en test obtenida con diferentes modelos de DA agrupados según el número de instancias en el conjunto de datos original. . . . 144

6.4. Diferencia relativa (RD) en capacidad predictiva frente al balanceo de los datos. Las líneas son una regresión lineal ajustada a los datos. Las aspas y la línea continua representan los resultados obtenidos con CGAN, los puntos y la línea discontinua los obtenidos con NOISE, y las cruces y la línea punteada los obtenidos con SMOTE. 145

6.5. Comparación de la distribución de capacidad predictiva en test obtenida con diferentes modelos de DA agrupados según el balanceo de clases presente en el conjunto original. 146

6.6. Curvas ROC obtenidas utilizando el conjunto de datos original y aplicando aumento de datos. La línea punteada representa el azar; la línea discontinua representa el enfoque con los datos originales; y la línea continua representa el aumento de datos. 150

UNIVERSIDAD DE MÁLAGA

6.7. Gráfico de PCA con las muestras originales y las muestras generadas por el método de adición de ruido para los conjuntos *ov*, *skcm*, *luad* y *thca*. Códigos de color: círculo verde, "vivo"; triángulo rojo, "muerto"; cuadrado morado, "vivo" sintético; triángulo invertido azul, "muerto" sintético. 152

6.8. Gráfico de PCA con las muestras originales y las muestras generadas con SMOTE para los conjuntos de datos *ov*, *skcm*, *luad* y *thca*. Códigos de color: igual que la Figura 6.7. 153

6.9. Gráfico de PCA con las muestras originales y las muestras generadas con los modelos basados en CGAN para los conjuntos de datos *ov*, *skcm*, *luad* y *thca*. Códigos de color: igual que la Figura 6.7. 154

6.10. Fréchet Inception Distance ($\widehat{FID}$) *versus* la diferencia relativa en test. Las aspas (x) representan el modelo CGAN, los puntos representan el método NOISE y las cruces (+) representan SMOTE. 156

7.1. Diagrama de flujo de todo el proceso de experimentación. División de los datos en conjuntos de entrenamiento y test en un proceso de validación cruzada; aplicación del modelo generativo con el conjunto de entrenamiento y aumento con las muestras sintéticas; transformación de variables de metabolitos a componentes principales; entrenamiento del clasificador y predicción con el conjunto de test. El proceso se repite con 5 semillas. 169

7.2. Proporción de varianza explicada acumulada con las diferentes componentes principales (PC) seleccionadas. Este proceso se realiza sobre el conjunto de datos original. El número de PC seleccionadas es de 14. 170

7.3. Comparación de la capacidad predictiva, especificidad y sensibilidad obtenidas usando regresión logística con diferentes modelos de DA, con respecto al logaritmo del número de instancias creadas. Los puntos representan los resultados con el modelo CGAN; las cruces con el método NOISE; y las aspas con el método NOISE Bal. 172

7.4. Comparación de las métricas de test obtenidas con diferentes modelos de DA y una regresión logística como sistema clasificador. 173

7.5. Comparación de la capacidad predictiva, especificidad y sensibilidad obtenidas usando un Random Forest con diferentes modelos de DA, con respecto al logaritmo del número de instancias creadas. Código de símbolos: igual que la Figura 7.3. . 174

UNIVERSIDAD DE MÁLAGA

7.6. Comparación de las métricas de test obtenidas con diferentes modelos de DA y una estrategia Random Forest como sistema clasificador. 175
7.7. Comparación de la capacidad predictiva, especificidad y sensibilidad obtenidas usando un sistema de SVM con diferentes modelos de DA, con respecto al logaritmo del número de instancias creadas. Código de símbolos: igual que la Figura 7.3. . 176
7.8. Comparación de las métricas de test obtenidas con diferentes modelos de DA y un sistema de SVM como clasificador. . . . 177
7.9. Gráfico de PCA con el conjunto original de la enfermedad NPC1. Código de colores: triángulos verdes, orina con enfermedad NPC1; círculos azul oscuro, orina de control de portadores heterocigotos. 179
7.10. Gráfico de PCA con el conjunto original y las muestras generadas con SMOTE. Código de colores: igual que la Figura 7.9, con muestras de orina de la enfermedad NPC1 generadas por SMOTE representadas por triángulos rojos invertidos. 179
7.11. Gráfico de PCA con el conjunto original y las muestras generadas vía adición de ruido. El porcentaje de muestras generadas es 100 % (izquierda) y 2000 % (derecha). Código de colores: igual que la Figura 7.9, con cuadrados de color azul claro, muestras de orina generadas de control de portadores heterocigotos; triángulos rojos invertidos, muestras de orina generadas de la enfermedad NPC1. 180
7.12. Gráfico de PCA con el conjunto original y las muestras generadas con la adición de ruido en un proceso de balanceo. El porcentaje de muestras generadas es 500 % (izquierda) y 2000 % (derecha). Código de colores: igual que la Figura 7.9, con triángulos rojos invertidos representando las muestras de orina NPC1 generadas. 181
7.13. Gráfico de PCA con el conjunto original y las muestras generadas con el modelo CGAN. El porcentaje de muestras generadas es 100 % (izquierda) y 2000 % (derecha). Código de colores: igual que la Figura 7.11. 182
7.14. Gráfico de PLS-DA componente 2 *versus* componente 1 (izquierda) y resultados de validación cruzada para el conjunto de datos original (derecha). Código de colores (izq): triángulos verdes, orina con enfermedad NPC1; círculos rojos, orina de control de portadores heterocigotos. 183
7.15. Gráfico de puntuaciones PLS-DA y resultados de validación cruzada adquiridos con los datos originales y las muestras generadas con la aplicación de SMOTE. Código de colores (izq): igual que la Figura 7.14. 183

UNIVERSIDAD DE MÁLAGA

7.16. Gráfico de puntuaciones PLS-DA y resultados de validación cruzada obtenidos con los datos originales y las muestras generadas con el modelo CGAN. Código de colores (izq): igual que la Figura 7.14. 184
7.17. Gráfico de puntuaciones PLS-DA y resultados de validación cruzada obtenidos con los datos originales y las muestras generadas con la estrategia Noise Bal. Código de colores (izq): igual que la Figura 7.14. 185
7.18. Valores la importancia de las variables en la proyección (VIP) derivados de la aplicación de PLS-DA al conjunto de datos original. 186
7.19. Valores de PLS-DA VIP obtenidos con el conjunto de datos original y las muestras generadas con la aplicación de SMOTE. 186
7.20. Valores de PLS-DA VIP obtenidos con el conjunto de datos original y las muestras generadas con el método Noise Bal (usando un 500 % de muestras sintéticas). 187
7.21. Valores de PLS-DA VIP obtenidos con el conjunto de datos original y las muestras generadas con el modelo CGAN. . . . 188

8.1. Diagrama de flujo del proceso de experimentación. Transformación de variables a componentes principales; división de los datos en conjuntos de entrenamiento, validación y test en un proceso de validación cruzada; aplicación del modelo generativo con el conjunto de entrenamiento y validación, y aumento con las muestras sintéticas; entrenamiento del clasificador y predicción con el conjunto de test. El proceso se repite con 10 semillas. 201
8.2. Comparación de la capacidad de predicción, especificidad y sensibilidad obtenidas usando diferentes modelos de DA, con respecto al logaritmo del número de instancias creadas. Código líneas: sólida azul, modelo CGAN; puntos naranja, ModCGAN con balanceo; guiones y puntos verde, Noise; guiones roja, Noise Bal. 204
8.3. Matriz de confusión normalizada obtenida con los datos originales (arriba izquierda) y realizando DA con NOISE Bal (arriba derecha), ACGAN (abajo izquierda) y ModCGAN_BM (abajo derecha). 206
8.4. Comparación del capacidad de predicción, especificidad y sensibilidad obtenidas usando un sistema de RF y de SVM mediante diferentes métodos de DA, con respecto al logaritmo del número de instancias creadas. Código líneas: igual que la Figura 8.2. 208

UNIVERSIDAD DE MÁLAGA

8.5. Matriz de confusión normalizada obtenida con diversos enfoques de DA. Se utiliza un sistema de RF como clasificador. . 210
8.6. Matriz de confusión normalizada obtenida con diversos enfoques de DA. Se utiliza un sistema de SVM como clasificador. 211
8.7. Gráfico de PCA con las muestras originales y las muestras generadas con los métodos SMOTE, Noise Bal, ACGAN y ModCGAN_BM para la predicción binaria del subtipo. Códigos de color: círculo verde, luminal; triángulo rojo, no luminal; cuadrado azul, luminal sintético; triángulo invertido morado, no luminal sintético. 212

9.1. Diagrama de flujo de todo el proceso de experimentación con el esquema meta-clasificador para el aumento de datos. 225
9.2. Comparación de las métricas de test obtenidas con los diferentes enfoques respecto al balanceo de los datos originales. Los puntos representan los resultados obtenidos con los datos originales, la cruz con los datos aumentados y las aspas con el esquema meta-clasificador. La línea continua es una regresión lineal ajustada a los resultados del esquema meta-clasificador. 234
9.3. Curvas ROC obtenidas utilizando los diferentes enfoques de aumento de datos para conjuntos de datos biomédicos. La línea punteada representa el azar; la línea discontinua representa el enfoque con los datos originales; la línea continua representa el aumento de datos; y la línea de puntos y guiones representa el uso del esquema meta-clasificador. 236
9.4. Tiempo de cálculo utilizado en los diferentes experimentos con una red neuronal. Código de colores: línea continua (punto), datos originales; línea de guiones y puntos (aspa) datos aumentados; línea discontinua (triángulo) esquema del meta-clasificador; línea de puntos (asterisco) datos aumentados y modelo DA basado en GAN; línea continua (cuadrado) esquema de meta-clasificador y un modelo DA basado en GAN. . . 240

A.1. Valores la importancia Gini de las variables aplicando Random Forest al enfoque binario, mostrando una selección de los 30 mayores valores (izquierda) y los 30 menores (derecha). 270
A.2. Valores la importancia Gini de las variables aplicando Random Forest al enfoque multiclase, mostrando una selección de los 30 mayores valores (izquierda) y los 30 menores (derecha). . . 271

B.1. Example of photorealistic generated faces using PG-GAN[17]. . 278
B.2. Example of synthetic sample generation with SMOTE. 281
B.3. Schematic diagram of the GAN model. 283

UNIVERSIDAD DE MÁLAGA

B.4. Generation of non-contrast brain CT scans from contrast-enhanced images[18]. 284
B.5. Scheme of the FNA procedure proposed for estimating the optimal level of noise. 286
B.6. The generative DA process for the creation of synthetic data. When the generative model is trained, synthetic samples are generated and the generative classifier predicts the appropriate labels. If the label predicted represents noise, sample is discarded and it generates another new sample. 290
B.7. Predictive ability using the original data, the best option for DA, WGAN-GP with 'Balanced Multiclass' 100 % DA, VAE with 'Multiclass' 100 % DA and CGAN with 100 % DA. Results are reported for four groups: all sets; grouped according to the number of instances; and the medical field. 293
B.8. ROC curves obtained using the original dataset and the data augmentation application. Dotted line represents chance; dashed line represents the approach with original data; and solid line represents data augmentation. 298
B.9. Flow diagram of the whole experimentation process. The dataset is divided in a cross-validation process. A generative model is fed with the Training set, and synthetic samples are created and added. Data are transformed to principal components (PCs) by following a PCA. Once the classifier is trained, the prediction is made with the Test set. The process is repeated with 5 different seeds. 302
B.10. PCA scores plot with: the original NPC1 disease dataset (top-left); the samples augmented with SMOTE (top-right); the samples augmented via the addition of noise with 2,000 % (bottom-left); and the samples augmented via CGAN with 2,000 % (bottom-right). Colour codes: green triangles, NPC1 disease urine; dark blue circles, heterozygous carrier control urine; red inverse triangles, generated NPC1 disease urine samples; light blue squares, generated heterozygous carrier control urine samples. 304
B.11. Accuracy, specificity and sensitivity obtained using a deep neural network as a classifier with different DA models with respect to the log of the number of instances created. Code lines: solid blue, CGAN model; orange dots, ModCGAN_BM; green dashes and dots, Noise; red dashes, Noise Bal. 309
B.12. Flow diagram of the whole experimentation process with the meta-classifier scheme for data augmentation. 312

UNIVERSIDAD DE MÁLAGA

UNIVERSIDAD
DE MÁLAGA

Índice de Tablas

3.1. Matriz de Confusión. 74
3.2. Interpretación de AUC 76

4.1. Valores de F_1 referencia obtenidos con los datos originales, número de características seleccionadas para el proceso DA, F_1 usando los datos aumentados (100 % DA) y diferencia relativa (RD). 95
4.2. Valores de F_1 referencia obtenidos con los datos originales, número de características seleccionadas para el proceso DA, F_1 usando los datos aumentados (200 % DA) y diferencia relativa (RD). 96
4.3. Valores de F_1 referencia obtenidos con los datos originales, número de características seleccionadas para el proceso DA, F_1 usando los datos aumentados (500 % DA) y diferencia relativa (RD). 96
4.4. Diferencias relativas (RD) de la métrica F_1 obtenidas medidas para los nueve conjuntos de datos de referencia utilizando diferentes niveles de aumento de datos (100 %, 200 % y 500 %). 97

5.1. Breve descripción de los diferentes modelos utilizados para los experimentos de DA. 111
5.2. Capacidad predictiva en test utilizando 50 %, 100 % y 200 % como porcentajes de datos aumentados, y la diferencia relativa máxima ($\widehat{RD}$) para todos los problemas de referencia ordenados según el número de instancias. Se aplicaron todos los métodos estudiados y, para el que produjo el menor error de validación, se muestran los resultados de test obtenidos. . . 116
5.3. Modelos que conducen a los mejores resultados de validación cuando se utilizan un 50 %, 100 % y 200 % de datos aumentados. 119
5.4. Número de veces que los diferentes modelos probados se eligen como mejores modelos (menor error de validación). El número total de casos (57) corresponde a los 19 conjuntos de datos considerados para los tres diferentes porcentajes de DA. . . . 120

UNIVERSIDAD DE MÁLAGA

5.5. Modelo seleccionado como mejor opción y sus resultados para tres casos: considerando todos los conjuntos de datos juntos o seleccionando el modelo para dos grupos de problemas según el número de instancias. 120

5.6. Capacidad predictiva en test obtenida cuando se utiliza como clasificador un modelo de regresión logística con los datos originales y los diferentes casos de datos aumentados. 122

5.7. Capacidad predictiva en test usando los datos originales y las muestras sintéticas generadas, y la mejor capacidad predictiva usando un 50 %, 100 % y 200 % de datos aumentados. 123

5.8. Capacidad predictiva en test obtenida con los datos originales y el promedio de los porcentajes de DA utilizados para los diferentes modelos generadores estándar, sobre los conjuntos del ámbito médico analizados. 123

6.1. Características del generador y discriminador que componen los modelos CGAN y ModCGAN utilizados para generar muestras sintéticas. 138

6.2. Características de la red neuronal utilizada como clasificador. 139

6.3. Capacidad de generalización obtenida con el conjunto original, tras haber realizado la selección de características y tras realizar DA, el modelo seleccionado y el porcentaje de muestras generadas, y el valor de $\widehat{RD}$ obtenido de acuerdo con esta selección. 141

6.4. RD obtenido con cada modelo de DA seleccionado de acuerdo con el mejor resultado de validación para cada conjunto de datos, el mejor valor de $\widehat{RD}$ obtenido de acuerdo con esta selección, y el modelo y $\widehat{RD}$ obtenidos si se selecciona por el mejor valor de test. 143

6.5. Modelo seleccionado como mejor opción y sus resultados para cuatro casos: considerando todos los conjuntos de datos juntos o seleccionando el modelo para tres grupos de problemas según el número de instancias. 147

6.6. Modelo seleccionado como mejor opción y sus resultados para cinco casos: considerando todos los conjuntos de datos juntos o seleccionando el modelo para cuatro grupos de problemas según el balanceo de clases. 148

6.7. Valor de AUC obtenido con el conjunto de referencia (FS) y con cada modelo de DA, el mejor AUC obtenido de acuerdo con esta selección, y el valor $\widehat{RD}$ correspondiente. 148

UNIVERSIDAD DE MÁLAGA

6.8. Modelo seleccionado como mejor opción y sus resultados considerando diferentes agrupaciones de los problemas: todos los conjuntos, según el número de instancias, y según el balanceo de clases. 149
6.9. Fréchet Inception Distance ($\widehat{FID}$) promedio obtenido con las muestras generadas por cada modelo de DA con respecto a las distribuciones reales de Train y Test, para cada conjunto de datos. Los valores más bajos representan una mayor similitud. 155
6.10. Standard Distance Deviation (SDD) obtenida con los datos originales y con las muestras generadas por cada modelo de DA, y desplazamiento del centro medio de la distribución (Shift) obtenidos para las muestras generadas con respecto a las distribución real. 157
6.11. Capacidad predictiva usando los datos originales, procesados con selección de características, solo las muestras sintéticas generadas, y con aumento de datos, y el mejor valor de $\widehat{RD}$ obtenido acorde a estos resultados. 159

7.1. Número de muestras generadas y tamaño del conjunto de entrenamiento aumentado, según el porcentaje de DA aplicado. La aplicación de SMOTE siempre crea el mismo número de muestras. 167
7.2. Resultados de test obtenidos utilizando regresión logística como clasificador (fuente negrita indica mejor valor). Comb 1 es una combinación de muestras creadas con las estrategias CGAN y SMOTE. Comb 2 es una combinación de muestras creadas con los enfoques CGAN y NOISE Bal. 171
7.3. Resultados de test obtenidos utilizando un sistema de Random Forest como clasificador. Comb 1 y Comb 2: descritas en la Tabla 7.2. 174
7.4. Resultados de test obtenidos utilizando un sistema de SVM como clasificador. Comb 1 y Comb 2: descritas en la Tabla 7.2. 176
7.5. Resultados de test promedios obtenidos con la aplicación de los diferentes clasificadores. Comb 1 y Comb 2 representan las combinaciones de creación de muestra descritas en la Tabla 7.2. 178
7.6. Valores de PLS-DA VIP obtenidos con los diferentes métodos de DA (los paréntesis representan la posición de ranking relativo de cada metabolito para cada método aplicado). . . . 189

8.1. Características del discriminador del modelo ACGAN utilizado para generar muestras sintéticas. 198
8.2. Características de la red neuronal utilizada como clasificador. 199

UNIVERSIDAD DE MÁLAGA

8.3. Número de muestras generadas y tamaño del conjunto de entrenamiento aumentado, según el porcentaje de DA aplicado. La aplicación de SMOTE siempre crea el mismo número de muestras. 200

8.4. Resultados de test obtenidos para la predicción binaria del subtipo molecular, con los distintos métodos de DA aplicados. 203

8.5. Resultados de test obtenidos para la predicción multiclase del subtipo, con los distintos métodos de DA aplicados. 205

8.6. Resultados de test obtenidos para el problema binario, con los distintos métodos de DA utilizando un Random Forest (RF) y un sistema de máquinas de vectores de soporte (SVM) como clasificador. 207

8.7. Resultados de test obtenidos para la predicción multiclase, con los distintos métodos de DA utilizando un Random Forest (RF) y un sistema de máquinas de vectores de soporte (SVM) como clasificador. 209

8.8. Fréchet Inception Distance ($\widehat{FID}$), Standard Distance Deviation (SDD) y desplazamiento del centro medio de la distribución (*Shift*) obtenidos con las muestras generadas por cada modelo de DA con respecto a las distribuciones reales de las clases binarias. 214

9.1. Características del generador y discriminador que componen los modelos CGAN y ModCGAN utilizados para generar muestras sintéticas. 229

9.2. Resultados de test (capacidad predictiva y área bajo la curva ROC) obtenidos con los datos originales, los datos aumentados y aplicando esquema de meta-clasificador para todos los problemas de referencia. 231

9.3. Resultados de test (sensibilidad y especificidad) obtenidos con los datos originales, los datos aumentados y aplicando esquema de meta-clasificador para todos los problemas de referencia. 232

9.4. Resultados de test (F_1 y Mathews Correlation Coefficient) obtenidos con los datos originales, los datos aumentados y aplicando esquema de meta-clasificador para todos los problemas de referencia. 233

9.5. Resultados obtenidos utilizando el esquema de datos aumentados y meta-clasificador con cada método de DA aplicado en comparación con los resultados de los datos originales. Los resultados de capacidad predictiva y AUC se calcularon utilizando RD. 235

UNIVERSIDAD DE MÁLAGA

9.6. Resultados de test utilizando una red neuronal como modelo clasificador, entrenada con los datos originales, los datos aumentados mediante un procedimiento simple, y aumentados aplicando el esquema de meta-clasificador, para los problemas biomédicos de referencia. 238

9.7. Resultados obtenidos utilizando aumento simple y el meta-clasificador con cada método de DA aplicado en comparación con el conjunto de datos original. Los resultados de capacidad predictiva y AUC se calcularon utilizando RD. Se emplea una red neuronal como clasificador. 239

A.1. Características de los nueve conjuntos de datos de referencia seleccionados para llevar a cabo los experimentos. 257

A.2. Porcentaje de muestras en las clases minoritarias de los conjuntos de datos de referencia y el método de procesamiento utilizado para la reconversión a problema binario. 258

A.3. Características de los diecinueve conjuntos de datos de referencia seleccionados para llevar a cabo los experimentos. . . . 260

A.4. Número de neuronas presentes en las capas ocultas de la red neuronal aplicada en los experimentos para los conjuntos de datos. 261

A.5. Características de los dieciocho conjuntos de datos de referencia seleccionados para llevar a cabo los experimentos. 263

A.6. Distribución de los grupos en el conjunto de datos ^{1}H NMR. . 266

A.7. Distribución de las variables moleculares HLADBR-1, MIC-A y HLA-B. 267

A.8. Distribución de las variables moleculares CTLA-4, NKG2DT, LMP y TAP. 268

A.9. Distribución del subtipo molecular en los datos. 269

A.10. Características de los doce conjuntos de datos de referencia seleccionados para llevar a cabo los experimentos. 273

B.1. F1 relative differences (RD) obtained measured for the nine benchmark datasets using different levels of Data Augmentation (100 %, 200 %, and 500 %). 287

B.2. Test accuracy using 50 %, 100 % and 200 % of augmented data, and maximum relative difference ($\widehat{RD}$) for all benchmark problems ordered according to the number of instances. . . . 292

B.3. Best option model and their results for three cases: considering all datasets together or selecting the model for two groups of problems according to the number of instances. 293

UNIVERSIDAD DE MÁLAGA

B.4. Test accuracy obtained (cols. 2-5) with the original dataset, when a feature selection method is applied, with DA methods applied, and the best RD obtained ($\widehat{RD}$). Also, test AUC obtained (cols. 6-8) with feature selection, with DA and the best RD obtained. 296
B.5. Test accuracy RD and AUC RD obtained with each DA model selected according to the best validation result for each dataset. 297
B.6. Average Fréchet Inception Distance ($\widehat{FID}$) and Standard Distance Deviation (SDD) obtained with the samples generated by each DA model, for each dataset. Lower values of $\widehat{FID}$ represent greater similarity. 300
B.7. Average test results acquired with the application of the different classifiers. Comb 1 is a combination of samples created with the CGAN and SMOTE strategies. Comb 2 is a combination of samples created with the CGAN and NOISE Bal approaches. 303
B.8. PLS-DA VIP values obtained with the different DA methods and the relative rank position of each metabolite variable. . . 305
B.9. Test results obtained for the binary subtype prediction using a deep neural network as a classifier with the different DA methods. 308
B.10. Test results obtained for the multiclass subtype prediction using a deep neural network as a classifier with the different DA methods. 308
B.11. Fréchet Inception Distance ($\widehat{FID}$), Standard Distance Deviation (SDD) and shift of the mean center of the distribution (*Shift*) obtained with the samples generated by each DA model with respect to the real distributions of the binary classes. 310
B.12. Test results using a deep neural network as a classifier model, trained with the original data, the augmented data and the meta-classifier scheme for the benchmark problems. 314
B.13. Results obtained using a deep neural network as a classifier model for the augmented data and meta-classifier scheme with each DA method, compared with original data results. 315

UNIVERSIDAD DE MÁLAGA

Índice de Abreviaciones

ACGAN Auxiliary Classifier GAN.

AUC Área Bajo la Curva ROC.

BM Balanced Multiclass.

CGAN Conditional GAN.

CVAE Conditional GAN.

CNN Convolutional Neural Network.

DA Data Augmentation, Aumento de Datos.

DVAE Denoising VAE.

DL Deep Learning, Aprendizaje Profundo.

FID Fréchet Inception Distance.

FNA Forward Noise Adjustment.

FNN Feedforward Neural Network.

GAN Generative Adversarial Network

GPU Unidad de Procesamiento Gráfico.

IA Inteligencia Artificial.

LASSO Least Absolute Shrinkage and Selection Operator.

LR Regresión Logística.

M Multiclass.

UNIVERSIDAD DE MÁLAGA

MCC Mathews Correlation Coefficient.

ML Aprendizaje Automático.

NPC Niemman-Pick Tipo C.

PCA Análisis de Componentes Principales.

PLS-DA Análisis Discriminatorio de Mínimos Cuadrados Parciales.

RD Diferencia Relativa.

ReLU Rectified Linear Unit.

RF Random Forest.

RNA Ácido Ribonucleico.

RNA-Seq Ácido Ribonucleico Secuenciado.

RNN Recurrent Neural Network.

ROC Reciever Operating Characteristic.

SDD Standard Distance Deviation.

SMOTE Synthetic Minority Oversampling Technique.

SVM Máquina de Vectores de Soporte.

TCGA The Cancer Genome Atlas.

VAE Variational Autoencoder.

VIP Variable Importance in Projection.

WGAN Wassertein GAN.

UNIVERSIDAD DE MÁLAGA

UNIVERSIDAD
DE MÁLAGA

UNIVERSIDAD
DE MÁLAGA

Parte I

Contexto y Fundamentos

UNIVERSIDAD
DE MÁLAGA

UNIVERSIDAD
DE MÁLAGA

Capítulo 1

Introducción

Resumen: Este capítulo presenta una breve introducción al trabajo de investigación realizado en el desarrollo de técnicas de aumento de datos y su aplicación en bioinformática, incluyendo la motivación del trabajo, los principales objetivos planteados al inicio y la metodología empleada en la presente tesis.

Esta investigación se encuentra integrada dentro de los proyectos desarrollados en el grupo de Inteligencia Computacional y Biomedicina (ICB) del Departamento de Lenguajes y Ciencias de la Computación de la Universidad de Málaga. El equipo investigador presenta una trayectoria consolidada en los últimos 15 años en la aplicación de la inteligencia artificial en el campo de la biomedicina, siendo financiado en sucesivas convocatorias por el Plan Nacional de I+D+i y la Junta de Andalucía a través de la ejecución de diferentes proyectos, todos ellos enfocados a la bioinformática, y en los que se ha avanzado en el diseño de algoritmos aprendizaje neuro-computacionales para análisis de supervivencia en cáncer (TIN2005-02984), algoritmos constructivos de redes neuronales artificiales para el procesamiento de información y minería de datos en oncología (TIN2008-04985, TIC-4026), y sistemas inteligentes bioinspirados aplicados a medicina personalizada (TIN2010-16556). Un indicio de la calidad del trabajo previo desarrollado por el equipo investigador es el compromiso mostrado por diferentes entidades/empresas (a través de la firma de contratos/convenios tipo art. 83 LOU) en la transferencia de resultados y tecnología derivada de la ejecución de los proyectos de investigación mencionados. A este respecto, el grupo forma parte activamente de la Unidad de Gestión Clínica Oncología Intercentros (UGCOI) del Hospital Universitario Virgen de la Victoria (HUVV) y el Hospital Regional Universitario (HRU) de Málaga, el Instituto de Investigaciones Biomédicas de Málaga (IBIMA) y la Fundación Pública Andaluza para la Investigación de Málaga en Biomedicina y Salud (FIMABIS). Uno de los mayores logros alcanzados por el grupo investigador es el sistema de información Galén (Ri-

UNIVERSIDAD DE MÁLAGA

belles et al., 2010), desarrollado en colaboración con personal clínico experto, y que actualmente almacena información relativa a más de 58.000 pacientes de cáncer (1978 - 2022) en la provincia de Málaga, siendo el grupo ICB responsable de la aplicación de algoritmos de IA a los datos bioinformáticos generados por el grupo de biología molecular del cáncer de la UGCOI.

Con el impulso en la última década de los modelos de aprendizaje profundo (DL) frente a los métodos de aprendizaje tradicionales, el grupo ICB ha trabajado de forma paralela en el diseño y adaptación de algoritmos de DL para su aplicación a problemas en el campo de la biomedicina, en el marco de ejecución de un proyecto financiado por el Plan Nacional I+D+i (TIN2017-88278). Estos últimos años han estado enfocados en dos estrategias para mejorar la aplicación de estos modelos profundos a datos de la bioinformática, el *transfer learning* y el *data augmentation* (aumento de datos, DA), siendo este último enfoque el tema central del presente trabajo de tesis. La motivación del trabajo desarrollado y el problema de estudio, la aplicación de DL y DA con conjuntos bioinformáticos, se encuentra expuesto de forma más amplia y profunda a continuación.

1.1. Motivación

Uno de los campos más prósperos de las ciencias de la computación es la inteligencia artificial (IA). Los problemas a resolver mediante la IA han ido cambiando con el tiempo, conforme ha avanzado esta tecnología. El gran reto de las últimas décadas al que se ha enfrentado la IA es la resolución de tareas que para los seres humanos resultan intuitivas, tales como el reconocimiento de objetos en imágenes o el procesamiento del lenguaje natural, pero de las que resulta complicado extraer reglas formales que pueda seguir un computador. Después del estudio de diferentes enfoques, la solución más prometedora es la concesión a las máquinas de la capacidad de aprender de la experiencia, entendiendo el mundo en términos de jerarquía de conceptos en vez de un conjunto de reglas matemáticas. En la jerarquía de conceptos, el mundo se define en forma de conceptos y su relación entre ellos, de forma que el agente inteligente aprende conceptos complejos construyéndolos a partir de conceptos más sencillos, evitando la especificación formal de todo el conocimiento posible. La concesión a las máquinas de la capacidad de aprender se conoce como aprendizaje automático o *machine learning* (ML), y ha sido el principal contribuyente al resurgimiento de la IA. Especialmente en la última década, el enfoque con mayor progresión y éxito del ML moderno se centra en el modelado mediante redes neuronales artificiales con múltiples etapas de procesamiento, actualmente conocido como aprendizaje profundo, representado comúnmente mediante su anglicismo *deep learning*.

El desarrollo del DL está relacionado con la organización del sistema nervioso de los mamíferos, fundamentándose en primer lugar en el funciona-

UNIVERSIDAD DE MÁLAGA

miento del córtex visual en gatos y primates (Hubel y Wiesel, 1959, 1962, 1968). Actualmente se ve influenciado por el complejo funcionamiento del sistema nervioso humano. La neurociencia ha descubierto áreas diferenciadas de redes neuronales especializadas en la realización de tareas específicas en nuestro sistema, las cuales son objeto de réplica en IA. Hay constancia de que existen diferentes áreas neuronales centradas en la detección de distintos aspectos de la visión, tales como el reconocimiento de bordes (Marr y Hildreth, 1980; Canny, 1986), el reconocimiento del lenguaje (Arbib, 2005), el reconocimiento de rostros (Chellappa et al., 1995) y la expresión emocional de los mismos (Adolphs et al., 1994). De esta forma, en el aprendizaje profundo se utilizan estructuras lógicas que se asemejan a las características del sistema nervioso, teniendo un número elevado de diferentes capas de unidades de proceso especializadas en la detección de determinadas características existentes, pero ocultas, en los datos observados. La diferencia principal entre los modelos de aprendizaje profundo y las redes neuronales artificiales convencionales es el número de neuronas y capas ocultas que presentan, ya que en las redes convencionales se utiliza una única capa oculta acompañada de las capas de entrada y salida, arquitectura del modelo conocido como perceptrón multicapa. Por el contrario, en el enfoque profundo se han desarrollado diferentes arquitecturas basadas principalmente en la incorporación de capas ocultas adicionales en redes *feedforward*, incluyendo diseños novedosos como las redes convolucionales o las redes recurrentes.

Las redes neuronales artificiales han demostrado ser capaces de aproximar cualquier función no lineal (Hornik et al., 1989). Sin embargo, este modelo ha sido habitualmente criticado por representar una caja negra, en el sentido que la falta de interpretabilidad limitó sus aplicaciones, especialmente cuando su rendimiento no destacaba entre otros métodos de ML más interpretables, como la regresión logística, las máquinas de vectores de soporte y los árboles de decisión. No obstante, durante la última década han surgido tres avances importantes que han llevado al rejuvenecimiento de las redes neuronales gracias al DL. En primer lugar, se generan y recopilan en el mundo moderno una cantidad de datos sin precedentes, planteando grandes desafíos en su aplicación con los métodos de ML convencionales. Sin embargo, las redes neuronales han demostrado manejar estos grandes volúmenes de datos con un gran rendimiento. En segundo lugar, el poder de cómputo ha aumentado rápidamente manteniendo costos relativamente asequibles, especialmente gracias a los nuevos dispositivos hardware con alta capacidad de cómputo en paralelo como las unidades de procesamiento de gráficos (GPU) (Shi et al., 2016). En tercer lugar, se han desarrollado diferentes algoritmos de optimización que han hecho que las redes neuronales profundas destaquen como el modelo ideal para el análisis de grandes volúmenes de datos.

En general, el nuevo enfoque del DL ha posibilitado la creación de modelos capaces de obtener un mejor rendimiento en la mayoría de campos

UNIVERSIDAD DE MÁLAGA

de investigación en los que han sido aplicados, destacando el procesamiento del lenguaje natural y la visión por computador, motivo por el cual ha captado nuestra atención, y es parcialmente objeto de este trabajo de tesis. El DL ha revolucionado además las competiciones de ML en las áreas de reconocimiento de patrones. En 2003 una red convolucional logro el récord en reconocimiento del conjunto MNIST (Simard et al., 2003), y una red recurrente ganó el ICDAR 2009 en reconocimiento de escritura árabe, farsi y francés (Grosicki y El Abed, 2009). Pero es a partir de la victoria en ImageNet 2012 de una red convolucional (Krizhevsky et al., 2012), cuando se suceden más logros de este enfoque en otras competiciones, entre las que destacamos el ICPR 2012 de detección de mitosis en imágenes histológicas de cáncer de mama (Ciresan et al., 2013), el *EM Stacks Challenge* de segmentación de imágenes (Ciresan et al., 2012), y el ImageNet 2014 (Szegedy et al., 2015), el récord en reconocimiento del habla TIMIT (Graves et al., 2013), o las aplicaciones en el reconocimiento de números de Google Street View (Goodfellow et al., 2013a), y el juego Go (Silver et al., 2016). Estos solo representan unos pocos ejemplos, ya que a partir de 2016 el número de artículos publicados relacionados con el DL se dispara exponencialmente, como puede observarse en la Figura 1.1.

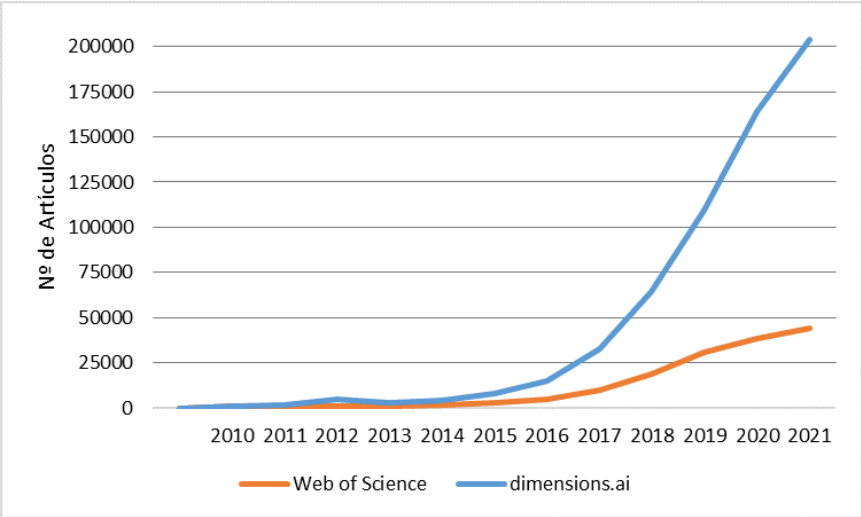

Figura 1.1: Número de artículos científicos publicados relacionados con aprendizaje profundo desde 2010.

El DL se ha convertido pues en el estado del arte para muchas tareas de ML, lo que se traduce en miles de millones de dólares invertidos por las industrias en este tipo de técnicas de inteligencia artificial. Por ejemplo, los métodos basados en redes neuronales convolucionales (CNN) dominan las tres áreas principales en el campo de la visión por computador, el reconocimiento de imágenes, la detección de objetos y la superresolución (Ren et

UNIVERSIDAD DE MÁLAGA

al., 2015; Szegedy et al., 2015; Yue et al., 2016); y los métodos basados en redes neuronales recurrentes (RNN) presentan el mejor rendimiento en una amplia gama de tareas en procesamiento del lenguaje natural, como la clasificación de textos, el reconocimiento de voz y la traducción automática (Graves et al., 2013; Sutskever et al., 2014; Yao et al., 2019). En un campo relevante socialmente como la bioinformática, el DL también ha mostrado un crecimiento explosivo en su aplicación, demostrando un potencial prometedor para explotar la compleja relación oculta en los datos biológicos y biomédicos a gran escala. A este respecto, existen varios estudios que revisan la aplicación del aprendizaje profundo a la bioinformática (Angermueller et al., 2016; Li et al., 2020), puesto que su rendimiento aplicado es un hecho innegable en el análisis de secuencias DNA y RNA (Alipanahi et al., 2015; Umarov et al., 2019), la predicción y reconstrucción de estructuras celulares y de proteínas (Wang et al., 2015; Zeng et al., 2016), el análisis y diseño de fármacos (Wallach et al., 2015; Karimi et al., 2019), y el procesamiento de imágenes biomédicas (diagnóstico y segmentación) (Ciresan et al., 2013; Havaei et al., 2017; Hu et al., 2020; Shin et al., 2020).

Como se ha mencionado, el DL tiene una estructura de modelo compleja en comparación con otros métodos de ML, con arquitecturas que incluyen varias capas ocultas con miles de parámetros que deben ajustarse durante el entrenamiento. Debido a esto, normalmente se requieren grandes conjuntos de datos con valores de miles de instancias para obtener buenos resultados en la aplicación de los métodos de DL (LeCun et al., 2015; Schmidhuber, 2015; Goodfellow et al., 2016). Generalmente, en el mundo actual esto no supone un problema, sin embargo, existen diferentes áreas con gran impacto social en las que la adquisición de datos para la aplicación de modelos de DL es una cuestión difícil y costosa en tiempo, organización y dinero. Y aunque hemos mencionado que en la bioinformática hay varias tareas para las que el DL presenta un rendimiento superior al resto de modelos de aprendizaje, hay otras tareas en las que esto no ocurre. Esto puede deberse a cuatro factores presentes en algunas tareas de la bioinformática, siendo el primero la escasez de datos disponibles para su resolución. Los registros y muestras médicas son datos sensibles, que pueden incurrir en problemas legales complejos de privacidad, y es difícil obtener el consentimiento del paciente para su difusión masiva. Este problema se vuelve más acuciante cuando se estudian enfermedades raras o en una región geográfica concreta. El segundo factor es la complejidad de la información representada en los conjuntos biológicos, lo que, unido a la complejidad de los modelos de DL con una gran cantidad de parámetros, provoca que estos sufran de un alto riesgo de sobreajuste a los datos de entrenamiento y una falta de generalización en test. En este sentido, existen algunos enfoques que intentan solucionarlo, como el *transfer learning* (Cai et al., 2020) o la recopilación de información biológica de diversas fuentes (Mamoshina et al., 2016). Otro factor se presenta porque los datos no

UNIVERSIDAD DE MÁLAGA

solo son escasos, sino que además suelen estar desequilibrados y las muestras que representan una clase son superadas en gran medida por las muestras del resto de clases. Por ejemplo, el número de muestras de control de una enfermedad suele ser mucho mayor que las muestras positivas o, por ejemplo, hay ciertos subtipos de cáncer mucho más frecuentes que otros. Por último, el DL presenta un rendimiento limitado cuando se aplica en problemas de clasificación/predicción que no presentan un conjunto de datos con patrones 2D/3D, secuencial u otro tipo de información estructurada. Esto se observa claramente en que los modelos de DL son el estado del arte en procesamiento de imágenes y lenguaje natural gracias a los modelos convolucionales y recurrentes, que son capaces de extraer la información estructurada.

De este modo, todavía hay áreas y tareas en la bioinformática que presentan limitaciones. Cabe destacar que las limitaciones de rendimiento debido a la escasez de las muestras disponibles o el desequilibrio de las clases, no solo está presente en la aplicación con los modelos de DL, sino que afecta a todos los modelos de ML. Más específicamente, en comparación con los conjuntos de datos de imágenes utilizados en problemas biomédicos, los conjuntos de datos de la exploración de microarrays de expresión génica, expresión molecular o metabolómica utilizados con el propósito de mejorar el diagnóstico de los pacientes, predicción de eventos en cáncer o del subtipo molecular, presentan una problemática mayor. Estos son significativamente más difíciles de obtener, tienen una mayor dispersión y presentan la llamada maldición de la dimensionalidad, es decir, tienen un gran número de características y un número comparativamente muy pequeño de muestras. En la literatura se encuentran trabajos publicados que aplican DL con este tipo de conjuntos (Fakoor et al., 2013; Danaee et al., 2016; Urda et al., 2017; Montesinos-López et al., 2019; Zhu et al., 2020; Pomyen et al., 2020; Talukder et al., 2021), presentando unos resultados menos exitosos en comparación con la eficacia de estos algoritmos en los problemas ya mencionados con anterioridad.

Para resolver la limitación presente cuando se tiene un número de instancias reducido en el conjunto de datos disponible y, en cierta forma, también el desequilibrio de los datos, se puede emplear la estrategia denominada aumento de datos (*data augmentation*, DA). Esta estrategia consiste esencialmente en aumentar la cantidad de instancias de entrenamiento disponibles, bien aplicando diversos métodos que transforman instancias ya presentes en el conjunto, o mediante la creación de instancias sintéticas totalmente nuevas. Los métodos más simples de DA han sido aplicados frecuentemente en el procesamiento de imágenes, donde se pueden encontrar varios métodos de transformación simples pero efectivos, aplicando modificaciones geométricas que no alteran la categoría de las muestras, como la inversión, la rotación, el recorte o el desplazamiento (Shijie et al., 2017). En cada época de entrenamiento del modelo de ML se puede aplicar una transformación a cada imagen de entrada, creando distintas instancias que aumentan el tamaño del

UNIVERSIDAD DE MÁLAGA

conjunto de datos. Un ejemplo de una transformación se ilustra en la Figura 1.2, donde se muestra una imagen original del conjunto de datos CIFAR-10 (izquierda) junto con una imagen transformada creada por una rotación de 15° (derecha). Al igual que los modelos de DL, la mayoría de los éxitos del DA se encuentran en la aplicación de problemas cuyos conjuntos de datos consisten en imágenes, cuyo mayor exponente aparece en el desafío de reconocimiento visual a gran escala de ImageNet (Russakovsky et al., 2015), donde la mayoría de los modelos ganadores son entrenados con DA (Simonyan y Zisserman, 2014; Szegedy et al., 2015; He et al., 2016).

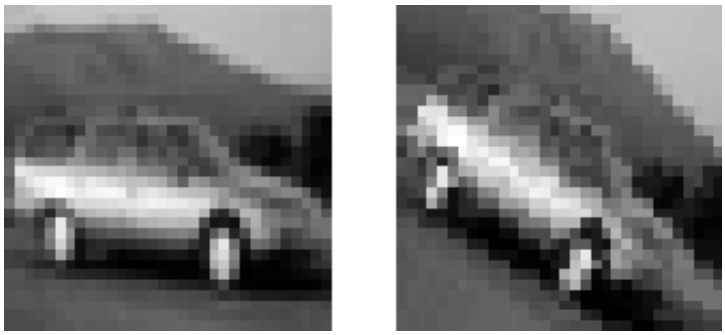

Figura 1.2: Ejemplo del proceso de aumento de datos aplicado a una imagen del conjunto de datos CIFAR-10. Izquierda: imagen de muestra original. Derecha: imagen creada a través de una rotación de 15°.

Dentro de las técnicas y métodos de DA más actuales, se encuentran algunos modelos que además están enmarcados dentro del DL. Entre estos destacan los *Variational Autoencoders* (VAEs) (Kingma y Welling, 2013) y las *Generative Adversarial Networks* (GANs) (Goodfellow et al., 2014a), dos modelos que han atraído mucho interés en los últimos años. Estos modelos se engloban dentro de los denominados modelos generativos profundos, cuyo propósito es aprender la distribución de datos a partir de un conjunto de instancias para posteriormente generar instancias sintéticas extraídas de la distribución aprendida. Aunque comparten propósito, ambos modelos se basan en ideas diferentes. Los VAE se basan en las redes *autoencoders* (Bourlard y Kamp, 1988), que se pueden considerar una versión previa de los VAE, y tratan de codificar el conjunto de datos capturando la distribución original en un espacio latente reducido a través de una distribución gaussiana multivariante. Mientras, los modelos GAN se basan en la idea de producir una confrontación entre dos redes neuronales adversarias que aprenden entre sí. La red denominada generador tiene el objetivo de capturar la distribución original y generar datos sintéticos, intentando engañar a la segunda red, denominada discriminadora, que tiene como objetivo discernir si la instancia recibida es verdadera o falsa. Partiendo de ambos modelos generativos profundos básicos, se han propuesto diversas variantes mediante la modificación

UNIVERSIDAD DE MÁLAGA

de la arquitectura de red, la función de pérdida o la inclusión de información adicional. Algunas de estas variantes muestran un gran rendimiento, siendo capaces de generar imágenes falsas que los expertos pueden considerar reales (Radford et al., 2015; Pu et al., 2016; Karras et al., 2018; Zhu et al., 2017). A parte de la generación de imágenes, los modelos generativos profundos se aplican otras tareas de visión por computador, como la mejora de la resolución de imágenes (Ledig et al., 2017) o la transferencia de imágenes a otros estilos (Isola et al., 2017), pero en particular han demostrado ser muy útiles para realizar DA (Hsu et al., 2017b; Frid-Adar et al., 2018; Han et al., 2018; Waheed et al., 2020).

Recientemente, los modelos GAN también se están empezando a utilizar para realizar tareas como la clasificación de texto (Xu et al., 2017), el procesamiento de voz y lenguaje natural (Hsu et al., 2017a; Pascual et al., 2017), y para realizar DA con otros tipos de datos estructurados como series de tiempo o señales (Hsu et al., 2017b; Sheng et al., 2018; Shao et al., 2019). Sin embargo, aplicar DA a conjuntos de datos que no presentan estructura espacial o temporal, o no tienen ningún tipo de información estructurada, es una tarea más desafiante. Se puede pedir a los expertos en un dominio que evalúen la calidad de una imagen o un texto generado, incluso de una serie temporal, y se puede verificar si son capaces de distinguir una muestra sintética generada de una real. Sin embargo, este tipo de evaluación con expertos humanos no es factible en su aplicación sobre datos que no presentan información estructurada, menos aún si tenemos en cuenta los complejos datos del ámbito de la bioinformática, siendo casi imposible identificar si una muestra biológica es real o sintética. Para esta tarea de DA, además de algunas aplicaciones de las técnicas SMOTE (técnica de sobremuestreo de minorías sintéticas) (Chawla et al., 2002), diseñadas para tratar con conjuntos de datos que tienen clases desbalanceadas, el resto de la literatura existente antes de 2015 se relaciona principalmente con técnicas de inyección de ruido como una forma de evitar el sobreajuste y mejorar la capacidad de predicción (Reed y Marks, 1998; Zur et al., 2009; Fernández-Navarro et al., 2011; Piotrowski y Napiorkowski, 2013). Si bien, en los últimos años, y de forma casi en paralelo con el presente trabajo de tesis, los modelos GAN también se están aplicando con conjuntos de datos sin ningún tipo de estructura espacial o temporal en una estrategia de DA (Douzas y Bacao, 2018; Liu et al., 2019; dos Santos Tanaka y Aranha, 2019).

Para finalizar esta introducción, se puede resumir el objetivo de este trabajo de tesis como la búsqueda y diseño de estrategias novedosas en aumento de datos en conjunto con el aprendizaje profundo, con objeto de mejorar su rendimiento en la aplicación a problemas de naturaleza bioinformática, tales como el diagnóstico médico a partir de datos de naturaleza genómica, la predicción de eventos en pacientes o el subtipo molecular de un cáncer, problemas de gran actualidad e impacto social.

UNIVERSIDAD DE MÁLAGA

1.2. Objetivos

Nuestro principal objetivo es diseñar, implementar y evaluar nuevos métodos y modelos robustos de aumento de datos para su aplicación en problemas del ámbito de la bioinformática. Las aplicaciones de DL con conjuntos de datos bioinformáticos de carácter ómico presentan un rendimiento menor en comparación con otros conjuntos. Este hecho tiene su raíz en cuatro características que comúnmente presentan los conjuntos ómicos: la escasez de muestras disponibles; la complejidad de la información biológica representada; el desequilibrio en el número muestras que representan las clases, por ejemplo, control/enfermedad; y la ausencia de patrones 2D/3D, secuencial u otro tipo de información estructurada.

Por esta razón, es objeto de estudio la aplicación de DA y DL con conjuntos de datos que no posean información ni estructura espacial o temporal y, especialmente, presenten un número reducido de muestras disponibles. La mayoría de métodos de DA desarrollados para lidiar con problemas sin estructura espacial o temporal se basan en técnicas de sobremuestreo o en técnicas de inyección de ruido para evitar el sobreajuste. Por lo tanto, uno de los objetivos será el desarrollo de métodos basados en la adición y ajuste de ruido creando muestras sintéticas aplicadas en un proceso de DA para mejorar el rendimiento de la tarea de predicción.

Considerando el objetivo de mejorar el rendimiento de la aplicación de DL, hay que tener en cuenta que el gran éxito de estos modelos se centra en el procesamiento de imágenes y vídeos con CNN. Sin embargo, el rendimiento de estos modelos decae con conjuntos de imágenes con un número de instancias reducido. Por ello el principal esfuerzo de los investigadores ha sido crear técnicas para aumentar los conjuntos de imágenes mediante la creación de imágenes sintéticas. Es fundamental estudiar la base del DA en imágenes con el objetivo de poder extraer las claves de los métodos y técnicas utilizadas, lo que permitirá abordar con mayores conocimientos los métodos de DA para aplicarlos a problemas sin estructura espacial o temporal. Específicamente, el DA basado en estructuras profundas ha demostrado un éxito impresionante en la generación de imágenes realistas, basándose en dos marcos de DL (VAE y GAN). Un objetivo general va a ser la adaptación de estas técnicas a problemas que no presentan información espacial. Con ello, no solo pretendemos obtener un mejor rendimiento en la aplicación de DL con conjuntos de datos no estructurados y con un número reducido de patrones, sino que pretendemos innovar utilizando directamente modelos generativos profundos con este tipo de datos.

La eficacia de los modelos desarrollados debe ser demostrada inicialmente con conjuntos de datos y problemas bien conocidos, que se encuentren disponibles de forma abierta. Con ello, se podrá demostrar su uso generalizado en problemas con diferentes características específicas, logrando una mayor

UNIVERSIDAD DE MÁLAGA

difusión. Además, el análisis de estos resultados nos ayudará a seleccionar los modelos de DA a aplicar en problemas del campo de la bioinformática, además de propiciar nuevos desarrollos de modelos. Para cumplir con estos objetivos generales se han abordado los siguientes objetivos más específicos:

- Estudiar los métodos y modelos de aumento de datos para tratar con conjuntos sin estructura espacial o temporal y un número reducido de muestras, aumentando el conocimiento existente sobre el problema.
- Estudiar los modelos tradicionales y de última generación sobre generación de imágenes para aumento de datos, analizando las cualidades y características presentes en ellos para trasladarlos a otros problemas.
- Desarrollar técnicas de aumento de datos basados en la adición y el ajuste de ruido para crear muestras sintéticas.
- Desarrollar métodos basados en modelos generativos profundos, particularmente las VAEs y GANs, para su aplicación en conjuntos sin estructura espacial o temporal y un número reducido de muestras.
- Estudiar la aplicación de los métodos de aumento de datos desarrollados en problemas del ámbito de la biomedicina/bioinformática, y comparar su rendimiento con diferentes métodos del estado del arte.

1.3. Metodología

El procedimiento metodológico empleado como base de nuestra investigación es el denominado método científico. Más concretamente el método empírico-analítico, cuyo uso es propicio con problemas experimentales (March y Smith, 1995). Una investigación desarrollada siguiendo esta metodología estará sujeta a los principios de reproducibilidad y refutabilidad. De este modo los experimentos realizados pueden ser repetidos con total independencia, y estos son suficientes para poder rechazar la hipótesis. Este método se caracteriza por seguir las siguientes etapas:

- Estudio del ámbito de investigación. Es necesario en primer lugar un estudio del estado del arte del tema a tratar en la investigación. El análisis de los problemas reales y los modelos existentes es esencial para construir y aplicar nuevos modelos con el fin de resolver los problemas.
- Formulación de un problema. Proposición de una cuestión que se desea resolver o estudiar tras la observación inicial.
- Identificación de factores importantes. Es necesario identificar qué factores están involucrados en el problema y su relación causal con el mismo. Es necesario observar los factores involucrados para deducir posibles limitaciones y soluciones.

UNIVERSIDAD DE MÁLAGA

- Formulación de una hipótesis. Elaboración de un enunciado o solución que explica de manera provisional los hechos observados, así como las causas posibles que los motivan.

- Diseño y realización del experimento. Desarrollo de los experimentos mediante la aplicación de los modelos, métodos y técnicas sobre conjuntos de juguete y reales, con el objetivo de comprobar la veracidad de la hipótesis.

- Análisis y prueba de la hipótesis. Análisis de los datos obtenidos durante la experimentación mediante los criterios de evaluación establecidos, de tal manera que se pueda extraer una conclusión y armar la veracidad de nuestra hipótesis o bien desecharla.

- Diseminación de resultados. Una vez estudiado los resultados producidos, se procederá a la difusión al conocimiento científico generado tras la elaboración del proyecto.

Para el desarrollo de cada sección del proyecto de investigación se seguirá una metodología iterativa e incremental. Mediante esta metodología se parte de un proyecto base al cual se añaden métodos y funcionalidades, sin que esta adición afecte de manera negativa al resto del proyecto ya implementado.

Para la implementación de los métodos, técnicas y modelos diseñados se debe utilizar un estilo de programación adecuado, de forma que estos sean modulares, fácilmente comprensibles y modificables, extensibles, optimizados, etc. Además, deberán estar bien documentados para facilitar su comprensión y uso por una persona inexperta en la materia.

También es necesario establecer un criterio de evaluación y de calidad para los algoritmos y soluciones propuestas. Para ello, se hará uso de medidas de rendimiento normalizadas y se llevará a cabo un análisis de los resultados con conjuntos de datos públicos. Dichas medidas estarán justificadas según el dominio del problema. Se tendrán en cuenta métricas basadas en la matriz de confusión (capacidad predictiva, especificidad, sensibilidad, precisión) y métricas resistentes al balanceo presente en los conjuntos de datos, como *F-measure* y AUC-ROC, especialmente apropiadas en aplicaciones bioinformáticas. Además, se usarán métodos y métricas para analizar las características de los conjuntos generados, como análisis de componentes principales, *Fréchet Inception Distance* o *Standard Distance Deviation*.

Por último, se aplicarán los modelos desarrollados con diferentes técnicas y algoritmos de aprendizaje computacional presentes en la literatura, con el objetivo de estudiar las diferencias en los resultados y comprobar si el sistema propuesto proporciona un mejor rendimiento para las medidas de evaluación y calidad establecidas.

UNIVERSIDAD DE MÁLAGA

1.4. Contribuciones de la Tesis

En este apartado se listan las principales aportaciones de esta tesis doctoral. Estas están dirigidas al diseño, estudio e implementación de nuevas técnicas de inteligencia computacional en el campo del aumento de datos para abordar problemas de bioinformática. Las contribuciones se pueden resumir de forma esquemática como sigue:

- Una revisión del estado del arte en DA y DL, especialmente centrado en métodos para tratar con conjuntos desbalanceados y en los modelos generativos profundos.
- Desarrollo de métodos de adición de ruido, especialmente un método con ajuste automático para las variables de entrada, y un método basado en el sobremuestreo.
- Diseño de un método adaptable a los modelos generativos perteneciente al DL, como VAEs y GANs, para su aplicación en DA en problemas de aprendizaje supervisado.
- Diseño de dos variaciones del método adaptable para añadir la funcionalidad de control del balanceo de los conjuntos generados.
- Diseño de un meta-clasificador para aprovechar el efecto del DA.
- Aplicación y validación empírica del rendimiento de los diferentes métodos, técnicas y modelos diseñados en problemas bien conocidos sin estructura espacial o temporal y un número reducido de muestras.
- Aplicación del DA en problemas de predicción de eventos en pacientes de cáncer a partir de muestras RNA-Seq.
- Aplicación del DA para mejorar el rendimiento en predicción de la enfermedad Niemann-Pick Tipo C a partir de muestras procedentes de un estudio metabolómico.
- Aplicación del DA en problemas de predicción del subtipo del cáncer de mama pertenecientes a pacientes de la provincia de Málaga a partir de muestras de expresión génica.

1.5. Estructura de la Tesis

El presente trabajo de tesis está conformado por cinco partes adicionales a esta introducción donde hemos planteado la problemática asociada al reducido número de muestras disponibles en problemas reales del ámbito de la bioinformática. A continuación, describimos de forma más detallada el contenido de las diferentes partes:

UNIVERSIDAD DE MÁLAGA

En los capítulos restantes de la presente Parte I, se presenta el estudio realizado sobre la base científica de esta investigación. En primer lugar, en el Capítulo 2 se realiza una extensa revisión del estado del arte de los métodos y técnicas pertenecientes al aumento de datos y al aprendizaje profundo. Adicionalmente, se detallan trabajos existentes que aplican modelos generativos profundos en tareas de aumento de datos, particularmente en problemas bioinformáticos (Capítulo 2). En segundo lugar, en el Capítulo 3 se describen los diferentes métodos y métricas utilizadas de forma recurrente en las investigaciones realizadas para alcanzar los objetivos planteados.

En la Parte II, se describen los métodos de aumento de datos desarrollados para la investigación y se muestran los resultados de su aplicación. En el Capítulo 4 se desarrollan los métodos basados en adición de ruido gaussiano utilizados, así como el esquema específico de ajuste del ruido desarrollado. Se presenta adicionalmente la aplicación de este último esquema en problemas bien conocidos. En el Capítulo 5 se expone el modelo de aumento de datos diseñado para ser adaptable al uso de modelos generativos profundos, y las funcionalidades añadidas para realizar un control sobre el balanceo del conjunto generado. También se muestra su aplicación a variaciones de los modelos VAE y GAN con conjuntos de datos de tamaño reducido y sin estructura espacial ni temporal.

En la Parte III, se aborda la aplicación de algunos de los modelos y métodos de aumento de datos desarrollados sobre tres problemas del campo de la bioinformática: la predicción de eventos en pacientes con cáncer mediante el análisis de muestras RNA-Seq (Capítulo 6); la predicción de la enfermedad rara Niemann-Pick Tipo C a partir de datos metabolómicos (Capítulo 7); y la clasificación del subtipo molecular del cáncer de mama utilizando la expresión génica del sistema inmune (Capítulo 8). En cada capítulo se incluye una introducción al problema, una breve descripción de los métodos aplicados y los resultados obtenidos.

En la Parte IV, conformada por el Capítulo 9, se presenta un esquema meta-clasificador diseñado para poder aprovechar los efectos observados tras realizar los análisis de las aplicaciones de aumento de datos descritos en los anteriores capítulos. También se realiza un breve repaso de otras metodologías existentes para tratar con conjuntos desbalanceados.

Finalmente, el último bloque está compuesto por el Capítulo 10, donde se exponen las conclusiones generales obtenidas tras la realización del trabajo presentado y se describen posibles líneas de trabajo futuro a partir del progreso alcanzado hasta el momento.

UNIVERSIDAD DE MÁLAGA

UNIVERSIDAD
DE MÁLAGA

Capítulo 2

Revisión del Estado del Arte

Resumen: Este capítulo presenta una revisión de los temas que se cubren en la presente tesis. En primer lugar, se realiza una revisión de técnicas y métodos clásicos de aumento de datos no relacionados con los modelos del aprendizaje profundo, tales como las transformaciones geométricas, los métodos de remuestreo y los basados en SMOTE. A continuación, se realiza un resumen de los modelos del aprendizaje profundo que han revolucionado el aprendizaje computacional y se explican los fundamentos del mismo. Para finalizar, se explican los modelos de aumento de datos más novedosos, que pertenecen al aprendizaje profundo, y se resumen los últimos trabajos sobre el uso de estos modelos generativos en bioinformática.

2.1. Aumento de Datos

El tema central de esta investigación es la aplicación de aumento de datos (DA) a problemas del dominio de la bioinformática. Inicialmente, el DA tenía el objetivo principal de reducir el sobreajuste sufrido por los modelos de aprendizaje computacional durante el entrenamiento. Esta no es la única técnica que se ha desarrollado para reducir el sobreajuste, y si nos centramos concretamente en el aprendizaje profundo (DL), otro de los temas de la tesis, nos encontramos con el método de dropout (Srivastava et al., 2014), el pre-entranamiento (Erhan et al., 2010) y, más recientemente, el transfer learning (Pan y Yang, 2010). Sin embargo, a diferencia de estas técnicas, el DA pretende reducir el sobreajuste desde la raíz del problema, el tamaño y composición del conjunto de datos de entrenamiento. Se aplican estos métodos bajo el supuesto de que los modelos de DL podrán extraer más información del conjunto de datos si tienen más instancias de ejemplo. Esto se encuentra apoyado en el hecho de que, por lo general, se necesita un gran conjunto de

UNIVERSIDAD DE MÁLAGA

datos con valores de miles de instancias de cada clase para lograr un buen nivel de éxito con estos modelos (Goodfellow et al., 2016).

A parte de esto, es necesario reseñar el contexto de seguridad de aplicación como una parte esencial del DA. La seguridad de un método de DA se refiere a su probabilidad de conservar la etiqueta de clase después de realizar un método de transformación o de generación profundo. Por ejemplo, las rotaciones e inversiones (*flipping*) son generalmente seguros en desafíos de ImageNet, por ejemplo, si se aplica a una tarea de reconocimiento entre gatos y perros. Sin embargo, no son seguras para otras tareas, como el reconocimiento de dígitos, 6 *versus* 9, o la extracción de información a partir de señales digitales. No existe ningún método de DA que no pueda dar como resultado una transformación de una instancia de forma que la etiqueta no lleve consigo alguna magnitud de distorsión. Esto demuestra que es necesario el diseño y desarrollo específico de métodos de DA que sean generalizables y que minimicen esta distorsión.

2.1.1. Métodos basados en transformaciones

Esta sección describe diferentes métodos de DA basados en transformaciones geométricas y otras funciones que habitualmente se utilizan en procesamiento de imágenes o lenguaje natural, si bien algunas de ellas pueden ser aplicadas a otros problemas. De este tipo de transformaciones simples, como la inversión horizontal de una imagen o el aumento del espacio de color, provienen las primeras demostraciones que indican la efectividad de la aplicación de DA. Los métodos basados en transformaciones se caracterizan por la modificación directa de las instancias originales, y normalmente presentan una implementación sencilla. La comprensión de estas transformaciones proporcionará una base útil para una mayor investigación sobre las técnicas de aumento de datos más complejas.

- **Inversión (*flipping*)**: Se trata de invertir la muestra, comúnmente una imagen, según su eje horizontal o su eje vertical. Este aumento es uno de los más fáciles de implementar, sin embargo, con conjuntos de datos sin información espacial no es aplicable. Aunque ha demostrado su utilidad en conjuntos de datos como CIFAR-10 e ImageNet, es importante reseñar que este tipo de transformaciones no es útil en ciertos problemas de reconocimiento de imágenes, pues no es una transformación que preserve la etiqueta de clase.

- **Rotación**: Este tipo de aumento se realiza girando la imagen hacia la derecha o la izquierda en un eje entre 1 y 359 grados. Como es lógico, esta técnica no es aplicable a problemas que traten conjuntos de datos sin información espacial. Al ser una versión amplia de la técnica de inversión, las etiquetas de clase pueden no ser válidas dependiendo del

UNIVERSIDAD DE MÁLAGA

problema, y la seguridad estará determinada en gran medida por el grado de rotación aplicado.

- **Desplazamiento (*shifting*)**: Esta técnica realiza un desplazamiento de los datos hacia la izquierda, derecha, arriba o abajo. Muy común en reconocimiento de imágenes, puede ser una transformación útil para evitar el sesgo introducido por el posicionamiento de los datos. Por ejemplo, en reconocimiento facial es útil comprobar que un modelo es capaz de realizar el reconocimiento con cierta independencia de si la cara se encuentra centrada o no. Esta técnica no es por lo general aplicable con conjuntos de datos sin información espacial ni temporal. Una cuestión importante es como se rellena el espacio restante tras aplicar el desplazamiento, y dependiendo de la aplicación puede realizarse utilizando un valor constante o con ruido aleatorio o gaussiano.

- **Recorte (*cropping*)**: Se trata de realizar un recorte, normalmente de una imagen, seleccionando una sección de la misma. Esto deriva en nuevas instancias con el centro focal desplazado respecto a la original, por lo que proporciona un efecto similar a la técnica de desplazamiento. La diferencia entre el desplazamiento y el recorte se da en que este último conlleva una reducción del tamaño de la entrada, por lo que es necesario un reescalado si se quiere aumentar el conjunto original. Aunque el enfoque más simple se produce en procesamiento de imágenes, se puede aplicar a otros tipos de problemas. Es posible que esta transformación no conserve la etiqueta de clase, al igual que ocurre con el desplazamiento.

- **Transformaciones del espacio de color**: Esta técnica está diseñada únicamente para realizar DA sobre el procesamiento de imágenes. Las imágenes actuales se codifican en 3 matrices que representan los valores de los píxeles para un color RGB individual. Esta técnica tiene múltiples posibilidades de aplicación según el tipo de variación deseada. Por ejemplo, se puede aplicar una transformación de luminosidad, oscureciendo o aclarando la imagen mediante el aumento o disminución de los valores de los píxeles en un valor constante. Otras transformaciones sencillas son variaciones para cambiar diferentes características como la tonalidad, el contraste, la saturación o la escala de colores, transformándolos en un negativo o en escala de grises. Aunque es una técnica simple, es necesario ser cuidadoso con su aplicación en ciertos problemas. Oscurecer o iluminar demasiado una imagen puede impedir reconocer un objeto y, aunque eliminar un sesgo de color favorece el reconocimiento de características espaciales, el color es una característica distintiva importante para algunas tareas.

UNIVERSIDAD DE MÁLAGA

Figura 2.1: Ejemplos de transformaciones geométricas y del espacio de color[1].

- **Adición de ruido**: este método consiste en añadir valores aleatorios, generalmente extraídos de una distribución gaussiana, a las características presentes en los datos originales. Los métodos de inyección de ruido han sido utilizados principalmente para mejorar la capacidad de generalización de la red neuronal (Matsuoka, 1992; Reed y Marks, 1998; Zur et al., 2009; Piotrowski y Napiorkowski, 2013). Con esto se intentaba evitar el sobreajuste de los modelos durante el entrenamiento, siendo un método habitualmente usado con este propósito. Sin embargo, no ha sido considerado habitualmente como un modelo propio de DA para crear nuevos datos sintéticos, salvo para preservar la confidencialidad de la información en los datos (Brand, 2002; Sarathy y Muralidhar, 2011). Una ventaja de estos métodos es el hecho de poder ser aplicado a prácticamente todos los problemas a tratar en aprendizaje computacional, independientemente de la estructura que presenten los datos. Una tarea importante es moderar el nivel de ruido aplicado para que la muestra transformada conserve la etiqueta de la muestra original.

- **Filtros de kernel**: Esta es una técnica popular en el procesamiento de imágenes para enfocar y desenfocar imágenes. Este método aplica una matriz desplazándola a través de una imagen con un filtro de desenfoque gaussiano para obtener una imagen más borrosa, o un filtro de borde vertical u horizontal de alto contraste para obtener una imagen más nítida a lo largo de los bordes. Debido a la similitud de los filtros de kernel con los mecanismos internos de las redes convolucionales, estos se utilizan normalmente como una capa interna de las redes, en lugar de como un método de DA.

[1] Extraído de Yu et al. (2020)

UNIVERSIDAD DE MÁLAGA

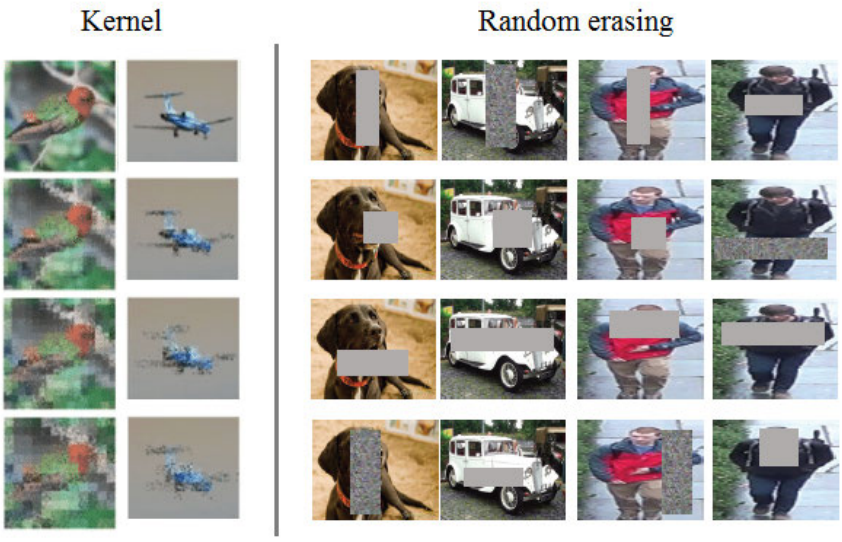

Figura 2.2: Ejemplos de filtros de kernel[2] (izquierda) y Random erasing[3] (derecha).

- **Borrado aleatorio (*random erasing*)**: Esta técnica desarrollada por Zhong et al. (2020) está inspirada por el método de regularización dropout, siendo aplicado directamente sobre la entrada en lugar de la arquitectura de la red. Esta técnica fue diseñada específicamente para mejorar los modelos de reconocimiento de imagen frente a la oclusión, es decir, cuando algunas secciones de la imagen no están claras. Este método obliga al modelo a aprender más características descriptivas sobre una imagen, evitando que se sobreajuste a una característica visual determinada en la imagen. Aparte de esta aplicación, su uso en otros conjuntos de datos es muy limitado.

- **Combinación de instancias**: En la aplicación de este método al procesamiento de imágenes o lenguaje natural, las instancias resultantes suelen parecer no poseer información útil para un observador humano. Cuando se aplica a otros ámbitos donde las instancias no son de por sí diferenciables para un observador externo, este método puede ser de utilidad. Obviamente es necesario ser cuidadoso en la elección de las instancias a combinar para que la instancia resultante de la combinación no se aleje especialmente del espacio original del conjunto de datos. Estos métodos son populares en la aplicación de algoritmos genéticos (Eiben et al., 1994; Tsoukalas y Uhrig, 1996), donde son usados en la etapa de reproducción. Hay diferentes variaciones de esta combinación, siendo las más habituales la combinación por sustitución a

[2]Extraído de Kang et al. (2017)
[3]Extraído de Zhong et al. (2020)

UNIVERSIDAD DE MÁLAGA

partir de un punto (*one-point*), sustitución por k-puntos, sustitución uniforme, o combinación por valores medios. Este método, al igual que la adición de ruido, puede ser aplicado a cualquier problema, independientemente de la estructura que presenten los datos. Un punto crítico de este método se encuentra a la hora de asignar una etiqueta de clase a la nueva instancia, cuando esta es resultado de la combinación de dos instancias de diferentes clases.

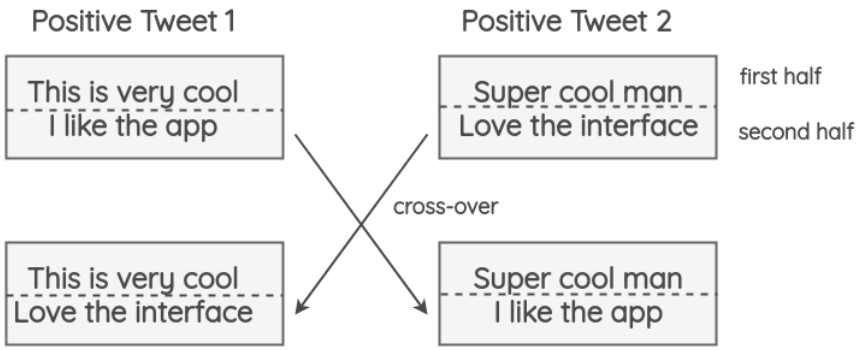

Figura 2.3: Ejemplo de combinación de instancias para procesamiento del lenguaje natural[4].

- ***Easy Data Augmentation Techniques* (EDA)**: Inspirados en los métodos de DA de visión por computador, Wei y Zou (2019) desarrollaron cinco operaciones diferentes para su uso en procesamiento del lenguaje natural: el reemplazo por sinónimos, el cual elije aleatoriamente palabras de la oración y las reemplaza con uno de sus sinónimos elegidos al azar; la inserción aleatoria, que busca un sinónimo aleatorio de una palabra aleatoria en la oración, y lo inserta en una posición aleatoria en la oración; el intercambio aleatorio, que elije aleatoriamente dos palabras en la oración e intercambia sus posiciones; y la eliminación aleatoria, la cual elimina aleatoriamente cada palabra en la oración con una probabilidad dada.

- **Substituciones por *Word-Embeddings* y máscaras**: Los *Word-Embeddings* son representaciones vectoriales de palabras o frases del lenguaje natural que han ganado popularidad en varias tareas semánticas. Los modelos de *Word-Embeddings*, como el *Word2Vec* son capaces de aprender diferentes representaciones de las palabras, como objetivo y contexto, e integrarlas en el mismo espacio. De esta forma, mediante de un algoritmo de vecinos más cercanos (*nearest neighbours*) se pueden escoger palabras que mejor encajen en la oración para realizar una sustitución (Melamud et al., 2015; Jiao et al., 2020). Algo similar se

[4]Extraído de Chaudhary (2020)

UNIVERSIDAD DE MÁLAGA

puede realizar mediante el uso de máscaras. Mediante el uso de modelos transformadores entrenados para predecir palabras enmascaradas según el contexto, se pueden crear de forma sencilla varias versiones de una misma oración (Garg y Ramakrishnan, 2020). Obviamente, la aplicación de estos métodos está también restringida al procesamiento del lenguaje natural. Normalmente, las instancias transformadas conservan la etiqueta de clase de la muestra original, debido a que en el proceso deben guardar el mismo significado.

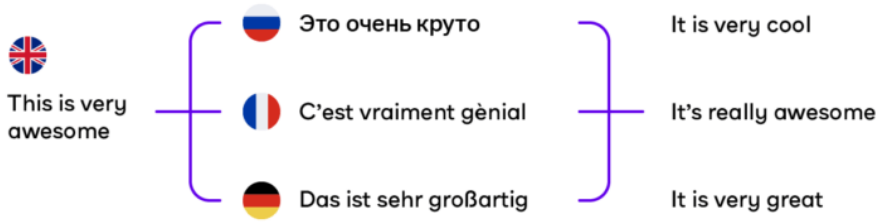

Figura 2.4: Ejemplo de aplicación sobre una oración del método de traducción inversa[5].

- **Traducción inversa (*Back-translation*)**: Este método implementa una idea muy sencilla, y que también es aplicable al procesamiento de imágenes con el modelo generativo profundo CycleGAN (Zhu et al., 2017), el cual será visto en una sección posterior. La traducción inversa (Sennrich et al., 2016; Xie et al., 2020) se refiere al procedimiento de traducir una oración existente a otro idioma diferente y luego volver a traducirlo al idioma original para obtener un ejemplo aumentado. Este método puede generar diversas frases conservando la semántica de las oraciones originales, por lo que es un método seguro.

2.1.2. Métodos de remuestreo

A parte de algunos de los métodos basados en transformaciones explicados en la sección anterior, uno de los primeros y más básicos métodos para aumentar el conjunto de datos de entrenamiento es el uso de remuestreo (*resampling*). El método *Jackknifing* (Quenouille, 1956; Tukey, 1958) es la primera aplicación de remuestreo en inferencia estadística para estimar el sesgo y la varianza de una estadística cuando se usa un conjunto aleatorio de observaciones para realizar los cálculos. La idea básica detrás del método *jackknife* radica en calcular de forma sistemática la estimación estadística, omitiendo una o más observaciones a la vez del conjunto de instancias, de forma que realiza diferentes subconjuntos del conjunto original. A partir de

[5]Extraído de Ilievski (2020)

UNIVERSIDAD DE MÁLAGA

estos nuevos subconjuntos se calcula una estimación del sesgo y una estimación de la varianza de la estadística.

Un método posterior de remuestreo es el llamado *bootstrap* (Efron, 1979, 1982). Esta es una técnica basada en la generación de conjuntos de datos de entrenamiento mediante la selección aleatoria uniforme de instancias y su duplicación con reemplazo. Esto permite incluir una instancia determinada en un subconjunto más de una vez. Este método tiene como mayor exponente el algoritmo de aprendizaje automático bagging, acrónimo de *"bootstrap aggregating"* (Breiman, 1996). La idea de este algoritmo es el remuestreo del conjunto de datos de entrada y la aplicación de clasificadores individuales sobre estos conjuntos remuestreados. Al integrar diferentes modelos de forma conjunta, se pretenden obtener mejores resultados gracias a que se mitigan los modelos con sesgo y con alta varianza. Para combinar los resultados de las distintas versiones, el método bagging utiliza el método del voto de pluralidad, también conocido como método del voto mayoritario simple, cuando el objetivo a predecir en el conjunto de datos es una clase; y el promedio cuando el objetivo a predecir es un resultado numérico.

En el Algoritmo 1 podemos ver el pseudocódigo del algoritmo de bagging. Como se puede observar, dado un conjunto de entrenamiento S de tamaño m, genera en cada iteración un nuevo subconjunto de instancias S_t a partir de él. Para ello, bagging hace uso de la técnica de *bootstrap*. Mediante el muestreo con reemplazo, algunas de las instancias se encontrarán repetidas en cada uno de los subconjuntos S_t. Si se tiene que $N = m$, entonces, para un valor grande de m, se prevé que los subconjuntos S_t presenten una fracción $1-1/e$ ($\approx 63{,}2\,\%$) de instancias únicas de S, el resto de instancias se encontrarán duplicadas.

Algorithm 1 Bootstrap aggregating

1: **procedure** BAGGING($S = \langle (x_1, y_1), ..., (x_m, y_m) \rangle$, $N\}$)
2: **for** $t = 1 \rightarrow T$ **do** ▷ loop over nº iterations
3: $S_t \Leftarrow$ initialize_Subset()
4: **for** $i = 1 \rightarrow N$ **do** ▷ loop over nº bootstrap samples
5: $sample_i \Leftarrow$ randomSel(S)
6: $S_t \Leftarrow$ add($sample_i$)
7: **end do**
8: $Classif_t \Leftarrow$ trainClassif(S_t)
9: $h_t \Leftarrow$ obtainHypothesis($X \rightarrow Y, Classif_t$)
10: **end do**
11: $H_f(x) = \arg \max_{y \in Y} \sum_{t=1}^{T} h_t(x)$
12: **return** H_f

El método bagging presenta una mejora más pronunciada si los clasifica-

UNIVERSIDAD DE MÁLAGA

dores sufren de sobreajuste o presentan inestabilidad, es decir, si cambiando ligeramente el conjunto de entrenamiento el clasificador individual presenta cambios significativos a la hora de predecir. Sin embargo, no funciona bien con algoritmos que presenten una baja varianza pero un alto sesgo. Aunque este algoritmo se puede aplicar con cualquier clasificador débil, denominados en la literatura inglesa como *weak learners*, son aplicados generalmente con árboles de decisión o de regresión, debido a que su uso es poco práctico cuando los modelos aplicados son complejos, y su entrenamiento y evaluación es costosa en tiempo de ejecución y memoria.

2.1.3. Métodos basados en SMOTE

Dentro del aumento de datos, hay algunas técnicas y métodos que se encargan de un subproblema concreto, aprender a partir de datos desequilibrados. El desequilibrio de clases en los conjuntos de datos es común en la aplicación del aprendizaje automático a muchos problemas del mundo real. La naturaleza de los problemas del mundo real a veces implica que las clases de datos no están representadas por igual, algunas clases tienen más instancias que otras, y esto introduce graves sesgos en el rendimiento de los algoritmos de aprendizaje. La mayoría de los algoritmos de aprendizaje automático en tareas de clasificación están centrados en mejorar el porcentaje de aciertos en predicción. Sin embargo, alcanzar una alta capacidad predictiva en un problema con un alto porcentaje de instancias mayoritarias es trivial. En un problema binario donde las clases están desbalanceadas 2:98 (2 instancias clase minoritaria, 98 mayoritaria), solo sería necesario que el clasificador asigne como predicción que todas las instancias pertenecen a la clase mayoritaria para que este obtuviera un porcentaje de aciertos del 98 %. Esto implica un riesgo potencial según el dominio de aplicación.

Los métodos de DA de sobremuestreo (*oversampling*) tratan de eliminar este tipo de problemas, procurando balancear las clases mediante el aumento del número de instancias de la clase minoritaria. El método más simple es la aplicación de remuestreo únicamente sobre el conjunto de instancias de la clase minoritaria. Sin embargo, la técnica de sobremuestreo predominante es la técnica SMOTE (*Synthetic Minority Oversampling Technique*) (Chawla et al., 2002). En lugar de aplicar una simple réplica de las instancias de clases minoritarias, la idea clave de SMOTE es la generación de ejemplos sintéticos. Este algoritmo utiliza la interpolación de varias instancias vecinas dentro de la clase minoritaria para la creación de nuevas instancias, razón por la que SMOTE está considerado como un método centrado en el espacio de características más que en el espacio de muestras, es decir, la generación se basa en los valores de las características y su relación, en lugar de considerar las instancias en conjunto como un todo. La Figura 2.5 muestra un ejemplo simple de la aplicación de este método de sobremuestreo. Se selecciona una instancia de clase minoritaria x como punto de referencia y, mediante un

UNIVERSIDAD DE MÁLAGA

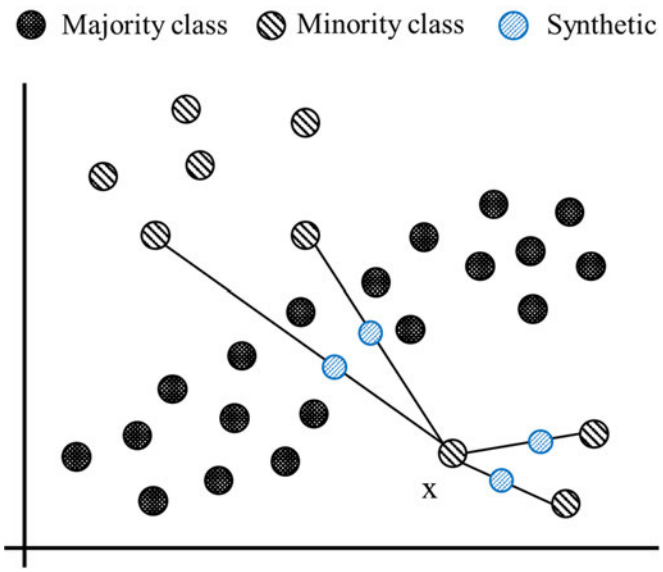

Figura 2.5: Generación de muestras sintéticas con la técnica de sobremuestreo SMOTE.

algoritmo de selección de vecinos más cercanos, se eligen otras instancias minoritarias. La creación de las nuevas instancias sintéticas se realiza mediante la interpolación aleatoria entre las instancias seleccionadas.

El procedimiento formal del método SMOTE se muestra en el Algoritmo 2, basado en el presente en Fernández et al. (2018). Teniendo las instancias originales de la clase minoritaria como entrada (*Samples*), en primer lugar, se configura la cantidad total de instancias que se han de crear a partir de cada instancia minoritaria seleccionada para que el conjunto aumentado esté balanceado (mismo número de instancias en ambas clases). Con este valor calculado, se realiza una iteración sobre el número de instancias T que pertenecen a la clase minoritaria, se selecciona aleatoriamente una de ellas y se obtienen los k vecinos más cercanos. Iterando sobre el número de instancias a crear a partir de $sample_i$, se elige aleatoriamente un vecino y se genera la instancia sintética mediante interpolación. De este modo, la interpolación calcula la diferencia entre la instancia $sample_i$ y el vecino en el espacio de características, multiplica la diferencia de cada característica por una normalización aleatoria entre 0 y 1, y luego agrega este valor al de la característica original de la instancia i. Se repite entonces el proceso, guardando las nuevas instancias sintéticas que estarán ubicadas dentro de este espacio entre los vecinos y $sample_i$.

Las desventajas de la aplicación del algoritmo SMOTE están relacionadas con la interpolación aleatoria que realiza y su diseño como técnica de sobremuestreo. Una de estas desventajas es la falta de control sobre la cantidad

UNIVERSIDAD DE MÁLAGA

Algorithm 2 Synthetic Minority Oversampling Technique

```
procedure SMOTE(Samples, N, T})
                    ▷ T nº minority class; N amount of oversampling
    if N < 100 then
        T ⇐ (N/100) * T
        N ⇐ 100
    end if
    N ⇐ (int) N/100
    for t = 1 → T do
        sample_t ⇐ randomSel(Samples)
        V[1...k] ⇐ kNearestNeigbours(Samples, sample_t)
        for i = N → 0 do
            sample_NN ⇐ randomSel(V)
            for attr = 1 → numAttr do          ▷ loop over attributes
                dif ⇐ sample_NN[attr] − sample_t[attr]
                gap ⇐ random(0, 1)
                synth_i[attr] ⇐ sample_t[attr] + gap · dif
            end do
            Synth ⇐ add(synth_i)
        end do
    end do
    return Synth
```

de instancias que se generarán a partir de los datos, ya que el sobremuestreo tiene como objetivo equilibrar completamente el conjunto de datos, es decir, solo puede generar instancias suficientes para esta circunstancia. Por lo tanto, es un método ineficaz en conjuntos de datos que presenten un buen equilibrio, para los cuales el número de instancias que se generarán para lograr el equilibrio total es bajo en proporción con el número absoluto de instancias involucradas. Otra desventaja derivada de la interpolación es la creación de instancias sintéticas que violan la geometría presente en el conjunto de datos para la clase minoritaria. Esto quiere decir que SMOTE puede crear instancias que se encuentren en el espacio de la clase mayoritaria, con lo que la etiqueta de clase asignada no sería correcta. En la Figura 2.5 también se muestra un ejemplo de esta desventaja.

Sobre el método SMOTE original se han realizado incontables estudios y se han desarrollado numerosas modificaciones sobre diferentes aspectos. Entre los aspectos que han sido modificados para desarrollar nuevos métodos destacan los siguientes (Fernández et al., 2018).

- Filtrado de instancias generadas: Las primeras extensiones de SMOTE estaban motivadas por la desventaja en la generación de instancias

UNIVERSIDAD DE MÁLAGA

superpuestas y que son consideradas ruidosas. De este modo, las variaciones añadían un proceso de filtrado de ruido justo después de finalizar la generación de datos artificiales mediante el procedimiento SMOTE.

- Selección inicial de instancias a sobremuestrear: Se trata de determinar conjuntos con los mejores candidatos para sobremuestrear antes de iniciar el proceso de generación de ejemplos sintéticos, con el objetivo de reducir la superposición y el ruido en el conjunto de datos final. Las variaciones habituales se basan en elegir las instancias límite de las clases, y en no generar una instancia dependiendo del número de ellas que pertenecen al vecindario.

- Integración con *undersampling*: Las instancias que pertenecen a la clase mayoritaria se eliminan mediante el uso de una técnica de *undersampling* antes o después de realizar el sobremuestreo.

- Operación de interpolación: Las modificaciones sobre este aspecto son más variadas, ya que afecta de forma directa a la creación de las instancias. Así, se puede realizar la interpolación no solo con los vecinos más cercanos de la clase minoritaria sino también de la clase mayoritaria; se puede usar una ponderación de características para hallar los nuevos valores; se pueden realizar interpolaciones múltiples involucrando más de dos ejemplos o siguiendo topologías basadas en formas geométricas; se puede realizar la interpolación solo con instancias que pertenecen a un mismo clúster; y es posible interpolar utilizando diferentes distribuciones aleatorias, como la gaussiana o estimaciones de la propia distribución de probabilidad de los datos.

- Operación con cambios de dimensionalidad: Esta modificación implica la integración de SMOTE con una técnica que aumente o reduzca la dimensionalidad de los datos antes o durante la generación. El enfoque más común es reducir la dimensionalidad al principio ya sea a través de métodos de componentes principales o selección de características, y luego seguir con el procedimiento en el nuevo espacio dimensional.

- Generación adaptativa: La idea es utilizar una distribución ponderada en función de cada instancia de la clase minoritaria según el grado de dificultad para ser aprendidas por el algoritmo de aprendizaje computacional utilizado. De esta manera, se pretenden generar más datos sintéticos partiendo como referencia de instancias que sea más complicadas de aprender en comparación con otras.

- Reetiquetado: Son modificaciones que realizan un etiquetado de las nuevas instancias generadas para comprobar si estas pudieran pertenecer a la clase mayoritaria, resolviendo el aspecto de seguridad de etiquetado de clase de las instancias.

UNIVERSIDAD DE MÁLAGA

Respecto a métodos que presentan variaciones concretas sobre SMOTE, se presenta el método Borderline-SMOTE (Han et al., 2005). Con SMOTE se hace una selección aleatoria de las instancias y sus vecinos, lo que hace que las instancias generadas puedan no otorgar información por estar concentrados en núcleos donde ya se encuentran otras instancias sintéticas. El método Borderline-SMOTE se basa en esta premisa de que los ejemplos alejados del límite contribuyen poco al aprendizaje en clasificación. Para evitar la creación de este tipo de instancias, el método identifica primero las instancias que pertenecen al límite entre clases, teniendo en cuenta para ello las clases de los vecinos más cercanos de cada instancia que se va a tomar como referencia para el sobremuestreo. Si todos los vecinos pertenecen a la clase mayoritaria, la muestra se clasifica como "ruido" y no es considerada. Si el número de vecinos que pertenecen a la clase minoritaria es mayor que los pertenecientes a la mayoritaria, tampoco se tiene en cuenta, pues es considerada una instancia "segura". De este modo, solamente se considera la estancia si el número de vecinos de la clase mayoritaria es mayor que el 50 % pero menor al 100 %, es decir, se encuentra en el límite. Estas instancias denominadas "peligrosas" son las utilizadas para la interpolación con los vecinos de la clase minoritaria. Un ejemplo se muestra en la Figura 2.6, con un conjunto de datos sintético en forma de círculo. En esta figura se muestra en a) los datos originales; b) las instancias de la clase minoritaria clasificadas como "peligrosas" (cuadrados sólidos); y c) las instancias generadas mediante Borderline-SMOTE (cuadrados huecos).

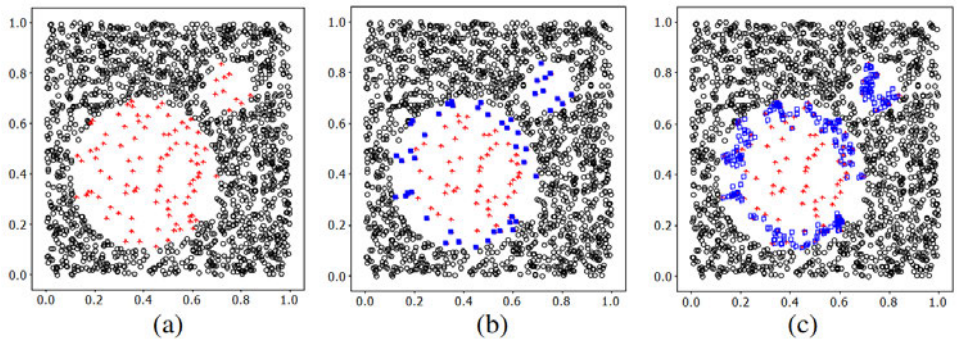

Figura 2.6: Generación con Borderline-SMOTE[6]: a) sobre el conjunto sintético Circle; b) instancias "peligrosas"; c) instancias generadas.

Otro método basado en SMOTE y desarrollado con un propósito similar al anterior es ADASYN (He et al., 2008). Este método se basa en la idea de generar de forma adaptativa instancias del conjunto de datos minoritario de acuerdo a como se encuentran distribuidas. Concretamente, el enfoque de ADASYN implica generar muestras sintéticas de forma inversamente propor-

[6]Extraído de Han et al. (2005)

UNIVERSIDAD DE MÁLAGA

cional a la densidad de los ejemplos en la clase minoritaria. Es decir, generar más muestras sintéticas en regiones del espacio de características donde la densidad de instancias minoritarias es baja, y menos o ninguna donde la densidad de instancias es alta. Ésta es una diferencia importante en comparación con la generación aleatoria de SMOTE, y con el modelo discriminativo creado mediante Borderline-SMOTE. Para ello, en ADASYN se modifica el algoritmo SMOTE de forma que primero se calculan los vecinos más cercanos para todas las muestras sintéticas, definiendo una ratio dependiente del número de instancias mayoritarias pertenecientes a los vecindarios. Con esa ratio se calcula una función de distribución y el número de instancias a generar para cada muestra seleccionada del conjunto minoritario. El resto del procedimiento es el mismo que en SMOTE. En ADASYN, el conjunto de datos resultante no solo proporciona una representación equilibrada de la distribución de datos, sino que también pretende forzar al algoritmo de aprendizaje a centrarse en los ejemplos difíciles de aprender. Sin embargo, este método presenta una acuciante desventaja. Es habitual que ADASYN no genere ninguna instancia, o genere un número muy reducido de ellas, debido a la configuración original del conjunto de datos.

2.2. Aprendizaje profundo

Las redes neuronales artificiales, conocidas por su anglicismo *artificial neural networks* (ANNs), se presentan generalmente como sistemas que agrupan elementos o nodos de procesamiento (neuronas) altamente interconectados a través de pesos sinápticos, que intercambian información entre sí para resolver problemas específicos. Cada una de las conexiones, las cuales tienen asignadas un peso sináptico, se ajustan durante el proceso en base a la experiencia según los resultados obtenidos basándose en las funciones de activación de las neuronas, haciendo que las redes neuronales sean capaces de aprender. Este es el principio básico de todas las redes neuronales. La diferencia primordial entre estas redes neuronales artificiales y las redes consideradas en aprendizaje profundo, se encuentra en la llamada profundidad de la red, es decir, el número de capas ocultas y el número de neuronas en cada una, que presenta el modelo de la red neuronal.

Si la profundidad de la red es 0 se tiene el modelo denominado perceptrón simple (Rosenblatt, 1958), el cual presenta una escasa capacidad de aprendizaje, ya que es un modelo que solo es capaz de aprender funciones lineales. Para ser capaces de aproximar o interpolar problemas que no son linealmente separables es necesario añadir profundidad a la red, creando el modelo denominado *Feedforward Neural Network* (FNN) o perceptrón multicapa. Con la incorporación de una capa intermedia de neuronas se otorga al modelo la capacidad de funcionar como un aproximador universal (Hornik et al., 1989), es decir, mediante su uso se puede aproximar cualquier función continua en

UNIVERSIDAD DE MÁLAGA

un espacio R^N. En una FNN se tienen tres tipos de capas neuronales: una capa de entrada, donde las neuronas se encargan de recibir la información de entrada y propagarla al resto de la red; la capa oculta, donde las neuronas realizan un procesamiento de los datos recibidos a través de la entrada, y transmiten el resultado que han obtenido; y la capa de salida, que transmite al exterior la respuesta de la red para cada patrón de entrada.

Para que una FNN pase a ser considerada un modelo de DL en lugar de un perceptrón multicapa, esta ha de presentar una profundidad mayor o igual de 3, es decir, la red ha de tener al menos 3 capas ocultas de neuronas. En la Figura 2.7 se pueden observar dos redes *feedforward* que ejemplarizan la diferencia existente entre modelos para su consideración de "profunda".

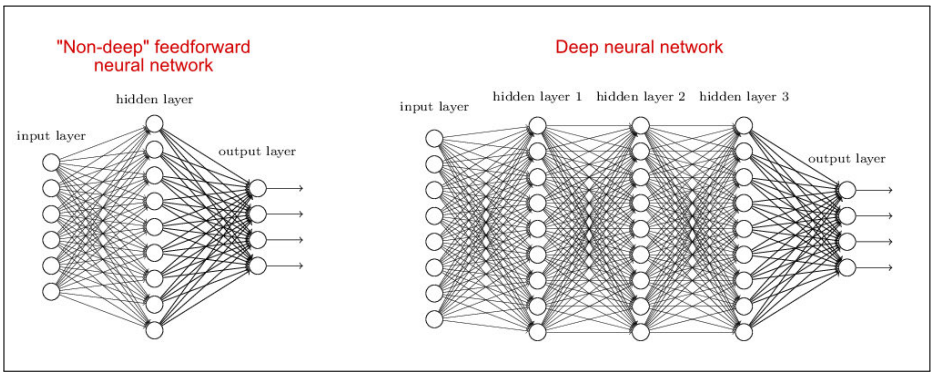

Figura 2.7: Redes Neuronales *Feedforward* profunda y "no-profunda"[7].

2.2.1. Redes neuronales feedforward profundas

Las primeras redes neuronales artificiales comenzaron a desarrollarse a partir de la década de 1940, cuando McCulloch y Pitts (1943) crearon un modelo primitivo que todavía era incapaz de aprender, y es unos años más tarde cuando se originan las primeras ideas de su uso en aprendizaje (Hebb, 1949). En las siguientes décadas se hicieron más progresos en este ámbito de investigación, se desarrolló el perceptrón (Rosenblatt, 1958), se avanzó en su aplicación en aprendizaje supervisado (Rosenblatt, 1961; Narendra y Thathachar, 1974), no supervisado (Kohonen, 1972; Von der Malsburg, 1973; Willshaw y von der Malsburg, 1976), y otras aplicaciones (Hopfield, 1982).

Es en la década de 1960, cuando se crean las primeras redes neuronales que pueden ser consideradas como el primer sistema de DL. Estas se basaban en una familia de algoritmos inductivos denominados *Group Method of Data Handling* (GMDH) (Ivakhnenko, 1968, 1971), las cuales han sido usadas en

[7]Extraído de Nielsen (2015)

UNIVERSIDAD DE MÁLAGA

distintas aplicaciones. Sin embargo, un algoritmo revolucionó la investigación en redes neuronales aplicadas al aprendizaje supervisado, el algoritmo llamado *backpropagation*, o retropropagación del error. La primera aplicación específica de este algoritmo en redes neuronales proviene del trabajo de Werbos (1974), si bien se popularizó su uso gracias a la contribución de Rumelhart et al. (1986). *Backpropagation* es comúnmente confundido con todo el algoritmo de aprendizaje de la red, cuando únicamente se refiere al método del cálculo de la propagación del gradiente, el cual es utilizado posteriormente por otros métodos para realizar el aprendizaje en la red.

La teoría básica de este algoritmo en su aplicación en la FNN es la repetición cíclica de tres fases (Hecht-Nielsen, 1989): la propagación de la entrada en la red (*feedforward*); la retropropagación del error por las capas; y la actualización de los pesos. De esta forma, ante un vector de entrada, el algoritmo propaga este por las neuronas de la red mediante el procesamiento por las funciones de activación de las neuronas de las capas ocultas hasta llegar a la capa de salida, momento en el cual se realiza el cálculo del error de la salida obtenida respecto a la salida deseada. Este cálculo proporciona un valor de error respecto a la entrada, valor que es retropropagado por la red. Cada neurona es responsable en cierta medida de este error, por lo que durante la retropropagación cada neurona tomará asociado un valor de error distinto en consideración con su contribución en la salida errónea. El valor de error es utilizado una vez que se ha propagado por toda la red, para calcular el descenso del gradiente necesario en la neurona. A partir de aquí, otro algoritmo, como puede ser el descenso del gradiente estocástico, es el encargado de actualizar los pesos sinápticos conectados a la neurona con el fin de minimizar el error en la siguiente iteración. Este proceso continúa hasta que se alcanza un mínimo de rendimiento impuesto, es decir, el error cometido en la salida es suficientemente pequeño, o bien se haya impuesto un máximo de iteraciones en la red y este haya sido alcanzado. Una vez completado el proceso, los pesos de las capas ocultas han sido estructurados de forma que las neuronas son capaces de reconocer distintas propiedades del conjunto de entrada, porque el modelo ha sido capaz de aprender.

Como se ha mencionado, el proceso anteriormente descrito pertenece al algoritmo de *backpropagation* básico. Las implementaciones más modernas de software que llevan a cabo este algoritmo, como Caffe (Jia et al., 2014) o TensorFlow (Abadi et al., 2016), se basan en la forma generalizada de *backpropagation*. Mediante este algoritmo se puede ajustar cualquier grafo computacional a través de la manipulación de una estructura de datos, con el objetivo de que escenifique una representación simbólica, una representación en símbolos o variables que no tienen valores específicos hasta que se le asignan al entrenar la red.

Con respecto al rendimiento que presentan la FNN profunda con la implementación del algoritmo *backpropagation*, este es aceptable cuando el número

UNIVERSIDAD DE MÁLAGA

de capas ocultas presentes en la implementación concreta del modelo es menor de 5. Si se aumenta el número de capas ocultas, sucede que durante la retropropagación las primeras capas ocultas son consideradas como carentes de responsabilidad en el error cometido en la salida de la red, es decir, el gradiente propagado a estas capas es nulo con lo que sus pesos sinápticos no son actualizados, quedando de esta forma limitada la capacidad de mejora del error cometido en la red, y por tanto, la posibilidad de mejorar su rendimiento.

2.2.2. Redes neuronales convolucionales

A parte de las redes basadas en GMDH, una de las primeras redes neuronales meritoria de considerarse modelo de DL es el Neocognitron (Fukushima, 1980), modelo que es además la primera red basada en los principios neurofisológicos del funcionamiento del córtex visual en la mayoría de mamíferos (Hubel y Wiesel, 1959, 1962, 1968). Asimismo, el Neocognitron incorporaba parte de los elementos que le corresponden a las denominadas *Convolutional Neural Networks* (CNN). La CNN está especializada en el procesamiento de datos con estructura espacial, que presentan una tipología de rejilla o *grid*, tales como conjuntos de imágenes, los cuales son píxeles en una rejilla de 2 dimensiones. Su aplicación en otros conjuntos de datos con diferente tipología presenta un rendimiento limitado.

El nombre de convolucional que presentan estas redes viene dado por el uso de las mismas de la operación denominada convolución. La convolución (2.1) involucra dos funciones f y g de forma que su salida representa la magnitud en la que se superponen la función f y la función g una vez esta última ha sigo invertida y trasladada una distancia t.

$$s(t) = (f * g)(t) = \int f(x)g(t-x)dx \tag{2.1}$$

En términos de una red convolucional, se conoce al primer argumento (función f) como entrada y al segundo argumento (función g) como kernel. Normalmente en la práctica ambos argumentos suelen ser matrices dimensionales denominados tensores, la entrada es una matriz de datos y el kernel una matriz de parámetros que han de ser adaptados por la red, y además suele ser aplicada sobre más de un eje a la vez. En el caso de que se quiera utilizar como entrada una imagen bidimensional I, probablemente también se utilice un núcleo bidimensional K. Teniendo en cuenta que la convolución es conmutativa, se obtiene una operación de convolución (2.2) más sencilla de implementar debido a que hay menos variación en el rango de valores válidos de m y n.

$$S(i,j) = (K * I)(i,j) = \sum_m \sum_n I(i-m, j-n)K(m,n) \tag{2.2}$$

UNIVERSIDAD DE MÁLAGA

La CNN explota además tres ideas que les ayudan a mejorar el rendimiento de la red: limitación de las interacciones, compartición de parámetros, y uso de neuronas con volumen 2D/3D. La limitación de las interacciones dentro de la red se puede lograr de dos formas, reduciendo el kernel para que este sea más pequeño que la entrada, centrándose en detectar características reducidas y significativas, con lo que es necesario almacenar menos parámetros y menos operaciones; o limitando el número de conexiones de cada neurona con la capa siguiente, de forma que se tiene una red localmente conectada. Respecto a compartir parámetros, se trata de la compartición de pesos en distintas partes de la red convolucional, de forma que están ligados por la suposición de que, si estos pesos son computacionalmente importantes en una parte de la red, entonces deben ser útiles en otra parte de la misma. Esto ahorra nuevamente espacio en memoria. Bien es cierto que algunas veces esta presunción puede ser peligrosa, por ejemplo, si se tratan de encontrar características muy diferentes en el mismo conjunto de datos, en cuyo caso se puede relajar la compartición de pesos. Por último, y ligado también a la idea de limitar las conexiones, están las neuronas de volumen 2D/3D, una arquitectura que se aprovecha en aplicaciones de reconocimiento de imágenes, y donde las neuronas en una capa están conectadas solamente a una pequeña región de la capa anterior.

Otro concepto importante de la CNN es el *pooling*. Una función de *pooling* trabaja de forma que sustituye la salida de la red por una estadística de resumen de las salidas cercanas. La función de *pooling* más común es la denominada *max pooling* (Zhou y Chellappa, 1988). Esta función divide la entrada en un conjunto de rectángulos no superpuestos y, para cada una de dichas subregiones, emite como salida el máximo de la subregión. La función de una capa de *pooling* es pues la reducción progresiva del número de parámetros que se tienen en cuenta en la red, reduciendo a su vez el cómputo necesario en la misma. Esto provoca además que la red posea la cualidad de invarianza a la traducción. Esta propiedad hace referencia a que, ante pequeños cambios en la entrada, la salida obtenida por la red se mantiene aproximadamente inmutable. Además de la operación de *max pooling*, hay otras funciones que pueden ser utilizadas, como la *average pooling*, que en vez de sustituir por el máximo lo hace por la media, o la *L2-norm pooling*. La función *average pooling* ha sido utilizada frecuentemente en la práctica como alternativa a *max pooling*, pero un estudio demuestra el mejor rendimiento de este último (Scherer et al., 2010).

En las redes convolucionales modernas, la arquitectura final que presentan está compuesta por dos bloques. En el primer bloque normalmente se repite la secuencia de capas [Convolucional, ReLU, Pooling] un número indeterminado de veces. La primera capa produce varias convoluciones en paralelo, también denominadas filtros, para producir un conjunto de activaciones lineales; en la segunda, denominada etapa detectora, cada activación

UNIVERSIDAD DE MÁLAGA

lineal se ejecuta a través de una función de activación no lineal como ReLU; y en la tercera se utiliza una función de *pooling*, para modificar la salida de la capa. Finalmente, en el segundo bloque se tiene comúnmente una FNN, varias capas totalmente conectadas de neuronas, para realizar el razonamiento de alto nivel y obtener una salida. En la Figura 2.8 se observa un ejemplo de arquitectura para reconocimiento de objetos.

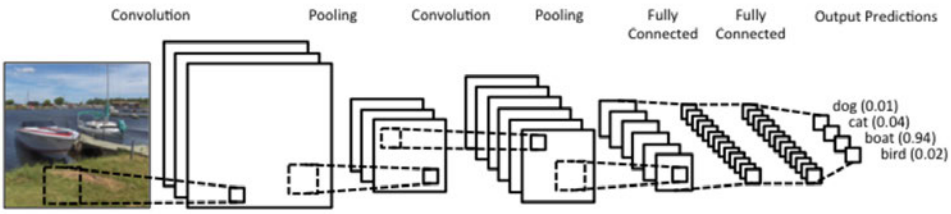

Figura 2.8: Red Neuronal Convolucional[8].

La primera CNN moderna se considera LeNet-5, creada por LeCun et al. (1998), quien en un estudio anterior (LeCun et al., 1989) ya diseñó una red convolucional basada en la aplicación de la convolución, la compartición de pesos y la *backpropagation*, en una red similar al Neocognitron de Fukushima. Además, en este primer trabajo introdujo uno de los conjuntos de datos que con el tiempo se ha convertido quizás en el referente más famoso del aprendizaje automático, el conjunto MNIST, una gran base de datos de dígitos escritos a mano que se utiliza comúnmente para el entrenamiento de sistemas de procesamiento de imágenes. El modelo LeNet-5 en su estudio original se aplicaba en el conjunto MNIST (LeCun et al., 1989), y tal era su rendimiento que a finales de la década de 1990 un sistema que implementaba LeNet-5 leía el 10 % de los cheques que se emitían en USA.

2.2.3. Redes neuronales recurrentes

Las *Recurrent Neural Networks* (RNN) (Rumelhart et al., 1986) son una familia de redes especializadas en el procesamiento de secuencias de valores $x(1), ..., x(t)$. De igual forma que las redes convolucionales escalan fácilmente extensos patrones 2D/3D, las redes recurrentes pueden escalar secuencias temporales mucho más largas que otras redes sin especialización secuencial. Esta arquitectura de DL se basa en las ideas de la compartición de parámetros, tal y como se procede en las redes convolucionales, y en el *unrolling* de la red (Figura 2.9). El *unrolling* es un procedimiento para realizar un mapeo de tiempo a espacio, es decir, es un proceso que reemplaza una red recurrente (especializada en secuencias de tiempo) por una red *feedforward* mucho

[8]Extraído de Clarifi/Technology (2017)

UNIVERSIDAD DE MÁLAGA

mayor con coeficientes repetidos. Esta técnica es necesaria para aplicar el algoritmo de *backpropagation*. Básicamente, el proceso toma las entradas, salidas y unidades ocultas de la RNN y las replica para cada paso de tiempo, de forma que estas replicaciones correspondan a capas en la nueva FNN, en la que se puede aplicar fácilmente *backpropagation*.

Con estos conceptos, se puede desarrollar una RNN siguiendo distintos patrones de diseño. Los diseños de redes recurrentes más característicos son: redes que producen una salida en cada paso de tiempo y tienen conexiones recurrentes entre las neuronas ocultas; redes que producen una salida en cada paso de tiempo y sólo tienen conexiones recurrentes desde la salida en un paso de tiempo a las neuronas ocultas en el siguiente paso de tiempo; y redes con conexiones recurrentes entre neuronas ocultas que leen una secuencia completa para producir una sola salida.

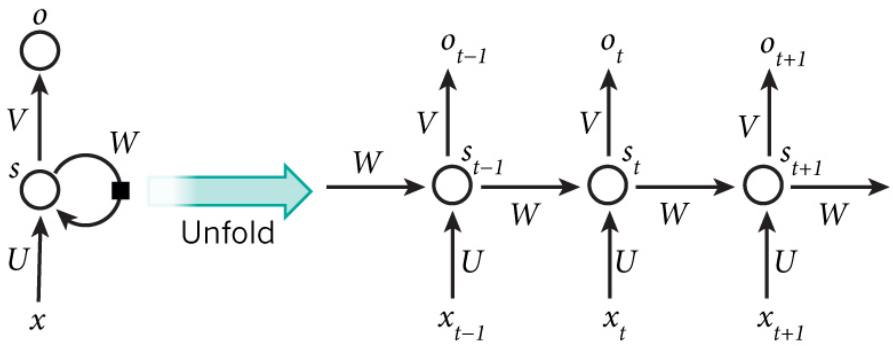

Figura 2.9: Una red neuronal recurrente y el desarrollo en el tiempo del proceso de *unfolding*[9].

En las versiones "básicas" de las redes recurrentes no es posible generar una predicción de $y(t)$ que dependa de toda la secuencia de entrada. Para comprender mejor esta limitación, se puede suponer el ejemplo de su aplicación para el reconocimiento de voz. Para poder realizar el reconocimiento de voz hay que tener en cuenta que la interpretación correcta del sonido actual como fonema puede depender de los próximos fonemas, e incluso depender de las siguientes palabras debido a las dependencias lingüísticas entre las palabras cercanas, por lo que es posible que sea necesario mirar lejos en el futuro (o el pasado) para evitar las ambigüedades del lenguaje. Con este propósito fueron diseñadas las redes neuronales recurrentes bidireccionales (Schuster y Paliwal, 1997). Tal y como sugiere su nombre, el modelado de estas redes combina dos RNNs, una de ellas desplazándose hacia adelante comenzando desde el inicio de la entrada, con otra RNN que se desplaza a la vez hacia atrás comenzando desde el final de la entrada. Esto permite a las neuronas de

[9]Extraído de LeCun et al. (2015)

UNIVERSIDAD DE MÁLAGA

salida computar una representación que dependa tanto del pasado como del futuro, pero siendo precavido y teniendo en cuenta al mismo tiempo que la representación $y(t)$ debe ser más sensible a los valores de entrada alrededor del tiempo t, es decir, presenta un cambio mayor por valores más cercanos al tiempo actual.

Las RNN bidireccionales obtienen un gran rendimiento en las aplicaciones en las que surge la necesidad de que la salida dependa de toda la secuencia de entrada, ya que están diseñadas con tal propósito, tales como son el reconocimiento de textos (Graves y Schmidhuber, 2008) y el reconocimiento de voz (Graves y Schmidhuber, 2005; Graves et al., 2013). También se pueden aplicar a entradas bidimensionales como imágenes, de forma que son necesarias cuatro RNNs para cada una de las cuatro direcciones, si bien se obtiene un modelo computacionalmente más costoso en comparación con una CNN.

Aun teniendo en cuenta que las RNN bidireccionales son capaces de obtener un buen rendimiento, actualmente los modelos más efectivos pertenecientes a las redes recurrentes son las llamadas *gated* RNNs. Entre estos modelos se encuentra incluido la red *Long Short-Term Memory* (LSTM) (Hochreiter y Schmidhuber, 1997). La idea principal de la LSTM es la introducción de bucles para producir caminos donde el gradiente pueda fluir durante largos periodos de tiempo, pero fue mejorado gracias al planteamiento de Gers et al. (2000), haciendo que el peso en este bucle se encuentre condicionado al contexto, en lugar de ser fijo. En lugar de unidades que simplemente aplican una no linealidad sencilla, las redes LSTM tienen "células LSTM" que presentan una recurrencia interna adicional a la recurrencia externa de la RNN. Estas células tienen las mismas entradas y salidas que cualquier otra RNN, sin embargo, presentan más parámetros de control y un sistema de unidades de control que dirigen el flujo de información. En la literatura ha sido demostrado que las redes LSTM aprenden dependencias a largo plazo de una forma más eficiente que las arquitecturas recurrentes simples, tanto en conjuntos de datos artificiales (Hochreiter et al., 2001), como en la traducción automática (Sutskever et al., 2014), o la generación de escritura a mano (Graves, 2013).

Otro modelo que se incluye dentro de este grupo de *gated* RNN, son las *Gated Recurrent Units* (GRU) (Chung et al., 2015), cuya principal diferencia con la red LSTM es que una única unidad de control controla simultáneamente el factor de olvido y la decisión de actualizar la unidad de estado. Las unidades encargadas de reiniciar y actualizar pueden ignorar por separado partes del vector de estado, de forma que las puertas de actualización pueden accionar linealmente cualquier dimensión, eligiendo copiarla (en un extremo del sigmoide) o ignorándola completamente (en el otro extremo) reemplazándola por el nuevo valor de estado objetivo hacia el cual se desea converger; y las puertas de reinicio pueden controlar qué partes del estado se utilizan para calcular el siguiente estado objetivo, introduciendo un efecto

UNIVERSIDAD DE MÁLAGA

no lineal adicional en la relación entre estado pasado y estado futuro. Sobre este grupo de modelos es posible diseñar muchas más variantes, por ejemplo, se puede hacer que la salida obtenida por una unidad sea compartida entre múltiples unidades ocultas, sin embargo, diversas investigaciones sobre las variaciones arquitectónicas de las redes LSTM y GRU no encontraron ninguna variante que pudiera superar claramente a ambas en una amplia gama de tareas (Jozefowicz et al., 2015).

2.2.4. Función de activación

De forma independiente a la implementación concreta de *backpropagation* que se emplee y puesto que en cada iteración del proceso se requiere del cálculo del gradiente según la función de error, es decir, según el error cometido por la red, es prioritario asegurar la continuidad y diferenciabilidad de esta función de error. Por esta razón, se necesita que la función de activación empleada por las neuronas sea diferenciable, quedando patente que las neuronas deben utilizar funciones de activación que no sean la función escalonada, usada en los perceptrones simples. La función compuesta producida por perceptrones interconectados en la red que presentan esta función escalonada, es discontinua y por lo tanto la función de error también. Hay diferentes funciones de activación que sí cumplen la propiedad de diferenciabilidad, entre las cuales las funciones de activación sigmoide, la tangente hiperbólica (tanh) y la softmax han sido las más habituales.

La función de activación sigmoide (2.3) es un caso especial de la función logística, nombrada así por Verhulst (1845). La imagen de la función sigmoide se encuentra en el rango (0,1), de forma que la función satura a 0 cuando x se hace muy negativa y satura a 1 cuando se hace muy positiva, lo que significa que la función se vuelve insensible a pequeños cambios en su entrada.

$$\sigma(x) = \frac{1}{1 + e^{-x}} \tag{2.3}$$

La función tangente hiperbólica (2.4) tiene una imagen en el rango (-1,1). Esta función de activación está íntimamente relacionada con la función sigmoide (2.5). Si bien, en la comparativa entre las funciones sigmoide y tanh, hay que tener en cuenta que esta última es más parecida a la función identidad ($i(x) = x$), ya que $tanh(0) = 0$ mientras que $\sigma(0) = \frac{1}{2}$.

$$\tanh(x) = \frac{\sinh(x)}{\cosh(x)} = \frac{e^x - e^{-x}}{e^x + e^{-x}} = \frac{e^{2x} - 1}{e^{2x} + 1} \tag{2.4}$$

$$\tanh(x) = 2\ \sigma(2x) - 1 \tag{2.5}$$

Es necesario tener en cuenta que a los valores negativos se le asignarán salidas negativas en la función tanh, y sólo a valores cercanos a cero se les

UNIVERSIDAD DE MÁLAGA

asignan salidas cercanas a cero. Esta propiedad hace que sea menos probable que la red no actualice los pesos durante el entrenamiento. Siempre y cuando las activaciones de la red puedan mantenerse pequeñas, la formación de la red con la función tanh es más sencilla. Por esta razón, si es necesario utilizar la misma función de activación para todas las neuronas de la red, es recomendable utilizar tanh frente a sigmoide. Si solo se empleara en la salida de la red, dependería de las características del problema.

La función de activación softmax (2.6) es una generalización de la función sigmoide. Esta toma un vector de entrada x de valores arbitrarios y lo transforma en otro con valores reales en el rango (0,1). Al ser una generalización de la función sigmoide, softmax al igual que esta puede saturar. En el caso de la función softmax, hay valores de salida que pueden saturarse cuando las diferencias entre los valores de entrada se vuelven extremas.

$$\text{softmax}(x)_i = \frac{e^{x_i}}{\Sigma_j \; e^{x_j}} \tag{2.6}$$

Como ya ha sido mencionado, las funciones de activación anteriores han sido las más utilizadas en la práctica, en especial la función sigmoide. Sin embargo, en la última década la tendencia a cambiado, y la función de activación recomendada actualmente para su uso en redes neuronales profundas es la *Rectified Linear Unit*, ReLU (2.7). La primera publicación que introdujo esta función en redes dinámicas fue realizada por Hahnloser et al. (2000), si bien su uso se extendió tras varias investigaciones que ratificaban que su empleo en las distintas arquitecturas de DL mejoraban el rendimiento (Jarrett et al., 2009; Nair y Hinton, 2010; Glorot et al., 2011).

$$\text{ReLU}(x) = \max\{0, x\} \tag{2.7}$$

Esta función de activación guarda una semejanza clara con la función lineal, lo que hace que esta sea fácil de optimizar. La imagen de la función de activación ReLU se encuentra claramente en el rango (0, x), por lo que únicamente se diferencia de la función lineal en que para la mitad de su dominio el valor del gradiente es 0. Además, el hecho de que se tenga un gradiente de valor elevado y consistente siempre que la unidad este activa, garantiza un entrenamiento más rápido en comparación con otras funciones no lineales. Sin embargo, el hecho de que para la mitad del dominio el gradiente sea 0 da lugar al problema llamado "*Dying ReLU*", donde algunas neuronas con ReLU esencialmente mueren para todas las entradas y permanecen inactivas sin importar la entrada que se suministre. Esto implica que durante el proceso de *backpropagation*, el gradiente no es propagado durante la actualización de los pesos. El rendimiento se ve entonces afectado si hay una gran cantidad de neuronas muertas en la red neuronal.

Hay varias generalizaciones de ReLU que modifican la función para que el gradiente tenga una pendiente no nula α, evitando este problema de forma

UNIVERSIDAD DE MÁLAGA

que la red pueda propagar gradientes negativos muy reducidos. Algunas de las variantes más utilizadas son: Leaky ReLU (Maas et al., 2013) (2.8), que fija un valor pequeño de 0.01 para gradientes negativos; Parametric ReLU (PReLU) (He et al., 2015) (2.9), que tiene en cuenta el parámetro adicional α, pudiendo tomar el valor deseado; *Exponential Linear Unit* (ELU) (Clevert et al., 2015) (2.10), que también tiene en cuenta α aplicándola junto a un término exponencial $(e^x - 1)$. En la Figura 2.10 (derecha) se muestra una representación de estas funciones de activación para su comparación visual.

$$\text{LeakyReLU}(x) = \begin{cases} 0.01\ x & si \quad x < 0 \\ x & si \quad x \geq 0 \end{cases} \tag{2.8}$$

$$\text{PReLU}(\alpha, x) = \begin{cases} \alpha\ x & si \quad x < 0 \\ x & si \quad x \geq 0 \end{cases} \tag{2.9}$$

$$\text{ELU}(\alpha, x) = \begin{cases} \alpha\ (e^x - 1) & si \quad x < 0 \\ x & si \quad x \geq 0 \end{cases} \tag{2.10}$$

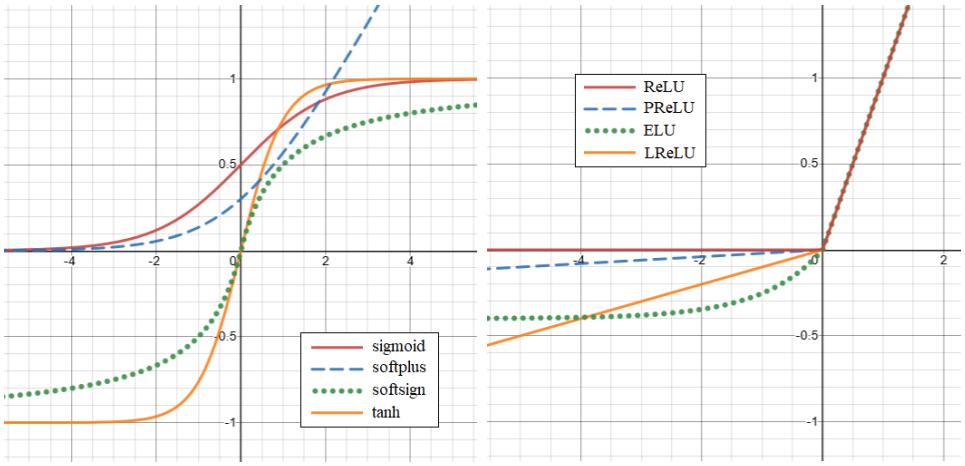

Figura 2.10: Representación de las funciones de activación sigmoid, tanh, softplus y softsign (izquierda) y las funciones ReLU, Leaky ReLU, PReLu y ELU (derecha).

Por tanto, se tiene que el hecho de que la función ReLU, y el resto de generalizaciones provenientes de la misma, se basan en el principio de que los modelos son más fáciles de optimizar si su comportamiento mantiene similitud con la función lineal. Esto hace que los gradientes obtenidos de los mismos sean más útiles para el aprendizaje de lo que serían con otras funciones de activación, lo que provocó que se hayan convertido en las funciones de activación de elección por defecto para las redes en los distintas modelos de DL. Aunque aquí se ha realizado una distinción entre las funciones Leaky

UNIVERSIDAD DE MÁLAGA

ReLU y PReLU, en la práctica son tratadas como equivalentes. Normalmente se utiliza la función de activación Leaky ReLU con valores de pendiente diferentes a 0.01, convirtiéndose de este modo en la función paramétrica, pero aun así se cita y menciona su uso como Leaky ReLU.

Aparte de las variedades de ReLU ya mencionadas, durante las últimas décadas se diseñaron otras funciones de activación, algunas de ellas relacionadas con la propia función ReLU, como es el caso de las funciones softplus y maxout, y otras que no, como la función softsign. La función de activación maxout será comentada en el apartado de regularización, y más concretamente, cuanto se comente la técnica de regularización dropout con la cual está estrechamente relacionada.

La función de activación softplus (2.11) puede verse como una versión flexible de ReLU. Planteada por Dugas et al. (2001), fue comparada con la función ReLU (Glorot et al., 2011), concluyendo que mediante la aplicación de esta última se alcanzan normalmente mejores resultados. Esta conclusión resulta además contrario a la hipótesis inicial de que la saturación a 0 de ReLU perjudica la optimización al bloquear la retropropagación del gradiente, por lo que se esperaba que al ser softplus totalmente diferenciable lo mejorara. Empíricamente fue pues contradicha esta hipótesis y actualmente se desaconseja de forma general el uso de softplus.

$$\text{softplus}(x) = \zeta(x) = \ln(1 + e^x) \tag{2.11}$$

La función de activación softsign (Glorot y Bengio, 2010) (2.12), por su parte, es similar a la función tangente hiperbólica y comparte con esta el mismo rango (-1, 1) en la imagen. Esta difiere de la tangente hiperbólica en su comportamiento en términos de saturación debido a que sus asíntotas responden de forma polinómica en lugar de exponencial, de forma que estas son más suaves. A la hora de su aplicación, las redes que utilizan la función softsign son más robustas en la inicialización que con tanh. Su uso no está extendido por su escaso rendimiento frente a ReLU.

$$\text{softsign} = \frac{x}{1 + |x|} \tag{2.12}$$

2.2.5. Regularización

Uno de los problemas más importantes en el aprendizaje automático, acentuado en las redes neuronales y los modelos de DL debido a su elevado número de pesos y sesgos, es el sobreajuste u *overfitting*. El sobreajuste surge cuando un modelo se ajusta excesivamente bien al conjunto de datos utilizado como entrada durante el entrenamiento, llegando a adaptarse al ruido que presenta el mismo, razón por la cual no es capaz de tratar bien con nuevas entradas que no hayan sido observadas durante el proceso de

UNIVERSIDAD DE MÁLAGA

entrenamiento. Afortunadamente existen diversas estrategias de regularización que están diseñadas para evitar el sobreajuste, es decir, reducir el error cometido con el conjunto de evaluación a expensas de un aumento del error en el entrenamiento. En el contexto del DL, la mayor parte de las estrategias de regularización se basan en la regularización de los estimadores, los cuales funcionan aumentando el sesgo para una varianza reducida, como es la regularización L^1 o L^2. Sin embargo, hay otros métodos como el aumento del conjunto de datos, la detección temprana o el método de dropout, ninguna de ellas excluyente con el resto.

Una de las técnicas de regularización más comunes es la técnica denominada L^2, también conocida como *weight decay*, *ridge regression* o regularización de Tikhonov. La idea de la regularización L^2 es añadir un término extra a la función objetivo, un término llamado el término de regularización $\Omega(w) = \frac{1}{2}\|w\|_2^2$, quedando la función objetivo del modelo (2.13) y el gradiente (2.14) como sigue:

$$\tilde{J}(w; X, y) = \frac{\alpha}{2}\ w^\top\ w + J(w; X, y) \tag{2.13}$$

$$\nabla_w\ \tilde{J}(w; X, y) = \alpha w + \nabla_w\ J(w; X, y) \tag{2.14}$$

El efecto deseado tras la aplicación de la regularización L^2 es que la red favorezca el aprendizaje de pesos pequeños, y que los pesos grandes únicamente sean permitidos si mejoran considerablemente la función de coste. Todo depende del valor de α: cuando α es pequeño significa que se elige minimizar la función de coste original, y cuando α es grande se favorecen los pesos pequeños.

Otra opción es utilizar la regularización L^1. Al igual que L^2, L^1 controla la intensidad de la regularización escalando la penalización Ω usando un hiperparámetro positivo α. Así, la función objetivo regularizada (2.15) y el correspondiente gradiente (2.16) distinguen el efecto de la regularización L^1 al de L^2. Específicamente, se observa que la contribución de regularización al gradiente ya no escala linealmente con cada w, sino mediante un factor constante con signo igual a $\text{sign}(w)$.

$$\tilde{J}(w; X, y) = \alpha\ \|w\|_1 + J(w; X, y) \tag{2.15}$$

$$\nabla_w\ \tilde{J}(w; X, y) = \alpha\ \text{sign}(w) + \nabla_w\ J(w; X, y) \tag{2.16}$$

Además, la regularización L^1 posee la propiedad de dispersión, lo que hace que se pueda utilizar como un mecanismo de selección de características. La selección de características simplifica los modelos al elegir un subconjunto de las características de los datos. En particular, el conocido modelo LASSO (Tibshirani, 1996) integra L^1 con un modelo lineal y una función de coste

UNIVERSIDAD DE MÁLAGA

por mínimos cuadrados, de forma que la penalización L^1 hace que un subconjunto de los pesos tenga valor cero. Esto sugiere que las características correspondientes pueden ser descartadas con seguridad.

Otra técnica de regularización común en DL es la parada temprana, debido tanto a su eficacia como a su sencillez. Normalmente, cuando se entrenan modelos grandes con capacidad de sobreajustarse, se observa que el error de entrenamiento disminuye constantemente con el tiempo, pero el error de validación llega a un punto en el que empieza a subir de nuevo. Esta técnica detiene el entrenamiento según la restricción que se desee, así puede ser directamente sobre el número de iteraciones de entrenamiento o sobre el error medido en la red, como error de mínimos cuadrados. De este modo, cuando se alcance un valor deseado, o bien se siga iterando y este no sea mejorado, el entrenamiento es detenido y se devuelve el mejor modelo alcanzado.

La última técnica de regularización mostrada es el dropout (Srivastava et al., 2014). Este proporciona un método computacionalmente ligero, pero de gran eficacia para llevar a cabo la regularización en modelos de DL. En una primera aproximación, el método de dropout puede ser considerado como un método de bagging práctico para conjuntos de grandes redes neuronales. El bagging (*bootstrap aggregating*) (Breiman, 1996), ya mencionado con anterioridad, es un algoritmo de aprendizaje automático basado en el muestreo con reemplazo del conjunto de datos de entrada, la aplicación de varios modelos diferentes separados sobre estos conjuntos muestreados y la integración de las salidas de los modelos mediante una votación. Esta técnica tiene poca practicidad cuando los modelos aplicados son grandes redes neuronales, ya que el entrenamiento y la evaluación de estas redes es costosa en tiempo de ejecución y memoria.

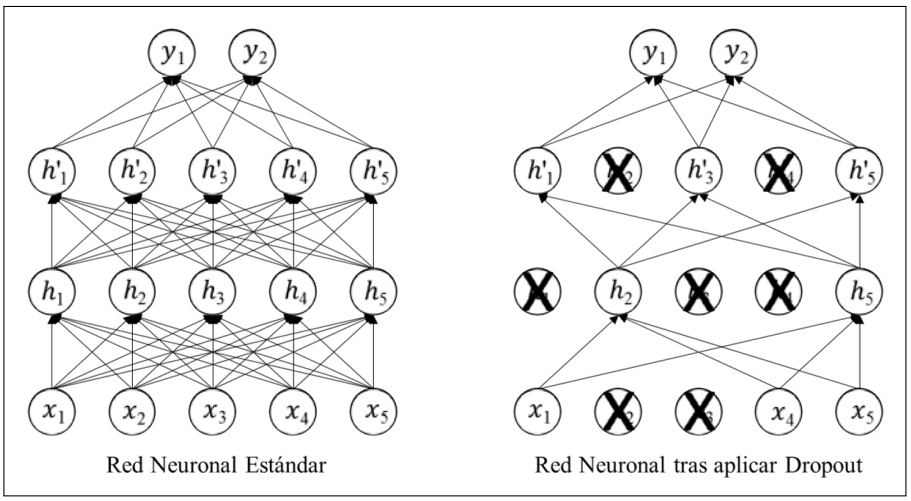

Figura 2.11: Ejemplo de aplicación del método dropout en una red neuronal.

UNIVERSIDAD DE MÁLAGA

El método dropout puede ser visto como una versión funcional de bagging para redes neuronales profundas, con importantes diferencias. Si se tiene en cuenta el entrenamiento, en bagging se tienen varios modelos independientes, mientras que en dropout se tiene un solo modelo. Dropout trabaja con diferentes configuraciones de la misma red neuronal, por lo que los parámetros se heredan de la configuración anterior permitiendo representar un número exponencial de modelos con una cantidad de memoria manejable. Además, en bagging cada modelo es entrenado con su respectivo conjunto de datos de entrenamiento, mientras que en dropout se utiliza el mismo conjunto de datos durante todo el proceso de entrenamiento de la red.

Específicamente, en el método dropout se abandonan unidades de la capa de entrada y las capas ocultas de una red neuronal, creando de esta forma subredes. Con abandonar una unidad nos referimos a su eliminación temporal de la red junto con todas sus conexiones entrantes y salientes (Figura 2.11). La elección de las unidades que deben ser abandonadas es aleatoria. En el caso más simple, cada unidad se conserva con una probabilidad independiente de otras unidades, donde esta probabilidad puede ser elegida usando un conjunto de validación o simplemente puede fijarse en 0.5, valor que parece ser cercano al óptimo en el abandono de unidades en las capas ocultas para una amplia gama de modelos y aplicaciones diferentes. Para las unidades de entrada la probabilidad óptima es normalmente más cercana a 1 que a 0.5, ya que la información se pierde directamente cuando se ignoran los nodos de entrada, mermando la capacidad de la red.

Aparte de la versión estándar de dropout, hay diferentes variaciones. La aplicación del método *fast dropout* (Wang y Manning, 2013) resulta en un tiempo de convergencia menor debido a una variación del cálculo del gradiente, siendo capaz de igualar el rendimiento del dropout estándar en problemas con redes neuronales pequeñas. Otra variación es el método *dropout boosting* (Warde-Farley et al., 2014), diseñado para carecer del efecto regularizador de dropout, por lo que no previene el sobreajuste del modelo. Debido a esto, su uso no se encuentra extendido.

Por último, se finaliza este apartado con un método mencionado anteriormente, la función de activación maxout (Goodfellow et al., 2013b). Esta función se encuentra diseñada con el propósito de mejorar la precisión de dropout, o más bien, de una versión temprana de este (Hinton et al., 2012a). De este modo, la función maxout se denomina así para señalar que la salida (*out*put) es el máximo (*max*) del conjunto de entradas. Las unidades que implementan la función maxout (2.17) dividen z en grupos de k valores, obteniendo como salida el elemento máximo de cada grupo, donde $z_{ij} = x^T W \cdots_{ij} + b_{ij}$, $W \in \mathbb{R}^{d \times m \times k}$ y $b \in \mathbb{R}^{m \times k}$ son parámetros aprendidos.

$$\text{maxout}(x)_i = \operatorname*{máx}_{j \in \{1,k\}} z_{ij} \tag{2.17}$$

UNIVERSIDAD DE MÁLAGA

Con un valor de k suficientemente grande, una sola unidad maxout es capaz de aproximar cualquier función convexa. Las unidades maxout presentan ciertas ventajas computacionales, al requerir en algunos casos menos parámetros. Si las características capturadas por n filtros lineales diferentes se pueden resumir tomando el máximo sobre cada grupo de k características sin perder información, la siguiente capa puede reducir el número de pesos k veces. Gracias a que cada unidad es impulsada por múltiples filtros, las unidades maxout mantienen cierta redundancia que les ayuda a resistir un fenómeno llamado olvido catastrófico, en el cual las redes olvidan cómo realizar tareas en las que fueron entrenados en iteraciones pasadas (Goodfellow et al., 2014b). Sin embargo, su uso no está extendido. Recientemente se ha demostrado que el entrenamiento con maxout es más costoso computacionalmente que mediante funciones de activación tradicionales, y que la función de activación ReLU funciona mejor que cualquier activación maxout cuando se aumenta el número de filtros convolucionales (Castaneda et al., 2019).

2.2.6. Optimización del gradiente

Se ha visto como en los distintos modelos de DL está implicada la optimización en muchos contextos, pero de todos los problemas en los que es aplicable la optimización, el más complejo es el entrenamiento de las redes. Actualmente, en cuanto a la práctica se refiere, una forma muy usual de optimizar el entrenamiento de estos modelos es el empleo de algoritmos de optimización de descenso del gradiente, sin embargo, su empleo es a menudo como optimizadores de caja negra.

Dentro de los diferentes algoritmos de optimización de este tipo, se pueden distinguir dos grupos. Los denominados básicos, que se basan en la introducción de hiperparámetros, y entre los que se encuentra el algoritmo de descenso del gradiente estocástico, el *momentum*, la aceleración de Nesterov y el *FTRL-Proximal*; y los denominados algoritmos con tasa de aprendizaje adaptativo, una serie de métodos incrementales (o basados en *mini-batchs*) introducidos en la última década, y que se basan en la adaptación automática de la tasa de aprendizaje de los parámetros del modelo, entre los que se encuentran los algoritmos AdaGrad, AdaDelta, Adam y RMSProp.

El descenso del gradiente estocástico (SGD) y sus variantes son probablemente los algoritmos de optimización más utilizados para el aprendizaje automático en general. Este algoritmo requiere de una tasa de aprendizaje ϵ, un hiperparámetro θ y un *mini-batch* $\{x^1...x^m\}$, para computar la estimación del gradiente (2.18) y realizar la actualización del hiperparámetro (2.19).

$$\hat{g} \leftarrow +\frac{1}{m} \nabla_\theta \sum_i L(f(x^i;\theta), y^i) \tag{2.18}$$

$$\theta \leftarrow \theta - \epsilon\, \hat{g} \tag{2.19}$$

UNIVERSIDAD DE MÁLAGA

En la práctica, la tasa de aprendizaje no puede ser fija, es necesario disminuir esta gradualmente con el tiempo. Esto se debe a que el estimador de gradiente SGD introduce una fuente de ruido (el muestreo aleatorio de m ejemplos de entrenamiento) que no desaparece incluso cuando se llega a un mínimo. SGD elimina la redundancia realizando una actualización a la vez, lo que hace que tenga un tiempo de ejecución moderado, si bien el tiempo de cálculo por actualización no crece con el número de ejemplos de entrenamiento, lo que permite la convergencia incluso cuando el número de ejemplos de entrenamiento es elevado. Aun así, el aprendizaje con SGD puede resultar lento.

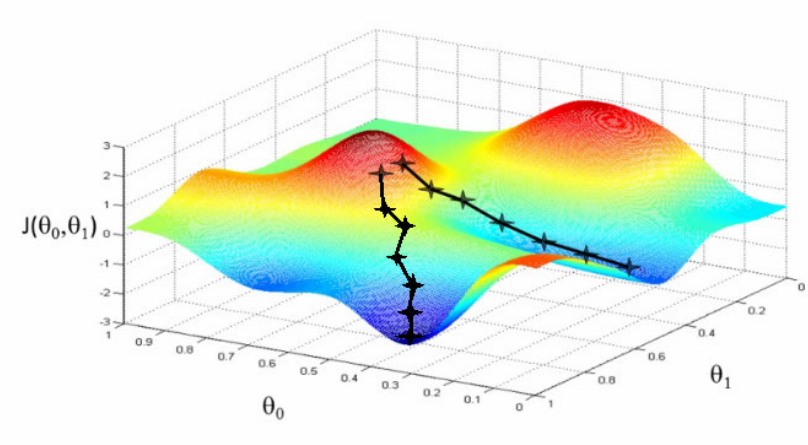

Figura 2.12: Gráfica ejemplificando la minimización del gradiente[10].

Si bien el algoritmo SGD ha sido la estrategia de optimización más popular, también se han utilizado otras como el método del *momentum* (Polyak, 1964). Este fue diseñado para acelerar el aprendizaje, especialmente frente a una alta curvatura, gradientes pequeños pero consistentes o gradientes ruidosos. El término *momentum* es cogido oportunamente del campo de la física, en la cual representa el producto de la masa por la velocidad, por lo que una partícula adquiere más *momentum* mientras esta va acelerando y su velocidad es mayor. Formalmente, el algoritmo de *momentum* introduce dos variables, una variable v que desempeña el papel de velocidad, y representa tanto dirección como velocidad a la que los parámetros se mueven a través del espacio de parámetros, además de acelerar el aprendizaje; y un hiperparámetro $\alpha \in [0, 1]$ que determina la rapidez con que las contribuciones de los gradientes anteriores decaen exponencialmente. De este modo, a la estimación del gradiente de SGD (2.18), se le añada la actualización de la velocidad (2.20) y la actualización del hiperparámetro (2.21).

[10]Extraído de Amazon (2017)

UNIVERSIDAD DE MÁLAGA

$$v \leftarrow \alpha v - \epsilon \, \nabla_\theta \left(\frac{1}{m} \sum_i L(f(x^i; \theta), y^i) \right) \tag{2.20}$$

$$\theta \leftarrow \theta + v \tag{2.21}$$

Más recientemente, Sutskever et al. (2013) introdujo una variante al algoritmo *momentum* basándose en el método de la aceleración del gradiente de Nesterov (Nesterov, 1983). La diferencia entre el *momentum* y el método de Nesterov surge del momento en el que se evalúa el gradiente. Con Nesterov el gradiente es evaluado después de aplicar la velocidad, de forma que se puede interpretar este método como un intento de añadir un factor de corrección al método *momentum*. De esta forma, la actualización del hiperparámetro sigue siendo igual, pero en la estimación del gradiente evaluamos además la aplicación de la velocidad (2.22).

$$v \leftarrow \alpha v - \epsilon \, \nabla_\theta \left(\frac{1}{m} \sum_i L(f(x^i; \theta + \alpha v), y^i) \right) \tag{2.22}$$

Antes de comenzar con los algoritmos de optimización con tasa de aprendizaje adaptativo, se tiene también el algoritmo *Follow the Regularized Leader* (FLTR-Proximal) (McMahan, 2011). Sin regularización, el algoritmo FTRL-Proximal es idéntico al método del descenso del gradiente estocástico, pero gracias a que utiliza una representación perezosa alternativa se puede agregar alguna regularización para estabilizar el modelo, de forma que el regularizador es el que da origen al algoritmo de gradiente en línea. En el estudio inicial la técnica de regularización utilizada en conjunto con FTRL-Proximal era la regularización L^1 (McMahan, 2011).

Uno de los primeros algoritmos con tasa de aprendizaje adaptativo es el algoritmo AdaGrad (Duchi et al., 2011). Este adapta individualmente las tasas de aprendizaje de todos los parámetros del modelo escalándolos de forma inversamente proporcional a la raíz cuadrada de la suma de todos sus valores históricos cuadrados. Los parámetros con unas mayores derivadas parciales tienen una disminución elevada en su tasa de aprendizaje, mientras que los parámetros con pequeñas derivadas parciales tienen una disminución relativamente pequeña. Se ha demostrado que la acumulación de gradientes cuadrados desde el inicio del entrenamiento puede resultar en una disminución prematura y excesiva de la tasa de aprendizaje durante el entrenamiento de redes profundas, razón por la que AdaGrad únicamente es eficaz en un número reducido de modelos de DL (Mukkamala y Hein, 2017).

Una extensión de AdaGrad es el algoritmo AdaDelta (Zeiler, 2012), el cual busca reducir el efecto decreciente, monótono y agresivo de AdaGrad sobre la tasa de aprendizaje. En lugar de acumular todos los gradientes cuadrados pasados, AdaDelta restringe la ventana de gradientes acumulados a

UNIVERSIDAD DE MÁLAGA

algún tamaño fijo w. Además, en lugar de almacenar ineficientemente w gradientes cuadrados, la suma de gradientes se define de forma recursiva como un promedio en descomposición de todos los gradientes cuadrados previos, el promedio depende entonces del promedio anterior y del gradiente actual. Otro algoritmo que parte de AdaGrad es el algoritmo RMSProp (Hinton et al., 2012b), el cual lo modifica cambiando la acumulación de gradiente por una media móvil ponderada exponencialmente. AdaGrad está diseñado para converger rápidamente cuando se aplica a una función convexa, pero reduce la tasa de aprendizaje tan rápidamente que puede que esta sea demasiado pequeña antes de llegar a una estructura suficientemente convexa. RMSProp utiliza un promedio exponencial para que pueda converger rápidamente después de encontrar la estructura covexa.

Por último tenemos uno de los algoritmos de optimización con tasa de aprendizaje adaptativo más recientes, el algoritmo Adam (Kingma y Ba, 2014). En el contexto de los algoritmos anteriores, puede ser contemplado como una variante de la combinación de RMSProp y el algoritmo *momentum*, si bien manteniendo algunas diferencias importantes. En primer lugar, en Adam el momento se incorpora directamente como una estimación con ponderación exponencial del gradiente, mientras que en RMSProp suele ser aplicado a los gradientes reescalados. En segundo lugar, Adam incluye correcciones de sesgo a las estimaciones tanto de los momentos de primer orden (el término del momento) como de los momentos de segundo orden (no centrados), mientras que RMSProp incorpora también una estimación del momento de segundo orden, pero carece del factor de corrección. Esto provoca que, a diferencia de Adam, la estimación de momento de segundo orden en RMSProp pueda tener sesgos altos en el entrenamiento. Adam es considerado el algoritmo de optimización por defecto para modelos de DL al suponer una mejora de RMSProp. Por ejemplo, Adam es el algoritmo de optimización recomendado por defecto en el software TensorFlow (Abadi et al., 2016).

2.3. Modelos generativos profundos

Por lo general los modelos de DL están compuestos de múltiples módulos no lineales, de modo que su función de pérdida es casi no convexa, lo que los hace difíciles de optimizar al poder generarse múltiples óptimos locales durante el proceso de aprendizaje. Por el contrario, la función de pérdida de una estructura poco profunda como una red neuronal tradicional con una sola capa oculta o una máquina de vectores de soporte, suele ser convexa, de forma que la optimización del parámetro se hace más efectiva. La primera aproximación de los modelos generativos profundos se fundamentaba en que la transformación del algoritmo de aprendizaje para resolver el problema de optimización convexo es una mejor manera de entrenar modelos profundos que el resolver problemas no convexos directamente.

UNIVERSIDAD DE MÁLAGA

2.3.1. Máquinas de Boltzmann

Las máquinas de Boltzmann (BM) (Ackley et al., 1985) son el primer acercamiento a los modelos generativos profundos. Estos modelos son capaces de aprender características importantes de una distribución de probabilidad desconocida basándose en muestras de la misma, sin embargo, es un proceso difícil y que requiere mucho tiempo de cómputo. Este problema se simplifica imponiendo restricciones a la topología de la red, lo que lleva a restringir las máquinas de Boltzmann. Las *Restricted Boltzmann Machines* (RBM), que fueron diseñadas bajo el nombre de *harmonium* (Smolensky, 1986), son un tipo particular de campos aleatorios de Markov con una arquitectura de dos capas, una capa de variables observables conectada a una capa de variables ocultas o latentes. Esta posee la ventaja de que se pueden apilar RBM para formar modelos que sean más profundos.

Otro tipo de modelo generativo profundo es la *Deep Boltzmann Machine* o DBM (Salakhutdinov y Hinton, 2009). A diferencia de la *Deep Belief Network*, que veremos a continuación, la DBM es un modelo completamente no dirigido y, a diferencia del RBM, el modelo DBM tiene varias capas de variables latentes, teniendo RBM únicamente una capa. Aunque dentro de cada capa, al igual que el modelo RBM, cada una de las variables son mutuamente independientes, condicionadas a las variables en las capas vecinas. Al igual que RBM y DBN, los DBM normalmente contienen solamente unidades binarias, pero es fácil incluir variables observables reales.

Tanto las máquinas de Boltzmann o las RBM, como el resto de modelos de DL se vieron resarcidos tras la introducción de las *Deep Belief Networks* (DBN) (Hinton et al., 2006), aunque actualmente estos modelos rara vez son utilizados, incluso en comparación con otros algoritmos de aprendizaje no supervisados o modelos generativos, pero todavía son merecidamente reconocidos por su importante papel en la historia del DL. Los DBN son modelos gráficos probabilísticos que contienen múltiples capas de variables ocultas, o latentes, de forma que cada capa no lineal capta patrones de los datos más complejos de forma progresiva. Este proceso representaba una forma innovadora de resolver problemas asociados con el reconocimiento de objetos, la comprensión del lenguaje y la percepción del habla (Bengio, 2009). Las variables latentes son normalmente variables binarias, mientras que las unidades visibles pueden ser binarias o reales. En las DBN no existen conexiones entre las diferentes capas, en su lugar cada neurona en cada capa está conectada a cada neurona en cada capa vecina, si bien también se pueden diseñar modelos que tengan conexiones más limitadas. Otra característica de las DBN es que las conexiones entre las dos capas superiores no están dirigidas, pero entre el resto de capas sí se tienen conexiones dirigidas hacia la capa que está más cerca de los datos, al contrario que ocurre en las *Deep Boltzmann Machines*, como mencionamos con anterioridad.

UNIVERSIDAD DE MÁLAGA

Este algoritmo de aprendizaje entusiasmó a los investigadores debido principalmente a tres características cruciales. La primera es el uso de un algoritmo voraz capa por capa, capaz de discernir conjuntos de parámetros del modelo con suficiente rapidez incluso en modelos con un número elevado de capas no lineales y millones de parámetros. La segunda es que el modelo es capaz de hacer un uso eficiente de conjuntos muy grandes de datos sin etiquetar, por lo que puede ser pre-entrenado de forma no supervisada, y posteriormente realizar un ajuste mediante datos etiquetados. Y la tercera es la posibilidad de realizar una inferencia aproximada de una manera eficiente, lo que permite que los valores de variables latentes presentes en la capa más profunda sean fáciles de inferir. En definitiva, las DBN abrieron nuevos debates en el ámbito del DL que produjeron nuevos avances y mejoras en modelos como las CNN, las RNN o los *autoencoders*.

2.3.2. Autoencoders

Una red neuronal *autoencoder* (LeCun, 1987; Bourlard y Kamp, 1988) es un algoritmo de aprendizaje no supervisado que aplica *backpropagation*, estableciendo los valores objetivo para que estos sean similares a las entradas recibidas. Esto significa que la red está diseñada de modo que aprenda una aproximación a la función identidad, no la función identidad en sí. De este modo, los *autoencoders* se diseñan incluyendo restricciones que únicamente permitan copias aproximadas de las entradas, por ejemplo, limitando el número de unidades ocultas en la red. Debido a estas restricciones, los *autoencoders* se ven obligados a priorizar qué propiedades de los datos de entrada deberían ser copiadas, descubriendo en el proceso información útil sobre los propios datos.

Lar redes *autoencoders* han sido aplicadas con éxito en tareas de reducción de la dimensionalidad, si bien a partir de los trabajos de Hinton et al. (2006); Hinton y Salakhutdinov (2006) son utilizadas más habitualmente en tareas de aprendizaje no supervisado en conjunto con otros modelos de DL, dando lugar a un aprendizaje semi-supervisado. El objetivo del aprendizaje no supervisado es aprender patrones de un conjunto de datos de entrada cuando no hay valores de salida especificados, al contrario que en el aprendizaje supervisado. En este caso de aplicación semi-supervisado, el objetivo final es realizar aprendizaje supervisado, ya que los datos de entrada sí poseen valores de salida a priori. Los resultados obtenidos de los *autoencoders* son entonces utilizados como un aprendizaje no supervisado suponiendo un pre-entrenamiento de una red neuronal profunda, para acelerar el aprendizaje en tareas de clasificación (Tissera y McDonnell, 2016). Lo habitual es el uso de los pesos obtenidos en las conexiones de la red *autoencoder* tras el pre-entrenamiento, como pesos iniciales de la posterior red profunda.

En cuanto a su estructura, las redes *autoencoders* pueden ser conside-

UNIVERSIDAD DE MÁLAGA

radas como un caso especial de FNN, de forma que durante el proceso de entrenamiento las distintas capas de la red van tomando una función de codificación de la entrada, o una función de decodificador encargado de realizar la reconstrucción. Los *autoencoders* pueden ser pues entrenados mediante las mismas técnicas que una FNN pero, a diferencia de estas, también pueden ser entrenados usando el algoritmo de recirculación (Hinton y McClelland, 1987), el cual está basado en la comparación entre las activaciones de la red en la entrada original y las activaciones en la entrada reconstruida. Este algoritmo de recirculación es considerado más plausible, biológicamente hablando, que el algoritmo *backpropagation*, pero rara vez es utilizado en aplicaciones de aprendizaje automático.

Como ya hemos mencionado con anterioridad, los *autoencoders* pretenden aproximar la función identidad, no hacer una copia perfecta entre entrada y salida, y lo consiguen mediante restricciones que le fuerzan a priorizar propiedades de los datos, revelando así información útil de los mismos. Estas restricciones pueden ser de diversa índole, lo que hace que se tengan diferentes variaciones de estos modelos. La restricción más básica es la limitación de la capa intermedia para que tenga una dimensión menor que la entrada, obligando al modelo a capturar las características más sobresalientes de los datos de entrenamiento. Los modelos que usan esta limitación se denominan incompletos (*undercomplete autoencoders*). Otra restricción se basa en el uso de una función de pérdida, de forma que esta incite al *autoencoder* a aprender otras propiedades más allá de la capacidad de copiar la entrada. Los *autoencoders* que se benefician de esta restricción se denominan regularizados (*regularized autoencoders*). Además, este tipo de limitación permite entrenar cualquier arquitectura del modelo, sin importar la dimensión de la capa intermedia respecto a la entrada.

Dentro de los *regularized autoencoders* se agrupan modelos como los denominados *sparse autoencoders* (Ranzato et al., 2006). Estos aplican como criterio de entrenamiento una penalización de dispersión $\Omega(h)$ en la capa intermedia. Esta penalización presenta una conexión con el término $\log Z$ que surge cuando se aplica máxima verosimilitud a un modelo probabilístico no dirigido, de forma que imponer la dispersión en un *autoencoder* evitara que este tenga errores de reconstrucción reducidos, al igual que minimizar el $\log Z$ impide que un modelo probabilístico tenga una alta probabilidad en todos sus puntos. Otra variante es el *denoising autoencoder* (Vincent et al., 2008), un modelo que, en lugar de agregar una penalización a la función de coste, aprende características de los datos mediante un entrenamiento cuyo objetivo es predecir la entrada original recibiendo como entrada datos corruptos (Figura 2.13). En el entrenamiento es necesaria una asignación estocástica para corromper los datos de entrada originales x, obteniendo unos datos corruptos $\tilde{x}$, que son usados posteriormente como entrada por un *autoencoder* estándar con la excepción de que la función de pérdida debe ser calculada

UNIVERSIDAD DE MÁLAGA

para la entrada original $L(x, \tilde{x}')$. Por último, tenemos los *stacked denoising autoencoders* (Vincent et al., 2010), una extensión de los anteriores y de los *stacked autoencoders* (Bengio et al., 2007). En los *stacked denoising autoencoders* se entrenan las capas de forma sucesiva con el objetivo de predecir la entrada original de la capa anterior dada una entrada corrupta, de forma que la capa h_1 se entrena para predecir x dado $\tilde{x}$, h_2 se entrena para predecir h_1 dado $\tilde{h}_1$, y así sucesivamente hasta la capa de salida.

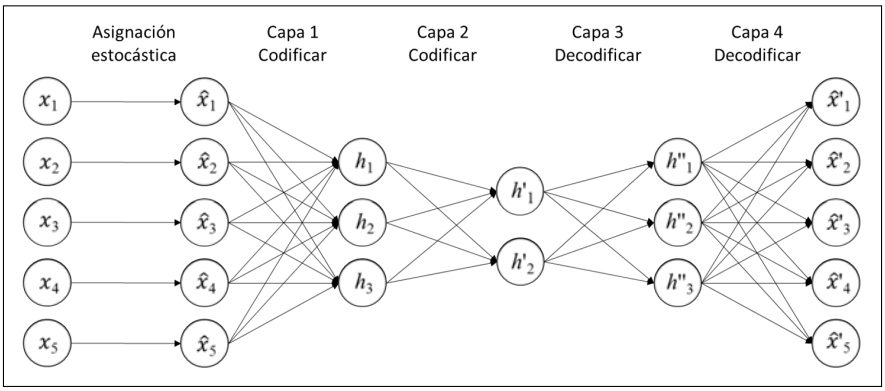

Figura 2.13: Ejemplo de *Denoising Autoencoder*

2.3.3. Variational Autoencoders

Los modelos *Variational Autoencoders* (VAE) (Kingma y Welling, 2013) son una variación de la red de *autoencoder* estándar (Bourlard y Kamp, 1988), descrita en la sección anterior. Un VAE consta de dos redes neuronales interconectadas denominadas codificador y decodificador. El codificador es un modelo de inferencia $q(z|x, \theta_e)$ que asigna los datos de entrada x a un espacio variable latente de menor dimensión z, donde el parámetro θ_e representa los pesos de las capas del codificador en la VAE. Por otra parte, la red decodificadora obtiene el espacio latente z de variables como entrada, y como salida modela la distribución de probabilidad de los datos $p(x|z, \theta_d)$, donde θ_d son los pesos de las capas del decodificador.

La principal diferencia de un VAE con una red *autoencoder* es que, en lugar de generar directamente un vector latente y maximizar la log-verosimilitud marginal (*log-likelihood*), se genera una distribución gaussiana multivariante. De este modo, siendo s el tamaño del espacio latente deseado, se tendrán dos conjuntos de neuronas de salida de la red codificadora: un conjunto con s neuronas para el vector de medias μ de la distribución; y un conjunto de s neuronas para las desviaciones estándar σ de la distribución, si se utiliza una distribución gaussiana multivariante con covarianza diagonal, que es lo más habitual. Ambos conjuntos de neuronas se combinan para formar el vector latente. Sin embargo, la combinación directa de estos

UNIVERSIDAD DE MÁLAGA

parámetros en el vector latente z implica que esta es una variable aleatoria continua, lo que no permitiría a la red conocer la distribución que proviene del codificador. Para resolver este problema y expresar la variable aleatoria z como variable determinista, es necesario realizar el truco de reparametrización (Kingma y Welling, 2013; Rezende et al., 2014), donde z depende de los parámetros de salida del codificador (μ, σ) y además de una variable ε muestreada de una distribución gaussiana (Ecuación 2.23).

$$z^{(i,l)} = \mu^{(i)} + \sigma^{(i)} \odot \varepsilon^{(l)}, \quad \varepsilon^{(l)} \sim \mathcal{N}(0, \mathbf{I}) \tag{2.23}$$

Otra diferencia del modelo VAE con respecto al *autoencoder* estándar es que VAE maximiza el *Evidence Lower Bound* (ELBO) en la log-verosimilitud marginal de $p(x)$, de forma que su función objetivo se muestra en la Ecuación 2.24. El término KL en la ecuación es la *Kullback-Leibler divergence* (Kullback y Leibler, 1951) entre la distribución del codificador $q(z|x)$ y la distribución anterior $p(z)$. Esto se usa como un regularizador, midiendo la cantidad de información que se pierde cuando se usa la distribución q para representar p. En la práctica, VAE se entrena mediante el algoritmo de retropropagación del error (Rumelhart et al., 1986) para actualizar los pesos de la red. De este modo, toda la red VAE se entrena en conjunto y no en pasos separados, usando como función de pérdida el valor de ELBO negativo.

$$\min_{p} \mathbb{E}_{q(z|x,\theta_e)}[\log p(x|z,\theta_d)] - \mathrm{KL}(q(z|x,\theta_e)||p(z)) \tag{2.24}$$

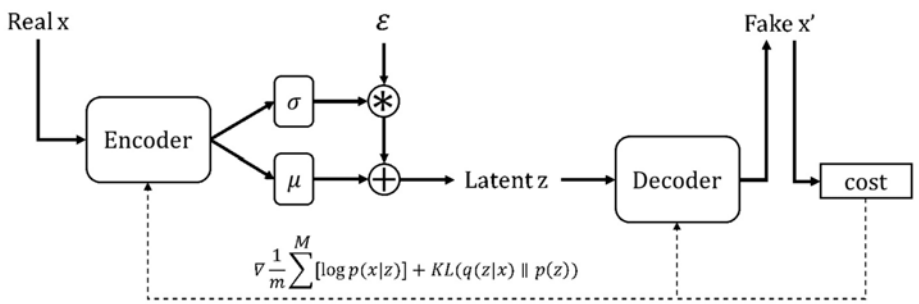

Figura 2.14: Diagrama esquemático del funcionamiento del modelo VAE.

El diagrama esquemático del modelo VAE representado en la Figura 2.14, muestra el flujo de información: la entrada x del codificador, el espacio latente intermedio z que se crea a partir de la distribución ε y los vectores de media μ y desviación estándar σ, y la salida del modelo. También se muestra la propagación de la pérdida a través de la red, pérdida que se infiere de la función objetivo (Ecuación 2.24). Esta pérdida corresponde a un *batch* completo en una iteración del modelo e indica la variación que debe aplicarse tanto al codificador como al descodificador. Para generar una muestra

UNIVERSIDAD DE MÁLAGA

sintética con este modelo una vez ha sido entrenado, es necesario pasarle algún tipo de ruido de tamaño igual al espacio latente, como entrada al decodificador. De este modo, el decodificador tratará de recrear una entrada, obteniendo con ello la muestra sintética.

El modelo *Denoising* VAE (DVAE) (Im et al., 2017) es una variación de los VAE estándar. Al igual que la versión *autoencoder*, el modelo DVAE recibe como entrada una deformación de la entrada original x, siguiendo una distribución aleatoria que crea una entrada ruidosa $\tilde{x}$. El propósito es que al mapear la deformación ruidosa $\tilde{x}$ a la entrada real x, el modelo genere una representación latente más robusta que una VAE estándar. La Ecuación 2.25 muestra la función objetivo a minimizar para el caso del modelo DVAE, donde se tiene en cuenta la nueva entrada con ruido. Una representación del flujo de información del modelo se muestra en la Figura 2.15, donde se observa que el resto del proceso de entrenamiento y generación del modelo es el mismo que con el modelo VAE.

$$\min_{p} \mathbb{E}_{q(\tilde{x}|x)}[\mathbb{E}_{q(z|\tilde{x})}[\log p(x|z)] - \mathrm{KL}(q(z|\tilde{x})||p(z))] \tag{2.25}$$

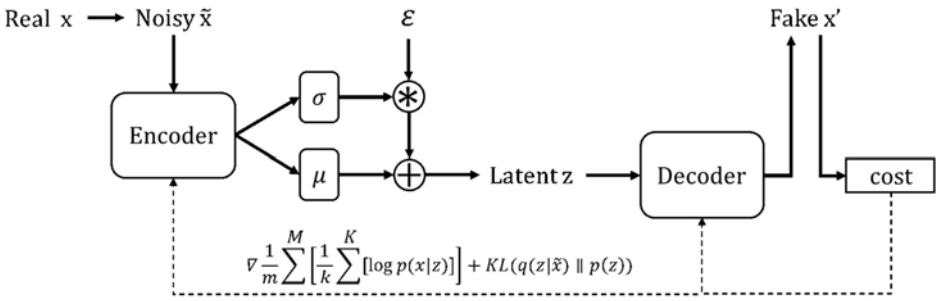

Figura 2.15: Diagrama esquemático del funcionamiento del modelo *Denoising* VAE.

Otra variante del modelo VAE es la *Conditional* (CVAE) (Kingma et al., 2014; Sohn et al., 2015). Como se muestra en la Figura 2.16, en esta variante se agrega tanto en el codificador como en el descodificador, la información referente a una condición y, que puede ser la etiqueta de clase u otra información de los datos. Esto hace que la red sea consciente del tipo de muestra, ya sea su clase u otra información complementaria añadida, que debe mapearse en el espacio latente z del codificador. La idea de esta modificación es doble: por un lado, debería mejorar la capacidad de aprendizaje de la red para discernir entre clases de muestras, mejorando recreación de las muestras originales; y por otro, permite la asignación de una etiqueta de clase a la muestra generada por el decodificador. La función objetivo en este

UNIVERSIDAD DE MÁLAGA

caso debe modificarse para considerar la condición y como se muestra en la Ecuación 2.26.

$$\min_{p} \mathbb{E}_{q(z|x,y)}[\log p(x|z,y)] - \mathrm{KL}(q(z|x,y)||p(z|y)) \tag{2.26}$$

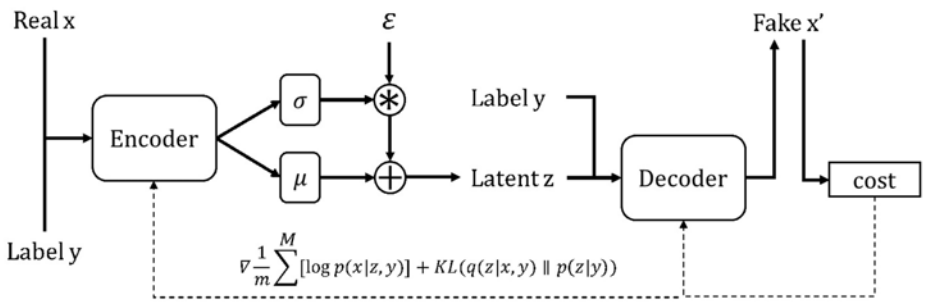

Figura 2.16: Diagrama esquemático del funcionamiento del modelo *Conditional* VAE.

Una aplicación de los modelos VAE en la biomedicina puede observarse en el trabajo de Kusner et al. (2017). En este trabajo se presenta el modelo *Grammar* VAE (GVAE) para trabajar con datos discretos siempre que exista una gramática para el problema, como las expresiones aritméticas o las estructuras moleculares. El problema con estos datos es la fragilidad de la representación, pequeños cambios en la cadena pueden llevar a objetos completamente diferentes o que no corresponden en absoluto a objetos válidos. Para abordar este problema, su propuesta GVAE incorpora directamente el conocimiento utilizando una gramática. Así, GVAE codifica y decodifica directamente desde y hacia los objetos gramaticales en lugar de las cadenas, garantizando que todas las salidas sean válidas. En la Figura 2.17 se muestra la aplicación de este modelo para generación de estructuras moleculares. Otra aplicación que también ha obtenido interés en los últimos años es el análisis transcriptómico de células individuales (*Single-cell RNA-seq*) (Grønbech et al., 2020; Svensson et al., 2020). El RNA-seq de una sola célula hace posible la investigación de la variabilidad en la expresión génica entre las células y la dependencia de la variación en el tipo de célula. Con estos modelos generativos profundos se puede hacer una comparación de modelos basada en la probabilidad, aprender una representación latente de las células y capturar mejor la variabilidad en diferentes poblaciones de células. Sin embargo, la mayoría de los modelos VAE diseñados no se aplican en DA con el propósito de mejorar el rendimiento en tareas de clasificación/predicción.

UNIVERSIDAD DE MÁLAGA

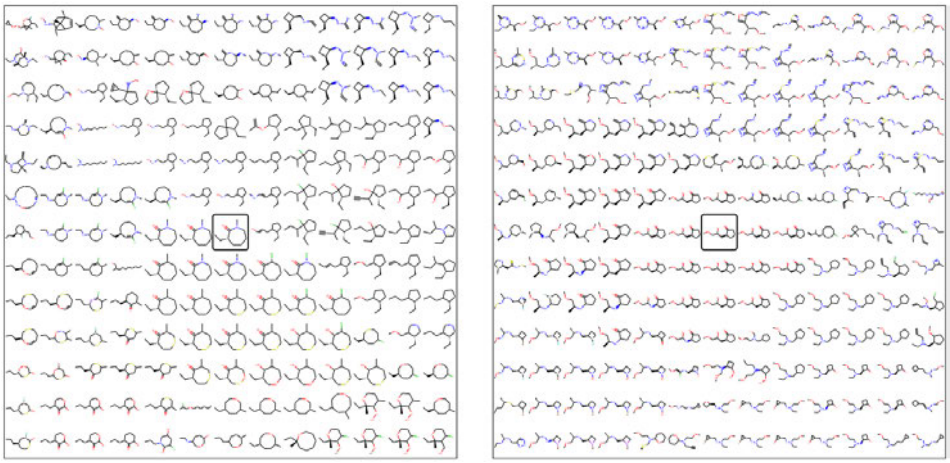

Figura 2.17: Aplicación del modelo *Grammar* VAE en la replicación de estructuras moleculares[11].

2.3.4. Generative Adversarial Networks

Las denominadas *Generative Adversarial Networks* (GAN) (Goodfellow et al., 2014a) son los modelos del DL que están liderando el camino de los modelos de generación de datos sintéticos, especialmente en el procesamiento y la generación de imágenes, debido a su velocidad de cálculo y la calidad de los resultados obtenidos. El impresionante rendimiento de las GAN ha dado lugar a una mayor atención sobre cómo se pueden aplicar en DA, multiplicándose exponencialmente en los últimos años. Estas redes tienen la capacidad de generar nuevos datos de entrenamiento que dan como resultado una mejora en el rendimiento de los modelos de clasificación.

Los modelos GAN consisten esencialmente en dos redes neuronales conectadas, llamadas respectivamente *generator* y *discriminator*, entrenadas simultáneamente en un proceso adversario. Estas dos redes se entrenan simultáneamente, dando lugar a un enfrentamiento entre ambas para que aprendan la una de la otra. En este contexto, el objetivo de la red discriminador (D) es distinguir si una muestra proviene del conjunto de datos 'realesó es una muestra generada, es decir, para la muestra de entrada x el discriminador estima la probabilidad de la muestra pertenece a la distribución real o no. Por otro lado, la red generador (G) toma como entrada una distribución aleatoria ruidosa z, y produce como salida una distribución $G(z)$ asignada al espacio de las muestras reales. El propósito del generador es crear nuevas muestras sintéticas con características que se aproximen a las presentes en las muestras reales. Por tanto, el generador crea nuevas muestras intentando que la red discriminador no pueda distinguir estas muestras sintéticas como muestras no derivadas de la distribución real. Así, el proceso de generación se opone

[11] Extraído de Kusner et al. (2017)

UNIVERSIDAD DE MÁLAGA

completamente al proceso de discriminación, dando lugar al entorno competitivo señalado anteriormente. Las herramientas de análisis derivadas de la teoría de juegos, como la estrategia *minimax* y el equilibrio de Nash (Fedus et al., 2018), sugieren que el generador eventualmente conseguirá engañar al discriminador, lo que hace que el modelo sea potencialmente satisfactorio.

$$\min_{G} \max_{D} \; \mathbb{E}_{x\sim p_{data}(x)}[\log D(x)] + \mathbb{E}_{z\sim p_z(z)}[\log(1 - D(G(z)))] \tag{2.27}$$

La función objetivo del modelo Vanilla GAN (modelo estándar) mostrada en la Ecuación 2.27), se pueden observar dos partes que se identifican con el comportamiento del modelo: una parte relacionada con el reconocimiento de muestras que son reales ($\mathbb{E}_{x\sim p_{data}(x)}[\log D(x)]$), y otro con el reconocimiento de aquellas muestras que son falsas ($\mathbb{E}_{z\sim p_z(z)}[\log(1 - D(G(z)))]$). De esta forma, la capacidad del modelo para reconocer si las muestras son reales o falsas se expresa en la Ecuación 2.28, y el error de la red relacionado con el reconocimiento de muestras falsas está modelado por la Ecuación 2.29.

$$\max_{D} \; \mathbb{E}_{x\sim p_{data}(x)}[\log D(x)] + \mathbb{E}_{z\sim p_z(z)}[\log(1 - D(G(z)))] \tag{2.28}$$

$$\min_{G} \mathbb{E}_{z\sim p_z(z)}[\log(1 - D(G(z)))] \tag{2.29}$$

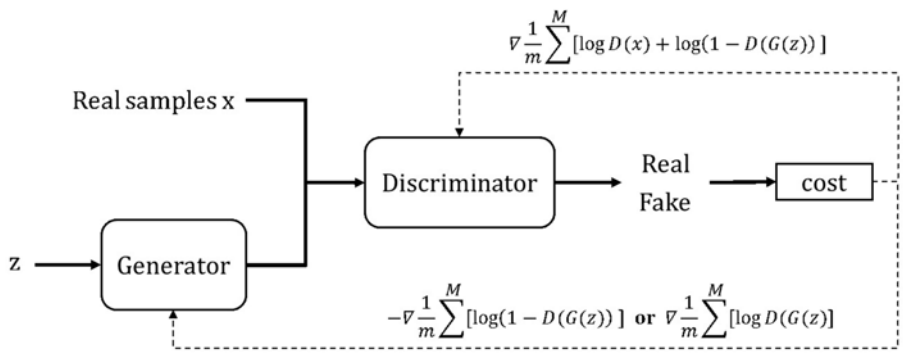

Figura 2.18: Diagrama esquemático del funcionamiento del modelo GAN.

En la Figura 2.18 se muestra una representación esquemática de una GAN. Se puede observar en ella el flujo de información, incluyendo las entradas de las redes generador y discriminador, la salida del modelo, que corresponde a la probabilidad de que la muestra de entrada en el discriminador sea real, los errores propagados a ambas redes, y todo el entorno competitivo en general. En cuanto al proceso de generación de muestras sintéticas con GAN, simplemente es necesario pasarle algún tipo de ruido a la entrada del generador, y este arroja como salida la muestra generada. Normalmente, se

UNIVERSIDAD DE MÁLAGA

realiza una comprobación adicional con la red discriminador, que recibe como entrada esta nueva muestra y predice la probabilidad de que provenga de la distribución real. En caso afirmativo, la muestra tiene suficiente calidad como para engañar a la red, y puede ser utilizada posteriormente para una aplicación de DA.

Una modificación esencial aplicada a la Vanilla GAN para mejorar su rendimiento en procesamiento de imágenes dio lugar al modelo *Deep Convolutional* GAN (DCGAN) (Radford et al., 2015). La modificación en concreto es el reemplazo de los perceptrones multicapa presente en las redes generador y discriminador de GAN, por capas convolucionales. Desde la introducción de DCGAN al campo, todos los demás modelos GAN, incluidos los diseñados con anterioridad, se implementan utilizando capas convolucionales, lo que limita su uso a problemas que presenten información espacial o temporal.

La Vanilla GAN presenta ciertos fallos recurrentes en su proceso, que no son solucionados por el uso de capas convolucionales. Uno de estos fallos, denominado "*vanishing gradient*", surge cuando la red discriminador es demasiado buena en su objetivo. En efecto, cuando se tiene un discriminador óptimo que siempre detecta cuales son las muestras creadas por el generador, este no recibe suficiente información para avanzar, de forma que durante el entrenamiento los gradientes van desapareciendo. Otro fallo común es el "*collapse mode*", que surge cuando el modelo se atasca en un mínimo local. Cuando se aplica una GAN se espera que esta produzca una amplia variedad de resultados, sin embargo, si un generador produce una salida especialmente plausible como muestra real, el generador puede quedarse atascado produciendo solo esa salida. Como se ha explicado anteriormente, en el entorno competitivo, el generador siempre trata de generar datos que logren engañar al discriminador. Si el generador comienza a producir la misma salida (o un pequeño conjunto) una y otra vez, la mejor estrategia del discriminador debería ser aprender a rechazar siempre esa salida. Pero, si el discriminador se atasca en un mínimo local y no encuentra la mejor estrategia, entonces es demasiado fácil para el generador encontrar la forma de engañarlo. Cada iteración el generador se optimiza en exceso para un estado particular del discriminador, y el discriminador no logra aprender cómo salir de la trampa. Como resultado, el generador alterna entre un pequeño conjunto de tipos de salida, sin ofrecer variedad en las muestras generadas.

Una de las variantes de la Vanilla GAN que soluciona las desventajas que esta presenta, es la Wasserstein GANs (Arjovsky et al., 2017). Para reducir el efecto causado por los problemas de estabilidad y velocidad en el entrenamiento, se utiliza la distancia Wasserstein (distancia de Earth Mover) para evaluar la distancia entre los datos sintéticos y los datos reales. De forma intuitiva, se puede ver como el trabajo mínimo necesario para transformar una distribución en otra, donde el trabajo se define como el producto de la masa de la distribución que se tiene que mover y la distancia que se

UNIVERSIDAD DE MÁLAGA

debe mover. En la Ecuación 2.30 se define la distancia Wassertein entre dos distribuciones $\mathbb{P}_r$ y $\mathbb{P}_g$. $\gamma(x, y)$ puede verse como la cantidad de masa que debe moverse de x a y para transformar $\mathbb{P}_r$ en $\mathbb{P}_g$

$$\mathbb{W}(\mathbb{P}_r, \mathbb{P}_g) = \inf_{\gamma \in \Pi(\mathbb{P}_r, \mathbb{P}_g)} \mathbb{E}_{(x,y)\sim\gamma}[\| x - y \|] \tag{2.30}$$

$$\mathbb{W}(\mathbb{P}_r, \mathbb{P}_g) \propto \operatorname*{máx}_{w \in W} \mathbb{E}_{x\sim\mathbb{P}_r}[f_w(x)] - \mathbb{E}_{z\sim Z}[f_w(g_\theta(z))] \tag{2.31}$$

Comparada con la divergencia Jensen-Shannon utilizada en la GAN original, la distancia Wasserstein es continua y casi diferenciable en todas partes, y también es significativa, converge a 0 a medida que las distribuciones se acercan y divergen a medida que se alejan. Para su aplicación en el modelo WGAN, es necesario su cálculo usando el teorema de dualidad de Kantorovich y Rubinstein (Villani, 2008), dando lugar a la Ecuación 2.31.

$$\operatorname*{mín}_{G} \operatorname*{máx}_{D} \; \mathbb{E}_{x\sim p_{data}(x)}[D(x)] - \mathbb{E}_{z\sim p_z(z)}[D(G(z))] \tag{2.32}$$

Como consecuencia del uso de la distancia Wassertein y el cambio en la función objetivo 2.32, también es necesario realizar un cambio en la red discriminador, que es renombrada como critic, pero conserva la denotación D. En WGAN, como salida de esta red se tiene una puntuación, que es realmente la distribución que se pasaría como entrada a las neuronas de salida con función de activación sigmoidea, las cuales se elimina de la red. Además, en WGAN se realizan varias iteraciones de la red critic por cada iteración del generador, en vez de realizar una iteración de cada. Finalmente, para lograr el objetivo de encontrar una función 1-Lipschitz, es necesario realizar una operación denominada *weight clipping*, que es un ajuste en los pesos de la red critic para que los valores de los pesos presentes en ella se mantengan dentro de un rango controlado.

$$\text{GP} \longrightarrow \lambda \mathbb{E}_{\hat{x}\sim p(\hat{x})}[(\| \nabla_{\hat{x}} D(\hat{x}) \|_2 - 1)^2] \tag{2.33}$$

$$\operatorname*{mín}_{G} \operatorname*{máx}_{D} \; \mathbb{E}_{x\sim p_{data}(x)}[D(x)] - \mathbb{E}_{z\sim p_z(z)}[D(G(z))] - \text{GP} \tag{2.34}$$

Una variante del modelo WGAN que ofrece un mejor rendimiento es el modelo WGAN con penalización del gradiente (WGAN-GP) (Gulrajani et al., 2017). En este modelo la operación de *weight clipping* se reemplaza por el uso de un término de penalización del gradiente. Este término, mostrado en la Ecuación 2.33, se calcula usando un coeficiente de penalización de gradiente λ, y usando $\hat{x}$, que se calcula a partir de datos reales y sintéticos. De esta forma, la función de costo objetivo del modelo WGAN-GP se modifica con respecto a WGAN, incluyendo la penalización del gradiente (2.34). Con esta penalización de gradiente, el modelo WGAN-GP tiene las ventajas de WGAN sobre GAN, como una mayor estabilidad durante el entrenamiento y una mayor independencia entre las redes discriminador y generador, para

UNIVERSIDAD DE MÁLAGA

que este último pueda seguir aprendiendo incluso si el discriminador tiene un buen desempeño; y evita los problemas de WGAN debido al tamaño de la ventana de recorte de peso, como la convergencia lenta si se tiene una ventana grande y la desaparición de gradientes si se tiene una ventana pequeña. Aun así, WGAN-GP tiene una mayor complejidad computacional y aun puede colapsar. En la Figura 2.19 se muestra un diagrama de flujo del modelo.

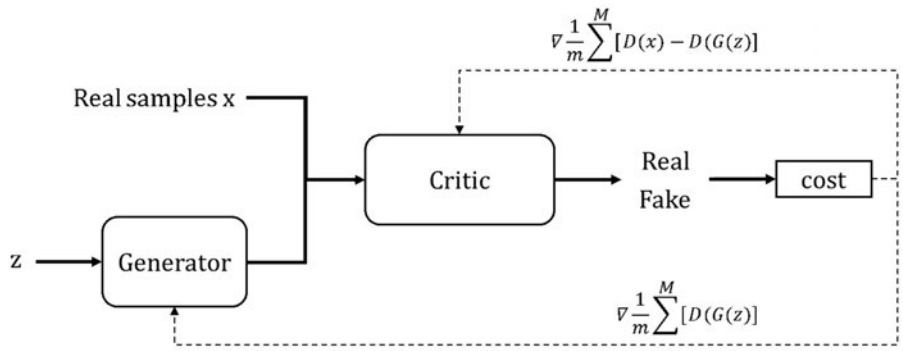

Figura 2.19: Diagrama esquemático del funcionamiento del modelo Wasserstein GAN con penalización del gradiente.

Otra variante de la Vanilla GAN que debe ser considerada es la *Conditional* GAN (CGAN) (Mirza y Osindero, 2014). Al igual que con el modelo CVAE con respecto a VAE, la característica que diferencia al modelo CGAN con respecto a GAN, es la adición de información relativa a la etiqueta y de la muestra. Como se muestra en la Figura 2.20, la etiqueta se agrega al espacio latente z en la entrada de la red generadora y a las muestras reales y generadas en la entrada de la red discriminador. La única variación con respecto a la función de costo objetivo del modelo GAN es la adición de la etiqueta de la muestra y en el cálculo de los resultados obtenidos de ambas redes (Ecuación 2.35). Se asume que introducir esta información adicional da un punto de partida al modelo CGAN para saber qué características buscar, introduciendo un sesgo en los nodos y dando lugar a mejores resultados, especialmente en el caso de las imágenes.

$$\min_{G} \max_{D} \; \mathbb{E}_{x\sim p_{data}(x)}[\log D(x|y)] + \mathbb{E}_{z\sim p_z(z)}[\log(1 - D(G(z|y)))] \tag{2.35}$$

Con el propósito de asignar una etiqueta de clase a las muestras generadas por el generador, se presentó a mediados de 2017 el modelo *Auxiliary Classifier* GAN (ACGAN) (Odena et al., 2017). Al igual que en CGAN, en ACGAN se proporcionan al generador como entrada tanto el espacio latente como la etiqueta de clase. La principal diferencia está en la red discriminador, a la que solo se proporciona la muestra como entrada, sin la etiqueta de

UNIVERSIDAD DE MÁLAGA

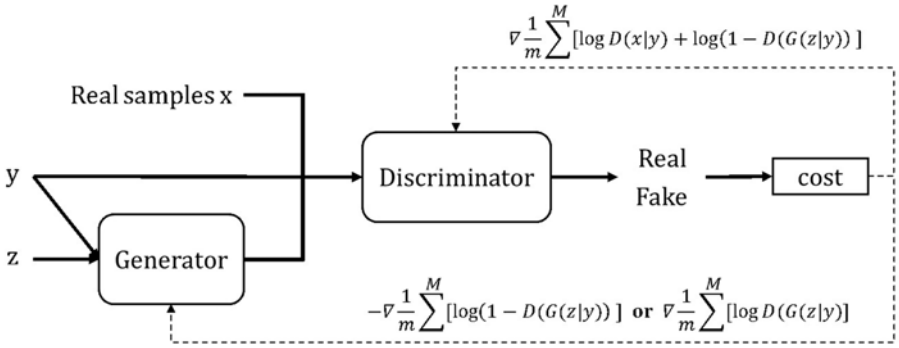

Figura 2.20: Diagrama esquemático del funcionamiento del modelo *Conditional* GAN.

clase. La red discriminador debe predecir si la muestra dada es real o falsa como antes, pero también debe predecir la etiqueta de clase de la muestra. La arquitectura del discriminador y el clasificador auxiliar utilizan la misma red, es decir, en la práctica se implementan como un único modelo de red neuronal con dos salidas. La primera salida es una probabilidad única a través de la función de activación sigmoidea que indica la probabilidad de que la muestra sea "real", y la segunda salida es la probabilidad de que la muestra pertenezca a cada clase. Sin embargo, al compartir una misma red para dos propósitos, ACGAN sufre de ciertos fallos a la hora de generar muestras y asignarles etiquetas, generando muestras casi idénticas para la mayoría de las clases al entrar en *collapse mode*, especialmente si el problema lleva consigo un número grande de etiquetas (Miyato y Koyama, 2018). También presenta un problema de rendimiento cuando el conjunto de datos no presenta una estructura espacial o temporal, debido a que el potencial de la dualidad de propósitos radica en el uso de capas convolucionales en el discriminador, ayudándose de la información espacial de los datos. De hecho, no se ha encontrado literatura que aplique este modelo a otro tipo de problemas que no sean procesamiento de imágenes, texto o señales.

Otro modelo GAN de interés por su magnífico rendimiento en generación de imágenes de alta resolución, se conoce como *Progressively Growing* GAN (Karras et al., 2018). Este modelo presenta una arquitectura en la cual se entrenan una serie de redes que van aumentando la complejidad en la resolución de la imagen de forma progresiva. Esta arquitectura se basa en el concepto de que las GAN pueden aceptar como entrada del generador tanto muestras reales como vectores aleatorios. Por lo tanto, las redes están encadenadas pasando muestras desde una GAN de menor resolución a una GAN de mayor resolución. Concretamente, las resoluciones varían de 4×4 a 8×8 y así sucesivamente hasta que se alcanzan salidas de tamaño 1024×1024, produciendo resultados excelentes en la generación de imágenes faciales.

UNIVERSIDAD DE MÁLAGA

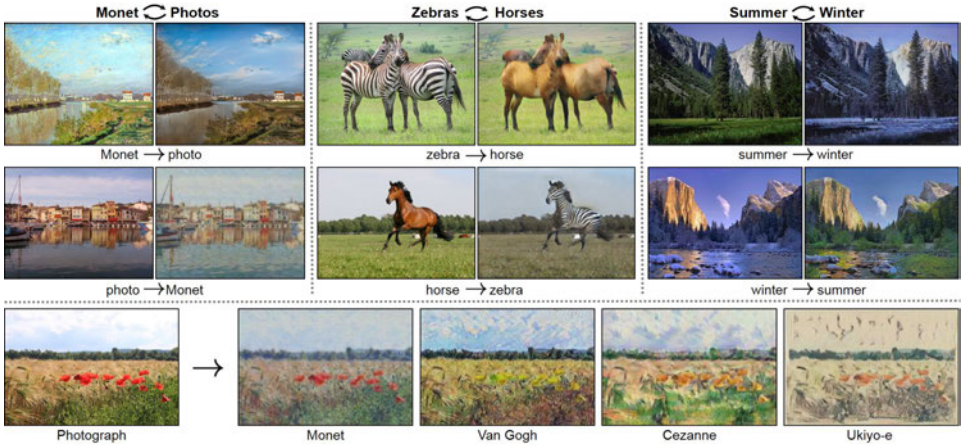

Figura 2.21: Aplicación del modelo CycleGAN en la traducción de imágenes entre varios dominios[12].

El último modelo general que vamos a mencionar en esta sección es la CycleGAN (Zhu et al., 2017). Este modelo es interesante porque permite la traducción de una muestra entre dominios, como se muestra en la Figura 2.21. Para realizar esta traducción, CycleGAN utiliza funciones de pérdida de coherencia adicionales aplicadas hacia adelante y hacia atrás. De esta forma, un generador toma muestras de un dominio y aprende a modificarlas para que un discriminador no pueda decir si originalmente eran parte de otro dominio. Después, estas nuevas muestras pasan a través de otra red que las traduce nuevamente al dominio inicial y un segundo discriminador determina si pertenece al conjunto original. Las pérdidas obtenidas de ambas redes discriminador se agregan para formar la pérdida de consistencia asignada durante el ciclo. Al igual que el resto de modelos GAN, su uso actual es casi con exclusividad el procesamiento de imágenes por su implementación con capas convolucionales, sin embargo, su potencial para cualquier otra aplicación es grande siempre que los dominios presenten características similares. Por ejemplo, podría permitir traducir muestras genéticas de un tipo de cáncer a otro, o de pacientes sanos a muestras de pacientes enfermos para balancear el conjunto de datos.

2.3.5. Aplicaciones de GAN en DA y biomedicina

Repasamos en esta sección trabajos recientes sobre la aplicación de modelos GAN en el campo del aumento de datos, con especial atención a los trabajos aplicados en problemas del ámbito de la biomedicina. Como hemos mencionado en la sección anterior, debido a la arquitectura de aprendizaje

[12]Extraído de Zhu et al. (2017)

UNIVERSIDAD DE MÁLAGA

profundo que presentan los modelos GAN con capas convolucionales, la mayoría de las aplicaciones están en la generación de imágenes sintéticas para la tarea DA. Por ejemplo, podemos citar el trabajo (Fajardo et al., 2021) que realiza DA con modelos generativos profundos condicionales, como los modelos CVAE, CGAN y ACGAN en el conjunto de dígitos manuscritos de MNIST. Aparte, realiza una comparación con SMOTE para mostrar el potencial de la aplicación de estos modelos para realizar sobremuestreo en conjuntos desbalanceados. Zhu et al. (2018) presenta otro ejemplo de aplicación en DA, presentando la generación de imágenes faciales mediante una CycleGAN para mejorar la clasificación de las emociones. La aplicación de GANs en tareas de DA también se han probado con éxito en otros dominios, como las tareas de procesamiento de lenguaje y el reconocimiento automático de voz (Hsu et al., 2017b); clasificación de modulación de señal digital (Tang et al., 2018a); y predicción en series de tiempo, específicamente en el diagnóstico de fallos en maquinaria industrial (Shao et al., 2019).

Si nos centramos en los problemas bioinformáticos, donde las muestras disponibles son escasas y difíciles de obtener, encontramos que multitud de estudios han demostrado que el uso de estos modelos de generación aplicados en DA con imágenes médicas puede mejorar los resultados obtenidos en tareas de clasificación. Wu et al. (2018) desarrolla una *conditional infilling* GAN para aumentar conjuntos de tomografías computarizadas (TC) de cáncer de mama. Este modelo tiene como entrada la lesión reemplazada con valores aleatorios, una máscara que representa la ubicación de la lesión y la clase a la que pertenece (maligno/no maligno). Sandfort et al. (2019) utiliza una CycleGAN para transformar imágenes TC cerebrales con contraste en imágenes sin contraste y mejorar la generalización en tareas de segmentación. Frid-Adar et al. (2018) prueba la eficacia del uso de DCGAN para generar imágenes TC de lesiones hepáticas de alta calidad y presenta un esquema novedoso para la clasificación de lesiones hepáticas utilizando CNN. Han et al. (2018) aplica los modelos DCGAN y WGAN para generar imágenes sintéticas de resonancia magnética (RM) cerebral de múltiples secuencias. Las principales dificultades estudiadas en el trabajo son el bajo contraste de las imágenes RM, la fuerte consistencia en la anatomía del cerebro y las variabilidades entre secuencias. Zhao et al. (2018) aplica un modelo generador llamado F&BGAN (Forward&Backward GAN) en su aplicación a imágenes TC de cáncer de pulmón. El modelo diseñado utiliza una primera red GAN para generar imágenes (Forward) y una segunda GAN a modo de autoencoder (Backward) para refinar la imagen generada. Finalmente, debido al impacto global generado, es necesario mencionar el trabajo de Waheed et al. (2020), en el que se aplica un modelo ACGAN para generar imágenes sintéticas de rayos X del tórax (CXR) con las que aumentar el entrenamiento de una CNN para detección de COVID-19.

UNIVERSIDAD DE MÁLAGA

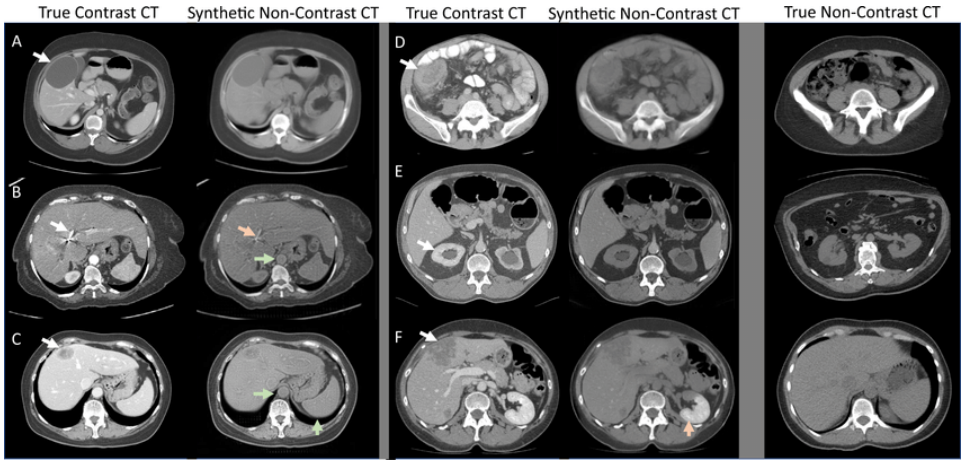

Figura 2.22: Generación de tomografías computarizadas cerebrales sin contraste a partir de imágenes con contraste[13].

Los trabajos presentados muestran que ya es un hecho que el uso de modelos GAN para la creación de datos sintéticos en problemas cuyo dominio son datos con información espacial o temporal, y la aplicación de estos datos en aplicaciones DA, mejora satisfactoriamente las tareas de clasificación. Este hecho hace que proliferen las aplicaciones con este tipo de imágenes médicas, como la tomografía computarizada, la resonancia magnética, las radiografías o las tomografías por emisión de positrones. Sin embargo, la aplicación de modelos generativos profundos en otros dominios no ha tenido tanto éxito y es más difícil encontrar trabajos relacionados. Fuera del área de la biomedicina y de los datos estructurados, el enfoque estudiado hasta el momento en tareas de DA y el uso de modelos generativos profundos, es su aplicación en conjuntos desequilibrados. Se utilizan estos modelos como algoritmos de sobremuestreo, creando únicamente muestras que pertenecen a la clase minoritaria, lo que equilibra los conjuntos de datos y debe ayudar a mejorar las tareas de clasificación y predicción. Con este enfoque podemos mencionar el trabajo de Douzas y Bacao (2018), que analiza el desempeño de una CGAN sobre 12 conjuntos de datos obtenidos de la base de datos UCI (Dua y Graff, 2017), con una ligera mejora de la sensibilidad de los clasificadores. También encontramos el trabajo de dos Santos Tanaka y Aranha (2019), que usa una GAN estándar para el mismo propósito, pero sobre un grupo excesivamente reducido de conjuntos de datos y una experimentación ligeramente deficiente. En ambos estudios, los resultados muestran que existe un potencial en su aplicación, consiguiendo una ligera mejora en la capacidad de predicción con respecto a otros métodos estándar.

Si nos centramos en el uso de modelos GAN para realizar DA sobre conjuntos de datos que no son imágenes en problemas biomédicos, podemos

[13]Extraído de Sandfort et al. (2019)

UNIVERSIDAD DE MÁLAGA

mencionar el trabajo de Liu et al. (2019). En él se realiza un análisis de la aplicación de un modelo WGAN como generador de datos a partir de un pequeño conjunto de muestras de suero (78 muestras) donadas por el Hospital Tongji en Wuhan, China, con el objetivo de identificar el estadio del cáncer. Una desventaja del enfoque utilizado es el uso de una WGAN para cada clase del conjunto de datos, y así poder mantener la clase en las muestras generadas. El estudio muestra un aumento en la eficiencia de predicción de modelos clasificadores gracias a este modelo, en comparación con métodos tradicionales como SMOTE. Chen et al. (2020) desarrolla un modelo de aumento de datos genómicos a escala poblacional basado en el modelo CGAN (PG-CGAN) tratando de mejorar la cantidad y diversidad de datos genómicos. La clave de su modelo se encuentra en el aprovechamiento de la información espacial presente en la estructura del genoma analizado, por lo que no es aprovechable para otros problemas. Los resultados para aumentar los genotipos en las regiones del antígeno leucocitario humano (HLA) que lograron con el modelo desarrollado, muestran que se generan nuevos genotipos con una distribución y estructura poblacional similar a los presentes en el conjunto de datos original. El trabajo realizado por Marouf et al. (2020) tiene como objetivo mejorar los escasos datos de secuencia de RNA unicelulares mediante el desarrollo de un modelo basado en *Similarity Constraint* en GAN (SCGAN) para el análisis de transcriptómica unicelular. Los resultados de los datos en poblaciones unicelulares aumentadas con células generadas mejoran ligeramente los análisis posteriores, así como la detección de genes marcadores y la clasificación de tipos de células.

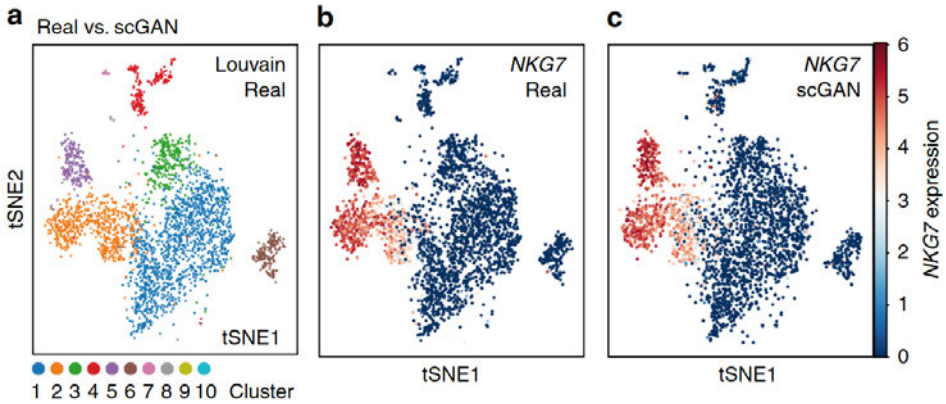

Figura 2.23: Visualización t-SNE de las células reales (a). Expresión del gen NKG7 en células reales (b) y generadas por scGAN (c)[14].

Por último, queremos profundizar en los trabajos de Chaudhari et al. (2020) y Xiao et al. (2021), que son muy similares a parte del trabajo que se

[14]Extraído de Marouf et al. (2020)

UNIVERSIDAD DE MÁLAGA

ha desarrollado en el presente trabajo de tesis. El objetivo de ambos trabajos es aplicar un modelo GAN en conjuntos de datos de expresión génica para generar nuevas muestras y mejorar el diagnóstico del cáncer. En nuestras investigaciones, en la sección 6.1, realizamos un experimento similar con el propósito de mejorar la predicción del estatus vital con 18 tipos de cáncer. En concreto, el trabajo de Chaudhari et al. (2020) propone un generador GAN donde el generador se alimenta con los datos originales a los que se le añade ruido multivariado. Según el artículo, como los datos generados se encuentran dentro del espacio latente, la red alcanza una convergencia antes. Como crítica al respecto, bajo nuestra experiencia, esto conlleva a dos posibles casuísticas: el colapso de la red generadora, más teniendo en cuenta que usan una GAN estándar, no una CGAN o WGAN; o a la copia casi directa de las muestras reales. Aplicado a datos de microarrays de expresión génica de conjuntos de datos disponibles en el repositorio NCBI, alcanzan un rendimiento extraordinario, con una mejora en el porcentaje de aciertos de la clasificación de aproximadamente el 15 %. Este rendimiento lleva asociadas varias dudas, pues no se menciona el tamaño de los conjuntos de entrenamiento y test, el haber realizado varias pruebas con validación cruzada o varias semillas, o no haber analizado los conjuntos generados. Tampoco reportan una comparación con otros métodos de DA como la adición de ruido, SMOTE o el remuestreo. Igualmente, aun con cierta desconfianza, este trabajo muestra, al igual que nuestros estudios, que la aplicación del DA conlleva un aumento en el rendimiento de los modelos clasificadores.

Por otra parte, en el trabajo de Xiao et al. (2021) se experimenta con 3 tipos diferentes de cáncer (brca, luad, stad) utilizando un modelo WGAN para generar datos únicamente de la clase minoritaria que no presenta tumor, por lo que esta es la única clase que es pasada al generador y, por tanto, la clase asignada de forma directa a los datos sintéticos generados. Utiliza una máquina de vectores de soporte para la clasificación y obtiene mejores resultados gracias al balanceo del conjunto con los datos generados. Aproximadamente obtiene una mejora del 30 % respecto a la aplicación de los datos sin aumento, y un 8 % frente a la aplicación de SMOTE. Como pequeña crítica, los experimentos son realizados usando conjuntos de test que los autores han escogido deliberadamente, con mismo número de muestras de test para ambas clases, sin realizar validación cruzada ni diferentes semillas. A esto hay que añadir que los resultados obtenidos con el conjunto sin aumentar presentan el peor escenario posible (capacidad predictiva del 50 %). Además, los resultados de t-SNE (Van der Maaten y Hinton, 2008) con los datos generados, una representación gráfica en dos dimensiones similar al análisis de componentes principales, muestran una configuración lineal extraña para las muestras. Estos resultados son similares a los obtenidos en el presente trabajo de tesis cuando aún nos encontrábamos ajustando los modelos generativos, e indicaban una creación de muestras errónea debido

UNIVERSIDAD DE MÁLAGA

a un mal rendimiento del modelo. La comparación de las muestras reales con las muestras sintéticas dejaba clara su mala calidad, con valores para las características completamente diferentes y cuasi-aleatorios. En la Figura 2.24 se puede observar los resultados obtenidos por Xiao et al. (2021), cuya comparación entre las muestras creadas con los modelos GAN respecto a las generadas con SMOTE, muestran diferencias extremadamente acuciadas, de igual forma si las comparamos con las muestras generadas por Marouf et al. (2020) (Figura 2.23). Aun así, dejando de lado esta pequeña desconfianza en la experimentación, al igual que pasaba con el anterior trabajo revisado (Chaudhari et al., 2020), este refleja que el uso de estos modelos generativos en el ámbito de la bioinformática empleando datos genómicos es un tema que actualmente se está tratando y tiene un gran interés.

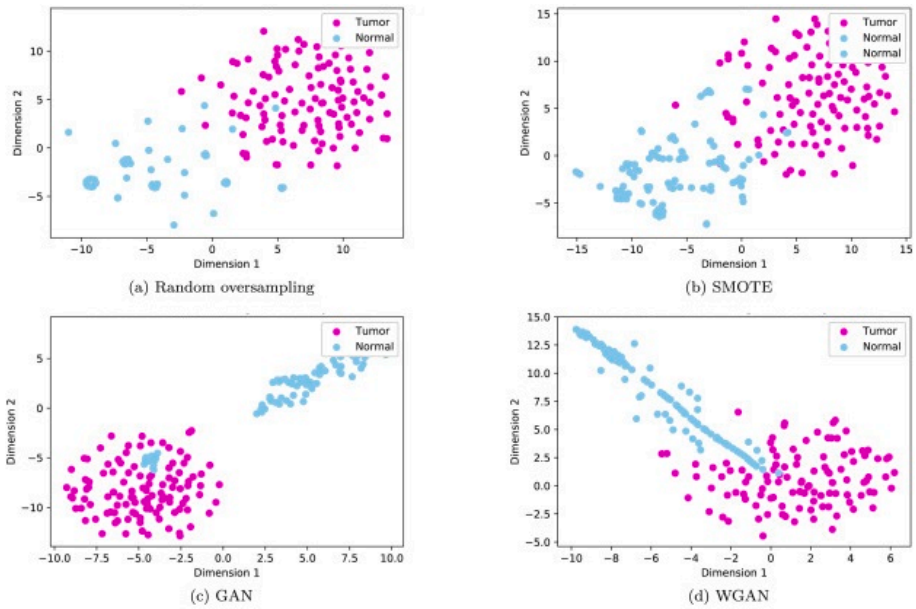

Figura 2.24: Visualización t-SNE de las muestras generadas mediante diverson modelos generativos en el estudio del diagnóstico de cáncer[15].

Teniendo en cuenta los trabajos anteriores, existe una idea clara de que el uso de modelos generativos profundos para realizar DA en dominios donde las muestras no son imágenes o series temporales (datos estructurados), es una tarea desafiante. Los trabajos en problemas biomédicos donde se manejan datos genómicos, que son significativamente más dispersos, costosos y limitados, son escasos y recientes, pero demuestran que estos modelos pueden ser beneficiosos para aumentar la tasa de predicción en clasificación.

[15]Extraído de Xiao et al. (2021)

UNIVERSIDAD DE MÁLAGA

UNIVERSIDAD
DE MÁLAGA

Capítulo 3

Técnicas de Aprendizaje y Evaluación

Resumen: En este capítulo se describen las diferentes metodologías utilizadas para alcanzar los objetivos de la investigación. En primer lugar, se explican los métodos de selección de características utilizados, presentes en la mayoría de aplicaciones bioinformáticas debido al elevado número de variables implicadas en los problemas. A continuación, se detallan las medidas de evaluación empleadas en casi todos los estudios realizados, para comprobar el rendimiento de los modelos y la calidad de los resultados. Finalmente, se explican los fundamentos de dos modelos de aprendizaje clásicos, como son la regresión logística y los sistemas de máquinas de vectores de soporte.

3.1. Métodos de selección de características

La selección de características es una técnica importante en muchas aplicaciones debido a los inmensos conjuntos de datos que se utilizan actualmente en ciertas áreas, donde se tienen hasta decenas de miles de variables; con especial atención a su uso con conjuntos de datos de la biomedicina, donde la mayoría de problemas presentan la maldición de la dimensionalidad, con conjuntos donde la cantidad de características no solo es alta, sino que es mayor que el número de muestras que contiene. Que los conjuntos presenten una gran cantidad de características puede dificultar enormemente el aprendizaje realizado en los modelos de generación y clasificación, no solo por la complejidad en tiempo de ejecución al entrenar los algoritmos, sino por la construcción e interpretación del modelo en sí, especialmente si una gran cantidad de características son irrelevantes y no proporcionan información útil para la predicción de clases o la diferenciación entre instancias. Por tanto,

UNIVERSIDAD DE MÁLAGA

el propósito de realizar una selección de características es múltiple, mejora el aprendizaje y la interpretación de modelos y algoritmos; mejora la velocidad de entrenamiento; y reduce el posible sobreajuste debido a variables redundantes.

La mayoría de métodos de selección de características pueden utilizarse con un objetivo adicional, proporcionar una nueva percepción sobre los datos y sobre el problema en cuestión, gracias a la obtención de ranking de importancia de las características (Guyon y Elisseeff, 2003). En la literatura se pueden encontrar varios métodos con los que es posible realizar esta tarea de selección de características, incluso diferentes métricas cuyo propósito más general no es este, como los métodos de filtrado, donde los métodos de correlación y las pruebas estadísticas (*Pearson's correlation*, *Chi-Square*) se pueden utilizar para calcular una puntuación para cada característica, clasificándolas según su utilidad al ser usadas para estimar el modelo de clasificación/predicción. También existen métodos *wrapper*, que utilizan modelos de aprendizaje para encontrar el subconjunto de características con las que se logra el mejor rendimiento en las pruebas de acuerdo a métricas de evaluación como la capacidad predictiva o el área bajo la curva ROC. Finalmente, existen métodos embebidos, que son una combinación de los dos métodos anteriores y que realizan un proceso de clasificación y selección de características mediante el uso de un algoritmo de aprendizaje. Entre estos métodos se encuentran el LASSO y los modelos de Random Forest que han sido utilizados en diversos estudios del presente trabajo de tesis.

3.1.1. Modelo LASSO

El modelo LASSO (Tibshirani, 1996), acrónimo de *Least Absolute Shrinkage and Selection Operator*, es un método que realiza dos tareas principales, selección de características y regularización para modelos de regresión lineal. El método LASSO intenta minimizar la suma del error cuadrático del modelo de regresión para mejorar la predicción e interpretación de los modelos de regresión. Para alcanzar este objetivo, LASSO aplica un proceso de regularización al penalizar los coeficientes de las variables de regresión, haciendo que un subconjunto de las características proporcionadas se use durante el proceso de ajuste del modelo, en lugar de usarlas todas. Al final del proceso, se realiza la selección de características, de manera que se seleccionan aquellas características cuyos coeficientes no han sido penalizados hasta que sean iguales a cero.

La capacidad de este modelo para seleccionar subconjuntos de características se basa en la estructura de las restricciones que realiza, y tiene una amplia variedad de interpretaciones en términos estadísticos geométricos y en estadística bayesiana. Si se tiene un conjunto de datos de entrada $D = (x_i, y_i)$, $i \in \{1...N\}$ donde x_i representa la i-ésima muestra, $x_i = (x_{i1}, x_{i2}, ..., x_{ip})$

UNIVERSIDAD DE MÁLAGA

son las características de los datos de entrada e y_i representa las clases del problema, la Ecuación 3.1 muestra la estimación LASSO que se minimiza dado un parámetro de afinación t y teniendo los coeficientes $\beta = (\beta_1, ..., \beta_p)$.

$$\hat{\beta}^{lasso} = \arg\min_{\beta} \sum_{i=1}^{N} \left(y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j \right)^2 \text{ sujeto a } \sum_{j=1}^{p} |\beta_j| \leq t \quad (3.1)$$

En la práctica bajo el enfoque LASSO, la ecuación anterior puede ser representada como un multiplicador de *Lagrange* (3.2).

$$\hat{\beta}^{lasso} = \arg\min_{\beta} \frac{1}{2} \sum_{i=1}^{N} \left(y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j \right)^2 + \lambda \sum_{j=1}^{p} |\beta_j| \quad (3.2)$$

El hiperparámetro λ controla la cantidad de características que retiene el modelo, que toma valores en el rango (0,1). Este hiperparámetro λ se aprende de los datos a través de un proceso de validación cruzada, y con valores λ más altos, el término de penalización tiene una mayor incidencia y más coeficientes (β_j) son iguales a cero. Solo se mantienen aquellas características x_j que tienen un mayor efecto positivo en las predicciones. Para la implementación de este método en el presente trabajo, se ha utilizado una validación cruzada de 10-folds para devolver el valor de lambda. La pérdida utilizada para la validación cruzada es un error cuadrático para modelos gaussianos, y el umbral de convergencia utilizado para el descenso de coordenadas tiene un valor de 1E-7.

3.1.2. Random Forest

El algoritmo denominado Random Forest (Breiman, 2001) es un método de aprendizaje supervisado que combina el bagging y la selección de variables aleatorias para construir un conjunto de árboles de decisión con varianza controlada. La idea de la selección aleatoria de características fue tomada por Breiman de estudios independientes sobre el subespacio aleatorio (Ho, 1998), método que hace una selección aleatoria de un subconjunto de características para aplicarla a un árbol de decisión; y sobre el reconocimiento de caracteres escritos (Amit y Geman, 1997), donde se hacía una definición de un gran número de características geométricas para posteriormente buscar, a través de una selección aleatoria de estas, la mejor división en cada nodo durante la generación de un árbol de decisión.

Random Forest opera mediante la creación de una colección de árboles de decisión, los cuales forman un bosque. Cada árbol en el bosque se construye usando la técnica de *bootstrap*, que selecciona aleatoriamente un conjunto

UNIVERSIDAD DE MÁLAGA

de datos con reemplazo y selecciona un subconjunto aleatorio de las características de entrada para realizar cada división de nodo, dejando una baja correlación entre cada árbol individual. Cuando se utiliza el modelo Random Forest como método de clasificación o regresión, a la hora de otorgar una predicción para una instancia, esta pasa como entrada de todos los árboles de decisión que componen el bosque. Cada uno de los árboles devuelve una predicción sobre la clase a la que pertenece la nueva instancia. Una vez todos han devuelto una clase, hay varios métodos posibles para obtener la predicción. Si es una tarea de clasificación, puede utilizarse el voto mayoritario simple, es decir, se toma como predicción aquella con mayor número de incidencias en las respuestas de los árboles individuales, o puede utilizarse un voto ponderado si se confía más en la respuesta de algunos árboles. Si es una tarea de regresión, se puede promediar la salida.

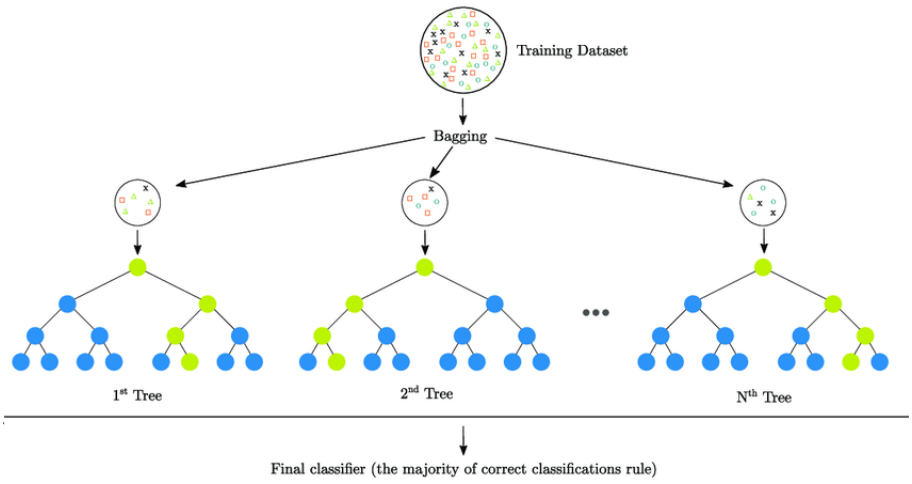

Figura 3.1: Representación gráfica del método Random Forest[1].

Aunque las principales tareas de Random Forest son la clasificación y la regresión, también se puede utilizar para la selección de características (Díaz-Uriarte y De Andres, 2006). Este método puede llevar a cabo la selección de características considerando diferentes medidas de importancia de características, por ejemplo, midiendo directamente el impacto que tiene cada característica en el porcentaje de aciertos obtenido por el modelo durante el aprendizaje supervisado. Como se mencionó anteriormente, el modelo utiliza una selección aleatoria de características de entrada para la división de cada nodo, por lo que se pueden realizar permutaciones en las características seleccionadas y medir el porcentaje de aciertos del modelo con cada una, dando mayor importancia a las características cuya permutación produce una mayor variación en esta métrica. La importancia Gini es otra medida para evaluar las características, que se puede calcular durante la construcción de

[1]Extraído de Sapountzoglou et al. (2020)

UNIVERSIDAD DE MÁLAGA

los árboles individuales. Esta medida se basa en el índice de impureza Gini (*Generalized Inequality Index*) calculado para cada división de un nodo dada una variable y ponderada por la probabilidad de alcanzar ese nodo (que se aproxima por la proporción de muestras que llegan a ese nodo). De este modo, el criterio de impureza Gini para los nodos descendientes es menor que el calculado para el nodo principal. Esta medida tiene por tanto un valor medio superior para aquellas características que se han utilizado en etapas tempranas de los diferentes árboles creados, lo que indica que su valor tiene un mayor impacto en el árbol de decisión, y por tanto la importancia de la característica es mayor. En este trabajo vamos a usar la medida de importancia Gini en lugar del impacto directo en precisión porque, aunque ambas medidas se correlacionan razonablemente bien, la importancia Gini es ligeramente mejor cuando se trata de características continuas (Archer y Kimes, 2008). Para las simulaciones numéricas realizadas, el número de árboles utilizados en el modelo Random Forest se estableció en 1000.

El método Random Forest presenta algunas limitaciones, como el posible sobreajuste en tareas de clasificación/regresión, o la imposibilidad de usar este modelo para la predicción en tiempo real, debido al enorme número de árboles que utiliza en el proceso. Aun así, presenta algunas ventajas, como son la capacidad de controlar espacios dimensionales altos, así como el manejo de un gran número de ejemplos de entrenamiento y la funcionalidad como método de selección de características.

3.2. Medidas de evaluación y validación

3.2.1. Métricas de clasificación

En la tarea de clasificación/predicción, las métricas de evaluación comparan la predicción realizada por el modelo clasificador con la verdadera pertenencia a la clase de cada instancia. Para visualizar esta evaluación, se puede graficar una matriz de confusión (Kohavi y Provost, 1998) que presenta los valores predichos por el modelo o método, en contraposición con los valores reales. De este modo se tiene que cada fila representa el número de instancias que han sido clasificadas por el modelo en una clase y cada columna representa esas instancias en la clase real a la que pertenecen en el conjunto de datos. Realmente, esta configuración puede encontrarse de forma alterna, mostrando la predicción en las columnas y las clases reales en las filas, pues solo implicaría trasponer la matriz. Cuando se aplica un problema binario, y derivado de la aplicación de las matrices de confusión en pruebas de diagnosis médicas, se tienen dos clases denominadas "negativa" y "positiva" haciendo referencia a la presencia de una enfermedad. Si se trabaja en problemas fuera del ámbito médico, por convenio se consideran como instancias positivas aquellas que pertenecen a la clase minoritaria, y

UNIVERSIDAD DE MÁLAGA

como negativas las pertenecientes a la clase mayoritaria (Japkowicz, 2013). Este convenio viene dado por la generalidad en los estudios médicos de ser minoritaria la presencia de muestras con la enfermedad (especialmente en enfermedades raras) frente a un gran número de muestras de control.

	Positiva (P)	Negativa (N)
Predicción Positiva (PP)	TP	FP
Predicción Negativa (PN)	FN	TN

Tabla 3.1: Matriz de Confusión.

En la Tabla 3.1 se puede observar una matriz de confusión para un problema binario. En esta se observan los diferentes nombres que reciben cada uno de los valores presentes. Se tienen pues los verdaderos positivos (TP), que recoge el número de instancias clasificadas correctamente como positivas; los verdaderos negativos (TN), número de instancias clasificadas correctamente como negativas; los falsos positivos (FP), número de instancias clasificadas de forma incorrecta como positivas; y los falsos negativos (FN), número de instancias clasificadas de forma incorrecta como negativas. A partir de la matriz de confusión se pueden obtener las diferentes medidas de evaluación de clasificación que se van a utilizar en la investigación:

- La capacidad predictiva, porcentaje de aciertos o exactitud (*accuracy*), indica la proporción de predicciones correctas frente al número total de instancias en el problema. Esta es la métrica más común para evaluar el rendimiento de los modelos en problemas de clasificación. Si bien presenta la desventaja de no tener en cuenta la proporción de las clases, por lo que su uso con conjuntos muy desbalanceados se encuentra restringido. Por tanto, no siempre una mejora en esta métrica es posible o deseable, frente a la mejora de otras métricas.

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{P} + \text{N}} \tag{3.3}$$

- La especificidad (*specificity*) mide el número de instancias negativas clasificadas correctamente con respecto al número total de instancias negativas. En diagnóstico médico indica la capacidad del modelo de predecir los pacientes de control.

$$\text{Specificity} = \frac{\text{TN}}{\text{TN} + \text{FP}} \tag{3.4}$$

- La sensibilidad (*sensitivity* o *recall*) mide el número de instancias positivas clasificadas correctamente con respecto al número total de instancias positivas. En diagnóstico médico indica la capacidad del modelo de

UNIVERSIDAD DE MÁLAGA

predecir los pacientes con la enfermedad, siendo pues una métrica muy importante para que los pacientes puedan recibir su tratamiento. La sensibilidad también se conoce como la tasa de detección del modelo.

$$\text{Sensitivity} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{3.5}$$

- La precisión (*precision*) indica la eficiencia del modelo clasificador para predecir una instancia como positiva. Esto se calcula como el número de verdaderos positivos entre el total de instancias clasificadas como positivas. Es importante no confundir esta métrica con la capacidad predictiva (*accuracy*).

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{3.6}$$

- La F_1 *score* se define como la media armónica de precisión (valor predictivo positivo) y sensibilidad. Esta es una una derivación de la *F-Measure* o F_β (Chinchor, 1992) en la que se toma 1 como valor del parámetro β que controla el equilibrio entre precisión y sensibilidad. Esta métrica permite una medida más confiable del desempeño del clasificador, particularmente bajo las circunstancias de problemas muy balanceados y de diagnóstico médico, donde la sensibilidad presenta una mayor importancia.

$$\text{F}_\beta = (1 + \beta^2) \cdot \frac{\text{precision} \cdot \text{sensitivity}}{(\beta^2 \cdot \text{precision}) + \text{sensitivity}} \tag{3.7}$$

$$\text{F}_1 = 2 \cdot \frac{\text{precision} \cdot \text{sensitivity}}{\text{precision} + \text{sensitivity}} = \frac{2\text{TP}}{2\text{TP} + \text{FP} + \text{FN}} \tag{3.8}$$

- El *Mathews Correlation Coefficient* (MCC) (Matthews, 1975; Baldi et al., 2000) indica el rendimiento general de un modelo clasificador para realizar predicciones correctas. En lugar de una métrica de calidad convencional para conjuntos de datos desbalanceados como F_1, se ha demostrado recientemente que es más recomendable realizar la evaluación del rendimiento en predicción mediante la métrica MCC (Chicco y Jurman, 2020). Si bien la métrica F_1 se ve alterada por la sensibilidad lograda con datos desequilibrados, para alcanzar un buen valor de MCC, el modelo clasificador debe realizar predicciones correctas tanto en la mayoría de los casos negativos como en la mayoría de los casos positivos, independientemente de sus proporciones.

$$\text{MCC} = \frac{\text{TP} \cdot \text{TN} - \text{FP} \cdot \text{FN}}{\sqrt{(\text{TP} + \text{FP}) \cdot (\text{TP} + \text{FN}) \cdot (\text{TN} + \text{FP}) \cdot (\text{TN} + \text{FN})}} \tag{3.9}$$

UNIVERSIDAD DE MÁLAGA

Para comparar los valores obtenidos por los métodos y modelos de clasificación para las diferentes métricas empleadas, usamos en diversas ocasiones la diferencia relativa (RD). Especialmente, usamos esta métrica para comparar capacidad predictiva y AUC. Esta medida (Ecuación 3.10) indica el aumento porcentual cuando se comparan los resultados obtenidos con un modelo (Result $_{model}$) frente a los resultados obtenidos como referencia (Result $_{ref}$).

$$\text{RD} = \frac{\text{Result }_{model} - \text{Result }_{ref}}{\text{Result }_{ref}} \times 100 \tag{3.10}$$

3.2.2. Área bajo la curva ROC

Una de las medidas de evaluación que se van a usar en el presente trabajo es la denominada área bajo la curva ROC, a la cual nos referiremos con el acrónimo AUC. A su vez ROC es acrónimo de *Reciever Operating Characteristic*. El análisis de esta medida procede del ámbito de la "Teoría de Detección de Señales", y fue desarrollada durante la II Guerra Mundial para el análisis de señales (Hanley y McNeil, 1982). Mediante este análisis, los operadores de radar intentaban distinguir si las imágenes que proyectaban las señales de radar pertenecían a enemigos, aliados o simplemente eran ruido.

La medida AUC es una representación gráfica de la sensibilidad, o ratio de verdaderos positivos (TPR), frente al valor de 1 – especificidad, o ratio de falsos positivos (FPR), para un modelo según se varía el umbral de discriminación. En el ámbito del aprendizaje automático, el área bajo la curva ROC proporciona una medida promedio de la bondad de los algoritmos de aprendizaje supervisado a los que se aplica (Bradley, 1997). También es usado el análisis mediante curvas ROC en el ámbito de la medicina para pruebas diagnósticas, como una medida global e independiente, por ello también se les puede denominar como curvas de rendimiento diagnóstico.

[0.5, 0.6) – Test malo	**[0.6, 0.75)** – Test regular
[0.75, 0.9) – Test bueno	**[0.9, 0.97)** – Test muy bueno
[0.97, 1) – Test excelente	

Tabla 3.2: Interpretación de AUC

El área bajo la curva posee un valor comprendido entre 0.5 y 1 para clasificación binaria. En esta medida, el valor 1 representa un valor de prueba perfecto y el valor 0.5 significa que la prueba no tiene capacidad discriminatoria, es decir, es igual que el azar. Al comparar modelos o algoritmos de aprendizaje supervisado, el que presente una mayor área bajo la curva

UNIVERSIDAD DE MÁLAGA

es considerado como superior. Para interpretar las curvas ROC, se pueden observar en la Tabla 3.2 los intervalos establecidos para los valores de AUC.

La métrica AUC como medida para la bondad de algoritmos de aprendizaje, supone una mejora respecto a la capacidad predictiva (*accuracy*), ya que este último presenta una serie de problemas debido principalmente a que, en primer lugar, no es sensible a la distribución de las clases, y en segundo lugar, no tiene en cuenta la evaluación del coste/beneficio cuando el modelo se equivoca en un sentido u otro (falsos positivos o negativos).

3.2.3. Análisis de componentes principales

El análisis de componentes principales (PCA) (Pearson, 1901; Hotelling, 1933) constituye la base del análisis de datos multivariados basado en métodos de proyección. El PCA es una técnica multivariante que analiza una tabla de datos en la que las instancias se describen mediante varias características dependientes cuantitativas interrelacionadas. El objetivo principal de este análisis es extraer información importante de los datos, representarla como un conjunto de nuevas características ortogonales llamadas componentes principales, y poder mostrar una representación de las instancias y de las características que presentan en un gráfico de dispersión (Jolliffe y Cadima, 2016).

En primer lugar, hay que tener en cuenta que un conjunto de datos se representa por n instancias que son vectores p dimensionales $x_1, ..., x_p$, es decir, una matriz de datos X de tamaño $n \times p$, cuya i-ésima columna es el vector x_i de observaciones sobre la i-ésima característica. PCA realiza una búsqueda de la combinación lineal de las columnas de la matriz X que presenta una máxima varianza. Tales combinaciones lineales vienen dadas por la multiplicación de un vector de constantes $a = a_1, a_2, ..., a_p$ por la matriz X (Ecuación 3.11). La varianza de cualquier combinación lineal viene dada por $var(Xa) = a^T \sigma a$, donde σ es la matriz de covarianza asociada con el conjunto de datos (a^T es la transpuesta).

$$\sum_{i=1}^{n} a_i x_i = Xa \tag{3.11}$$

La matriz de covarianzas es una matriz simétrica que tiene como entradas las covarianzas asociadas con todos los pares posibles de las características iniciales, lo que permite identificar las correlaciones entre las características y poder eliminar información redundante (alta correlación). Para calcular la covarianza de dos características j y k, se sigue la Ecuación 3.12, donde se resta a cada valor la media de la característica a la que pertenece.

$$\sigma_{jk} = \frac{1}{n-1} \sum_{i=1}^{n} (x_{ij} - x_j)(x_{ik} - x_k) \tag{3.12}$$

UNIVERSIDAD DE MÁLAGA

Por tanto, identificar la combinación lineal con varianza máxima es equivalente a obtener un vector p-dimensional a que maximiza la forma cuadrática $a^T Sa$. El problema equivale a maximizar $a^T Sa - \lambda(a^T a - 1)$, donde λ es un multiplicador de *Lagrange*. Por lo tanto, a es un *eigenvector* y λ el *eigenvalue* correspondiente de la matriz de covarianza σ. Al ser σ una matriz simétrica $p \times p$, tiene exactamente p *eigenvalues*, $\lambda_k (k = 1, ..., p)$, y sus correspondientes *eigenvectors*.

Los *eigenvectors* de la matriz de covarianza son en realidad las direcciones de los ejes donde hay más varianza (más información), y también se denominan componentes principales (PC). Los *eigenvalue* son los coeficientes adjuntos a los *eigenvectors*, que indican la cantidad de varianza que aporta cada componente principal. De este modo, las componentes principales se pueden ordenar según la magnitud de varianza explicada, siendo la primera componente (PC1) el *eigenvector* con el *eigenvalue* asociado más alto. En la terminología de PCA, los elementos de los *eigenvectors* a_k se denominan comúnmente PC *loadings*, mientras que los elementos de las combinaciones lineales Xa_k se denominan PC *scores*, ya que son los valores que cada instancia presenta en una componente principal.

Si se quiere utilizar el PCA para reducir la dimensionalidad del conjunto de datos, se puede analizar la cantidad de varianza explicada por cada PC. Mediante la varianza explicada acumulada, es decir, la suma de las varianzas por cada componente en orden descendente, se puede seleccionar el número óptimo de PCs a utilizar, descartando los de menor importancia (de bajos valores propios). Con las PCs seleccionadas se forma una matriz de vectores denominada vector de características, que es simplemente una matriz que tiene como columnas los *eigenvectors* de los componentes que se deciden mantener. Para reorientar los datos de los ejes originales a los representados por los componentes principales, se multiplica la transposición del conjunto de datos original por la transposición del vector de características.

3.2.4. Validación Cruzada

Como método de validación para los algoritmos de aprendizaje supervisado se utiliza la estrategia de la validación cruzada (*K-fold Cross-Validation*). Esta estrategia consiste en la creación de k particiones (folds) del conjunto de datos disponible en el problema tratado, en conjuntos disjuntos. La división comúnmente aplicada agrupa $k - 2$ conjuntos para el entrenamiento del algoritmo de aprendizaje, 1 conjunto como resultados de validación, y el restante como conjunto de test, para comprobar su capacidad de generalización. Se repite el proceso k iteraciones, cambiando en cada iteración los conjuntos utilizados en cada rol, y la medida de evaluación se deriva de las obtenidas en cada iteración. Otra posibilidad radica en realizar únicamente una división en conjunto de entrenamiento y conjunto de test, o validación.

UNIVERSIDAD DE MÁLAGA

Es necesario tener en cuenta que la clave de esta estrategia es la transmisión de los roles, existiendo la posibilidad de variar la proporción de los conjuntos, pudiendo tomar más particiones para los conjuntos de validación o test.

De este modo, la validación cruzada tiene la ventaja frente a la validación simple, en la cual se realiza una sola división del conjunto de datos, de no presentar una fuerte dependencia de la posible aleatoriedad al elegir la partición, gracias a la división en k particiones y la reutilización de estas. También se soluciona el problema de la representatividad de las muestras mediante el uso de la estratificación, la división del conjunto en grupos homogéneos respecto a la clase de la población.

Sin embargo, la validación cruzada presenta una desventaja cuando se aplica al análisis de conjuntos con un número reducido de instancias. Debido a la división realizada, en un conjunto de datos pequeño existe una posible pérdida de información en los conjuntos tomados para entrenamiento, que no puede ser recuperada. La pregunta clave es determinar el número de particiones en las que se debe dividir el conjunto de datos para obtener el rendimiento óptimo. Intentar averiguar este número implica hacer numerosas pruebas con el conjunto de datos, no obstante, la validación cruzada con 10 divisiones es utilizada generalmente en el análisis de algoritmos como medida fiable y con una buena complejidad en tiempo real.

De esta forma, un procedimiento estratificado de validación cruzada con 10-folds, es concretamente el método usado en las distintas investigaciones recogidas en la presente tesis. Si bien, en cada estudio específico realizado, se ha variado la proporción y división dependiendo de las características del problema y la experimentación desarrollada. Un proceso con división en entrenamiento, validación y test se observa gráficamente en la Figura 3.2.

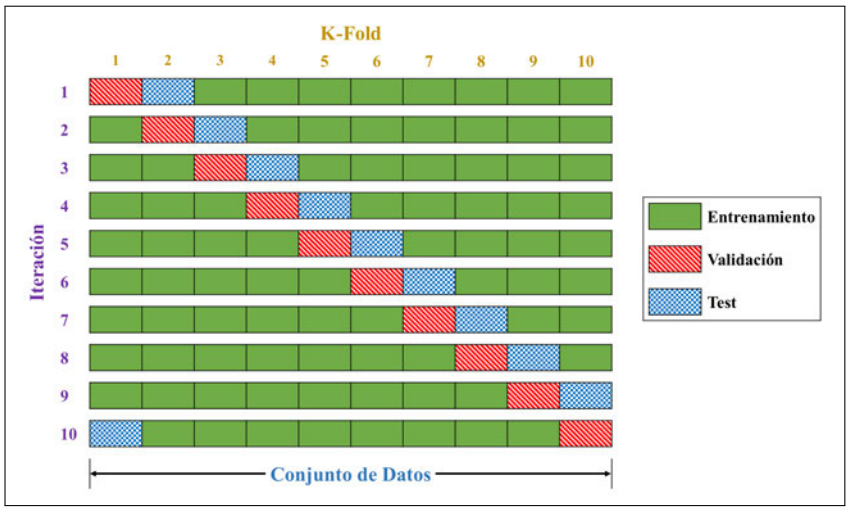

Figura 3.2: Validación cruzada con 10-folds.

UNIVERSIDAD DE MÁLAGA

3.3. Modelos clasificadores auxiliares

El aprendizaje automático o aprendizaje computacional, denominado en la literatura inglesa *Machine Learning*, tiene como objetivo el desarrollo de modelos capaces de realizar tareas de forma automática y eficiente gracias a la capacidad de los mismos de aprender, entendiendo por aprender el captar comportamientos y patrones a través de la experiencia. Al seleccionar un algoritmo de aprendizaje automático para efectuar una tarea u obtener cierto conocimiento, se debe tomar en cuenta el tipo de experiencias del que se dispone, además del conocimiento que se quiere extraer. En el presente trabajo de tesis nos enfocamos especialmente en la tarea supervisada de clasificación. A este respecto, la tarea de clasificación consiste en la construcción de modelos concisos que representen la distribución del atributo dependiente, denominado clase (nominal o numérica), en función de los atributos predictores, denominados atributos o características, a partir de unas experiencias o instancias previas etiquetadas correctamente. El objetivo del aprendizaje supervisado es justamente aprender una función que sea capaz de realizar dicha predicción, obteniendo la salida que le corresponde a una nueva instancia a partir del conjunto de datos de entrada, denominado conjunto de datos de entrenamiento. Los modelos utilizados tienen que ser capaces de generalizar a partir de los datos de entrada, para tener un menor error ante situaciones no vistas previamente.

Entre los modelos de aprendizaje usados se encuentran las redes neuronales profundas, descritas en la sección 2.2.1, los Random Forest, descritos en la sección 3.1.2, la regresión logística y los sistemas de máquinas de vectores de soporte. Estos dos últimos modelos de aprendizaje automático van a ser descritos a continuación.

3.3.1. Regresión logística

La regresión logística es un modelo estadístico que se utiliza como algoritmo de clasificación. Este algoritmo es una extensión del modelo de regresión lineal y, a diferencia de la regresión lineal que genera valores numéricos continuos, la regresión logística transforma su salida utilizando la función sigmoidea logística para devolver un valor de probabilidad que luego se puede asignar a dos o más clases discretas. La regresión logística como modelo estadístico general fue originalmente desarrollada y popularizada principalmente por Berkson (1944) en problemas binarios. La regresión logística multinomial es una extensión simple de la regresión binaria que permite más de dos clases. Al igual que la regresión logística binaria, la regresión multinomial utiliza la estimación de máxima verosimilitud (*likelihood*) para evaluar la probabilidad de pertenecer a cada clase, y la pérdida minimizada es el ajuste de pérdida de entropía cruzada multinomial en toda la distribución de probabilidad.

UNIVERSIDAD DE MÁLAGA

Teniendo en cuenta lo anterior, la regresión multinomial puede utilizar diferentes algoritmos para el problema de optimización. Los más utilizados son los basados en el método de Newton (Steihaug, 1983), que utiliza la aproximación cuadrática para realizar la minimización, es decir, utiliza las primeras y segundas derivadas parciales mediante la matriz Hessiana (una matriz cuadrada de derivadas parciales de segundo orden). Su mayor inconveniente es el costo computacional asociado al método debido a los cálculos de segundas derivadas parciales en la matriz Hessiana, especialmente cuando las dimensiones son grandes. La variación Newton-CG se diseñó para para dar solución a este inconveniente (Nocedal y Wright, 2006). Concretamente, el método de Newton-CG aplica el método de gradiente conjugado (CG) a la aproximación de serie de Taylor de segundo orden de la función de optimización. Dado que el método CG solo requiere de los productos vectoriales Hessianos, este algoritmo requiere de menos recursos computacionales. Un método análogo al método de Newton, es el algoritmo de Broyden-Fletcher-Goldfarb-Shanno de memoria limitada (L-BFGS) (Byrd et al., 2016). La diferencia que presenta L-BFGS respecto al método de Newton, es el cálculo de la aproximación de la matriz Hessiana inversa a partir del conjunto de pares de corrección, realizando actualizaciones especificadas por evaluaciones de gradiente. El término de memoria limitada significa que el método almacena solo unos pocos vectores que representan la aproximación implícitamente. Su mayor desventaja es que, en ocasiones, es posible que el modelo no converja.

Un método innovador en la optimización estocástica es el denominado gradiente medio estocástico o SAG (Schmidt et al., 2017). Este método optimiza la suma de un número finito de funciones convexas suaves, y al igual que los métodos de gradiente estocástico (SG), el costo asociado a cada iteración del método SAG es independiente del número de términos en la suma. Sin embargo, este método incorpora una memoria para los valores de gradiente obtenidos con anterioridad, consiguiendo que el método presente una tasa de convergencia más rápida que los métodos de gradiente estocástico convencionales. Este hecho hace que SAG sea más rápido que otros algoritmos si se utilizan grandes conjuntos de datos.

Una variante de SAG es el algoritmo SAGA (Defazio et al., 2014), que admite la regularización L^1, mientras que el resto de algoritmos aquí estudiados solo permiten regularización L^2. Sin embargo, dado que en las investigaciones realizadas ya empleamos selección de características con los métodos LASSO y Random Forest, descritos en la sección 3.1, no es necesario el uso adicional de la regularización L^1 en el modelo de regresión logística. De este modo, el algoritmo SAGA no proporciona una ventaja significativa. Además, tanto SAG como SAGA están enfocados para su empleo con grandes conjuntos de datos, mientras que el objetivo estudiado es justo el contrario, aplicaciones con conjunto de datos reducidos.

UNIVERSIDAD DE MÁLAGA

Para finalizar, una de las ventajas que presenta la regresión logística frente al resto de modelos de aprendizaje supervisado estudiados es la reducida complejidad que presenta, especialmente en lo que a tiempo de ejecución se refiere. Si bien las redes neuronales que pertenecen al DL, pues presentan miles de neuronas interconectadas, pueden requerir de un tiempo para realizar el entrenamiento de horas si el conjunto de datos es extenso en número de muestras o atributos, la regresión logística puede no necesitar más de un par de minutos para el mismo proceso. Sin embargo, la reducida complejidad de implementación que presenta este modelo hace que presente una clara limitación en rendimiento frente al resto de modelos, viéndose afectado por el sobreajuste de los datos y el desequilibrio presente en los mismos.

3.3.2. Máquina de vectores de soporte

Una Máquina de Soporte Vectorial, cuyo acrónimo en la literatura inglesa es SVM, son algoritmos de aprendizaje supervisado cuyo objetivo es la obtención de un hiperplano que separe los datos de entrada mediante el mapeo de los puntos de entrada a un espacio de características de una dimensión mayor, de forma que se maximice el margen entre las diferentes clases de este espacio y provocando un agrupamiento de los datos (Cortes y Vapnik, 1995). Cuando entran nuevas experiencias, estas son relacionadas con el modelo y son clasificadas en la clase correspondiente según la proximidad con los agrupamientos realizados.

Las SVMs fueron diseñadas con el propósito de resolver problemas de clasificación binaria, si bien actualmente son utilizadas para resolver otros problemas diferentes, como pueden ser la regresión, la clasificación de múltiples clases o el agrupamiento. Dentro de la clasificación podrían situarse como clasificadores lineales, ya que, al igual que las redes neuronales artificiales, crean modelos mediante la inducción de hiperplanos o separadores lineales de los datos. Estos hiperplanos o separadores inducidos por las SVMs son seleccionados creando una separación equidistante entre los ejemplos que se encuentren más cercanos de cada clase. De este modo se logra que entre ellos haya un margen máximo a cada lado del hiperplano. En esta inducción de los hiperplanos solo son considerados aquellos ejemplos que son denominados vectores de soporte, ejemplos de cada clase y presentes en el conjunto de datos de entrenamiento, que se encuentran en la divisoria de los márgenes de separación.

De una manera más formal, dado un conjunto de datos de entrenamiento de pares $(x_i, y_i) \mid i = 1, ..., l$, donde $x_i \in R^N$ e $y_i \in 1, -1^l$ las máquinas de vectores de soporte enfocan un problema de minimización (3.13) en el cual los vectores de entrenamiento x_i son mapeados a un espacio de dimensiones superior, e incluso infinito, por la función ϕ. Para este espacio dimensional superior, las SVMs tratan de localizar una separación de hiperplano con el

UNIVERSIDAD DE MÁLAGA

máximo margen posible. La constante $C > 0$ es denominada parámetro de penalización del término de error.

$$\min_{w,b,\xi} \frac{1}{2} w^T w + C \sum_{i=1}^{l} \xi_i \text{ sujeto a } y_i(w^T \phi(x_i) + b) \geq 1 - \xi_i \xi_i \geq 0 \tag{3.13}$$

Las SVMs precisan de las funciones kernel para poder solucionar problemas que no son linealmente separables, y como hemos mencionado, llevar el problema a un espacio dimensional mayor. Una forma de designar una función kernel es $K(x_i, x_j) \equiv \phi(x_i)^T \phi(x_j)$. En este ámbito hay investigaciones que proponen diferentes funciones para mejorar el rendimiento de las SVMs, así como nuevas estrategias con el mismo propósito. Para este trabajo presentamos cuatro funciones kernel básicas, muy presentes en la literatura, y que pueden ser seleccionadas en el paquete software *sklearn* (Pedregosa et al., 2011) de Python:

- Lineal: $K(x_i, x_j) = x_i^T x_j$.
- Polinómico: $K(x_i, x_j) = (\gamma x_i^T x_j + r)^d, \gamma > 0$, donde γ, r y d son parámetros kernel.
- Gaussian radial basis function: $K(x_i, x_j) = \exp(-\gamma \|x_i - x_j\|^2), \gamma > 0$.
- Sigmoide: $K(x_i, x_j) = \tanh(-\gamma x_i^T x_j + r)$.

Por último, mencionar que una de las ventajas que presenta este modelo es una buena capacidad de generalización, es decir, no es un modelo propenso al sobreajuste del conjunto de datos de entrenamiento. Sin embargo, también muestran algunas limitaciones en el tratamiento de conjuntos de datos discretos, y limitaciones acuciantes en tiempo y espacio de cómputo que hacen que el entrenamiento de conjuntos de datos extensos con millones de vectores de soporte sea irresoluble (Burges, 1998).

UNIVERSIDAD DE MÁLAGA

UNIVERSIDAD
DE MÁLAGA

Parte II

Métodos de Aumento de Datos

UNIVERSIDAD
DE MÁLAGA

UNIVERSIDAD
DE MÁLAGA

Capítulo 4

Métodos de Adición de Ruido

Resumen: Para realizar aumento de datos existen algunas técnicas y métodos convencionales con enfoques que pueden considerarse simples, como son la inversión, la rotación, el recorte, el desplazamiento, el remuestreo, o la inyección/adición de ruido. Si bien el planteamiento de la adición de ruido es sencillo, el diseño y aplicación de un procedimiento basado en la adición de ruido puede extenderse, complicando su funcionamiento para su adaptación sobre diferentes conjuntos de datos, y con la capacidad de obtener resultados efectivos. Aquí, se proponen tres métodos de inyección de ruido, dos basados en adición aleatoria de ruido gaussiano y un método que crea nuevas muestras ajustando el nivel de ruido para variables de entrada individuales clasificadas previamente por su nivel de relevancia. Los resultados experimentales se analizan utilizando diferentes conjuntos de datos de referencia comparando el uso de los datos originales y aumentados en el entrenamiento supervisado en arquitecturas de aprendizaje profundo. Estos demuestran un buen desempeño en mejora de la predicción, pero un mal desempeño en términos de velocidad.

4.1. Método de adición de ruido variable FNA

El primer método de adición de ruido desarrollado para crear nuevas muestras (aumento de datos) mediante un procedimiento progresivo e incremental fue denominado como método de *Forward Noise Adjustment* (FNA) (Moreno-Barea et al., 2018). La motivación para el desarrollo de este método se encuentra en la diferencia de importancia que presentan los atributos/-características que componen los conjuntos de datos a la hora de realizar un proceso de clasificación. Conforme a las diferencias en importancia, hay características que de forma práctica no afectan al proceso de clasificación,

UNIVERSIDAD DE MÁLAGA

bien porque no varían o debido a que no son relevantes. Estas no deberían pues ser consideradas a la hora de aplicar ruido, ya que una variación podría desvirtuar la predicción realizada o simplemente ser irrelevante. Además, se quería tener en cuenta que las diferencias en importancia hacen que el nivel de ruido óptimo para cada característica sea a su vez distinto. Así, este método realiza una optimización de los niveles de ruido ajustados. Para realizar la medición de la importancia de las características individuales de las muestras, se ha utilizado Random Forest. Este método está descrito en el capítulo anterior, en la sección 3.1.2 de métodos de selección de características.

El método FNA se encuentra descrito en el pseudocódigo que se muestra en el Algoritmo 3. Como entrada, el algoritmo recibe el conjunto de datos (dataset) sobre el que se quiere aplicar el aumento de datos y los diferentes niveles de ruido que se utilizan (ruido), cuyos valores se encuentran entre 0.1 y 1.0. En primer lugar en la experimentación se realiza una división de los datos utilizando un procedimiento de validación cruzada estratificado de 10 folds, de modo que 8 folds son utilizados para el entrenamiento, 1 fold se usa para elegir el mejor nivel de ruido (validación) y el fold restante se usará para probar el rendimiento en clasificación ([Train, Val, Test] respectivamente). El procedimiento de validación cruzada está diseñado de tal manera que el conjunto de test no es utilizado durante el algoritmo para elegir el nivel óptimo de ruido, con el fin de que los resultados de test sean totalmente independientes y evitar cualquier efecto de sobreajuste. Los datos aumentados (Aug) que se crean a partir de las muestras originales Train mediante la adición de ruido, son inicializadas primero en la línea 3 del algoritmo. Antes de realizar el aumento, se aplica un método de selección de características, en este caso Random Forest y la importancia Gini, para seleccionar y ordenar las características de los datos de entrada Attr según su relevancia (ordenadas en orden descendente).

En este momento, para cada característica y nivel de ruido, el algoritmo recorre cada uno de los folds, agregando ruido a las muestras de Aug en la característica $Attr_a$ considerando el nivel de ruido $noise_n$. Los datos ruidosos generados se concatenan con los datos de entrenamiento originales $Train_i$, dando lugar a un conjunto de entrenamiento aumentado. Con el conjunto aumentado se entrena un modelo clasificador, en este caso se utilizó una red neuronal descrita más adelante en la sección 4.3.1. Una vez entrenado, se obtienen los valores de la métrica F_1 ($[f1_i^V, f1_i^T]$) para los conjuntos de validación y test correspondientes (Val_i y $Test_i$). Cuando se prueban todos los folds para el grado de ruido $noise_n$, la medida F_1 asociada a esta prueba es el promedio de todos los folds. Cuando se prueban todos los niveles de ruido para la característica $Attr_a$, la medida F_1 que tiene asociada esta prueba ($f1_a^V$) es la medida F_1 asociada con el nivel de ruido que maximiza el valor de validación $f1_{i:1,\ldots,N}^V$ registrado. Si $f1_a^V$ es mayor o igual que el mejor valor obtenido hasta este momento, entonces el algoritmo crea nuevas muestras

UNIVERSIDAD DE MÁLAGA

partiendo del conjunto Aug agregando el nivel de ruido que ha maximizado la métrica ($ruido_{\text{máx}}$) a la característica correspondiente. Las nuevas muestras reemplazan a las presentes en el conjunto Aug. Es importante tener en cuenta que Aug no cambia dentro de los bucles internos, solo en este punto. Los resultados almacenados para este atributo a son $ruido_{\text{máx}}$ y la medida de prueba F_1 asociada a él. Cuando se prueban todos los atributos clasificados, el algoritmo devuelve al final los resultados obtenidos para cada atributo: el nivel óptimo de ruido encontrado utilizando los folds de validación y la medida F_1 promediada en sus folds de test correspondientes.

Algorithm 3 Forward noise adjustment algorithm (FNA)

1: **procedure** FNA($dataset, noise\{1..N\}$)
2: $[Train, Val, Test]\{1.,10\} \Leftarrow$ StratifiedSplit($dataset$)
3: $Aug \Leftarrow Train$
4: $[Attr]\{1..P\} \Leftarrow$ VariablesRank($dataset$)
5: **for** $a = 1 \rightarrow P$ **do** ▷ loop over features
6: **for** $n = 1 \rightarrow N$ **do** ▷ loop over noise levels
7: **for** $i = 1 \rightarrow 10$ **do** ▷ loop over folds
8: $T_i \Leftarrow$ AddNoise($Aug_i, Attr_a, noise_n$)
9: $T_i \Leftarrow$ Concatenate($Train_i, T_i$)
10: $[f1_i^V, f1_i^T] \Leftarrow$ Classifier($T_i, Val_i, Test_i$)
11: **end do**
12: $f1_n^V \Leftarrow$ mean($f1^V$)
13: $f1_n^T \Leftarrow$ mean($f1^T$)
14: **end do**
15: $f1_a^V \Leftarrow \max(f1_{i:1,\ldots,N}^V)$
16: $noise_{\text{máx}} \Leftarrow \arg\max(f1_{i:1,\ldots,N}^V)$
17: **if** $f1_a^V >= f1_{best}$ **then**
18: $Aug \Leftarrow$ AddNoise($Aug, Attr_a, noise_{\text{máx}}$)
19: $f1_{best} \Leftarrow f1_a^V$
20: **end if**
21: Results$_a \Leftarrow [noise_{\text{máx}}, f1_{noise_{\text{máx}}}^T]$
22: **end do**
23: **return** Results

Un esquema del procedimiento de ajuste de ruido FNA también se muestra gráficamente en la Fig. 4.1. En la figura se representa usando cajas p el ajuste del nivel de ruido correspondiente a cada una de las características de entrada p seleccionadas del conjunto de datos que se aumentará. Como se mencionó anteriormente, primero las variables se ordenan utilizando un método de selección de características, y el nivel adecuado de ruido se ajusta en un procedimiento progresivo hacia adelante a partir de la característica más

UNIVERSIDAD DE MÁLAGA

relevante. Dentro de cada recuadro, se utiliza un procedimiento de validación cruzada de diez folds, en el que se utilizan 8 folds para el entrenamiento, uno para la validación que es utilizado para realizar el ajuste del nivel de ruido, y el restante es para obtener la estimación del rendimiento de todo el procedimiento (test).

Así, en el primer cuadro de la figura, indicado por X_1, se prueban los diferentes niveles de ruido ($ruido_i$, $i = 1$, $ldots, N$) solo para la primera variable de entrada clasificada como la más importante en la tarea de clasificación. Utilizando el pliegue de validación, se elige el nivel adecuado de ruido de acuerdo con el mejor valor de validación obtenido (proceso indicado por la ecuación que se muestra fuera del cuadro de la derecha). Una vez elegido el nivel de ruido adecuado para la primera variable, el proceso continúa con las restantes probando nuevamente diferentes niveles de ruido para la variable actual, pero utilizando el valor de nivel de ruido elegido obtenido previamente para las variables consideradas más relevantes.

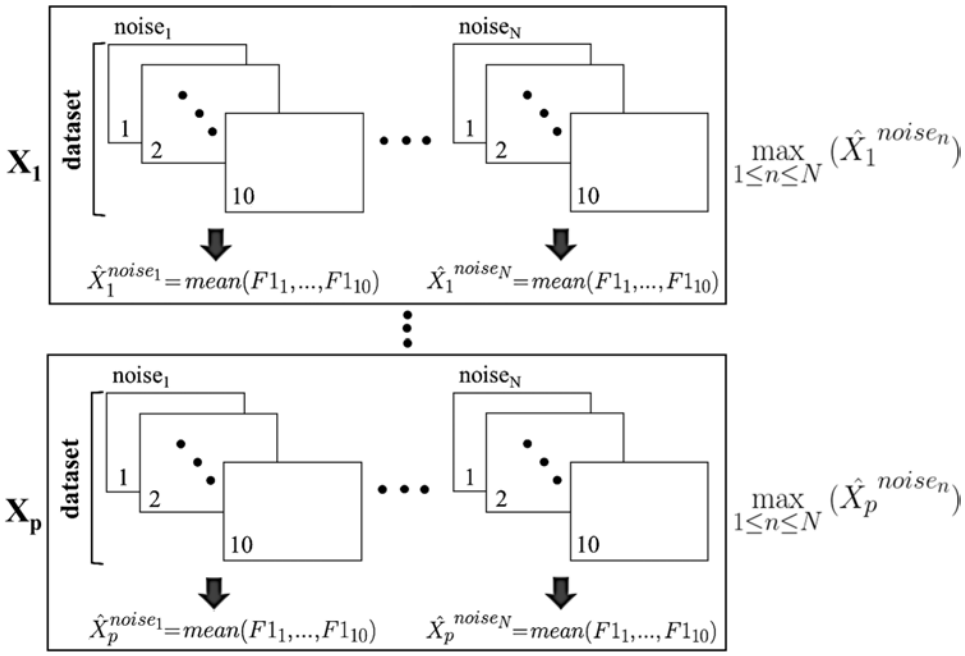

Figura 4.1: Esquema del procedimiento FNA propuesto para la estimación del nivel óptimo de ruido.

Una ventaja del método propuesto es que puede ser aplicado con diferentes métodos de selección de características y modelos de clasificación. En el pseudocódigo del algoritmo están representados como VariablesRank y Classifier respectivamente. De esta forma, se podrían realizar otras pruebas utilizando el método LASSO como selector y un árbol de decisión como clasificador. Es totalmente modular, por lo que existe independencia con el

UNIVERSIDAD DE MÁLAGA

procedimiento de ajuste de ruido. También podría ser tomada en cuenta cualquier otra métrica de evaluación del rendimiento, necesitando a lo sumo un cambio en la elección del nivel de ruido si se desea minimizar la métrica, por ejemplo, si se elige el error cuadrático medio.

4.2. Métodos de adición de ruido gaussiano

Debido al enorme tiempo de cómputo que precisa la ejecución del procedimiento de ajuste de ruido (FNA) descrito en la sección anterior, fue necesario implementar otros métodos de adición de ruido para su comparación con otros modelos de aumento de datos. De este modo, se desarrollaron dos métodos más simples y que requieren de un tiempo de cómputo significativamente menor. De estos, el primer método de adición de ruido diseñado realiza un aumento en el número de muestras disponibles en base a una selección aleatoria de las muestras de "entrenamiento". Para cada muestra seleccionada se realiza una copia, y de esta se modifican aleatoriamente un 25 % de las características.

$$\tilde{x} = \min(\text{MAX_Val}, \max(\text{MIN_Val}, x + \text{RND}(-\sigma, \sigma))) \tag{4.1}$$

La ecuación 4.1 describe matemáticamente el proceso de obtener un nuevo valor para la característica $\tilde{x}$ a partir del valor original x. Si se elige el atributo para modificarlo, el ruido que surge de una distribución normal (gaussiana) aleatoria (denotado "RND" en la ecuación 4.1) se agrega al valor original. Aquí hay que tener en consideración que el valor de la desviación estándar (σ) de la distribución normal aleatoria depende de las características del conjunto de datos a aplicar. De esta forma, se aplica un valor de $\sigma = 0.2$ cuando se aplica a conjuntos de datos cuyos valores se encuentran normalizados entre $[0, 1]$, y un valor de $\sigma = 1.0$ en cualquier otro caso. Estos valores de desviación estándar han resultado suficientes para crear muestras que no se alejan demasiado del espacio de muestras original. Se realiza además un control de los valores "ruidosos" resultantes para que no excedan los límites establecidos para su característica. De esta manera, el nuevo valor debe ser mayor que el valor mínimo presente en la característica (MIN_Val), un proceso que establece un límite inferior. El valor también debe ser menor que el valor máximo presente en la característica (MAX_Val) que establece un límite superior. Gracias a este control se logra además que el nuevo valor no se vea excesivamente afectado por preprocesamientos realizados en los datos, como normalizaciones, transformaciones logarítmicas o escalados.

Este método de adición de ruido está descrito adicionalmente en el Algoritmo 4. En él se puede observar todo el proceso descrito anteriormente. Como entrada recibe el conjunto a aumentar, que siempre debe ser el conjunto de entrenamiento, y el número de muestras que se desean generar ($numG$).

UNIVERSIDAD DE MÁLAGA

El proceso incluye un bucle para controlar el número de muestras generadas y dentro de este se realiza la selección aleatoria de muestras, lo que permite que una misma muestra sea utilizada para generar varias muestras sintéticas diferentes. Una vez seleccionada y copiada, para no sobrescribirla, se itera sobre las características del conjunto de datos, pero solo se modifica la característica si se obtiene un valor menor de 0.25 de una distribución uniforme aleatoria (RUD), lo que representa un 25 %. Si se modifica, se aplica la ecuación 4.1 y se remplaza en la muestra con el nuevo valor. Se repite con todas las características, y se guarda la muestra generada para su posterior salida, una vez se hayan generado las *numG* muestras.

Algorithm 4 Addition of Noise

1: **procedure** NOISE($trainset, numG$)
2: **for** $i = 1 \rightarrow numG$ **do** ▷ loop over nº samples to generate
3: $sample_x \Leftarrow \text{randomSel}(trainset)$
4: $sample_\tilde{x} \Leftarrow sample_x$
5: **for** $a = 1 \rightarrow P$ **do** ▷ loop over features
6: $aux \Leftarrow \text{RUD}(0, 1)$
7: **if** $aux <= 0{,}25$ **then**
8: $x \Leftarrow sample_x_a$
9: $\tilde{x} = \min(\text{MAX_Val}_a, \max(\text{MIN_Val}_a, x + \text{RND}(-\sigma, \sigma)))$
10: $sample_\tilde{x} \Leftarrow \text{replace}(a, \tilde{x})$
11: **end do**
12: $samplesGen \Leftarrow \text{add}(sample_\tilde{x})$
13: **end do**
14: **return** $samplesGen$

El segundo método de adición de ruido gaussiano diseñado es una variación del método anterior. Este método, abreviado durante el presente trabajo como "Noise Bal", se diferencia del primer método de adición de ruido estándar en que realiza la selección aleatoria solo de las muestras de entrenamiento de la clase minoritaria. Por lo tanto, solo se crean muestras sintéticas que pertenecen a esta clase. Esto lo convierte en un método de sobremuestreo, pero a diferencia de los métodos basados en SMOTE, en lugar de crear un número de muestras suficientes para balancear perfectamente el conjunto de datos, el método diseñado puede crear el número de muestras que se deseen. El resto del método sigue el mismo proceso que el enfoque de adición de ruido estándar descrito en el Algoritmo 4.

Si bien ambos métodos han sido descritos en el presente capítulo, los resultados de aplicación de estos para realizar DA en diferentes problemas serán mostrados y analizados en los capítulos 5, 6, 7, 8 y 9. Este hecho se debe a que fueron aplicados en otro punto de la investigación, con el objetivo de servir de comparación respecto a otros modelos de DA más novedosos.

UNIVERSIDAD DE MÁLAGA

4.3. Resultados de aplicación con el algoritmo FNA

4.3.1. Arquitectura del modelo de clasificación

Tal como se comentó en la introducción de este trabajo de tesis, un objetivo es mejorar el rendimiento de los modelos clasificadores pertenecientes al aprendizaje profundo mediante la aplicación de DA. Por ello, para comprobar el efecto de los métodos de DA desarrollados en los conjuntos de referencia, se hizo uso de una red neuronal feedforward. La arquitectura de la red utilizada presenta 3 capas ocultas, con 32 neuronas en las capas ocultas primera y tercera y 64 en la segunda capa oculta. La Figura 4.2 muestra un esquema de la arquitectura de la red. Esta fue obtenida mediante un proceso de optimización para validación, acotando el rango entre 2 y 3 capas ocultas, y 16, 32, 64 y 128 neuronas para cada capa. La intención que llevo a probar con esta cantidad de capas y neuronas era experimentar con una red neuronal que podría considerarse simple, sin tener que lidiar con gran cantidad de capas intermedias y miles de neuronas debido a las limitaciones en capacidad de cómputo disponible, pero que podría considerarse dentro de la categoría de modelos de DL. Mencionar que en el momento de la búsqueda de una arquitectura no se disponía aún de procesamiento gráfico (GPU). Las neuronas de las capas ocultas utilizan la función de activación Leaky ReLU (Maas et al., 2013), más estable en tareas de clasificación que la función ReLU (Xu et al., 2015). El modelo también presenta 2 neuronas de salida para clasificar en controles y casos, dado que en esta primera experimentación se aplicaron únicamente conjuntos de entrada con clasificación binaria, y se utiliza la función sigmoide como función de activación en estas neuronas.

Como técnica de regularización para evitar el sobreajuste de la red se utiliza una combinación de dropout (Srivastava et al., 2014) y la norma L^2 (*weight decay*), una combinación que se utiliza con frecuencia en la literatura. La tasa de abandono utilizada en las capas ocultas del clasificador 0.4, que corresponde a la probabilidad de que se abandone una neurona. Debido a la falta de capacidad de cómputo comentada para la búsqueda de una arquitectura, el parámetro de dropout no pudo ser optimizado, seleccionando un valor suficiente para lograr un rendimiento aceptable según la literatura (Baldi y Sadowski, 2013; Wang y Manning, 2013). Para la experimentación final si se pudo disponer de una GPU NVIDIA ® Titan Xp, si bien la repetición del proceso de búsqueda y optimización de la arquitectura requería de demasiado tiempo adicional de cómputo. Como algoritmo de optimización con tasa de aprendizaje adaptativo se utiliza el algoritmo Adam (Kingma y Ba, 2014). Los valores usados en la parametrización del algoritmo de Adam son 2e-4 como tasa de aprendizaje, 0.5 para la tasa de caída exponencial para las estimaciones del primer momento, 0.9 para la tasa de caída exponencial para el segundo momento y 1e-8 para ϵ ($\hat{\epsilon}$ en el artículo original (Kingma y Ba, 2014)).

UNIVERSIDAD DE MÁLAGA

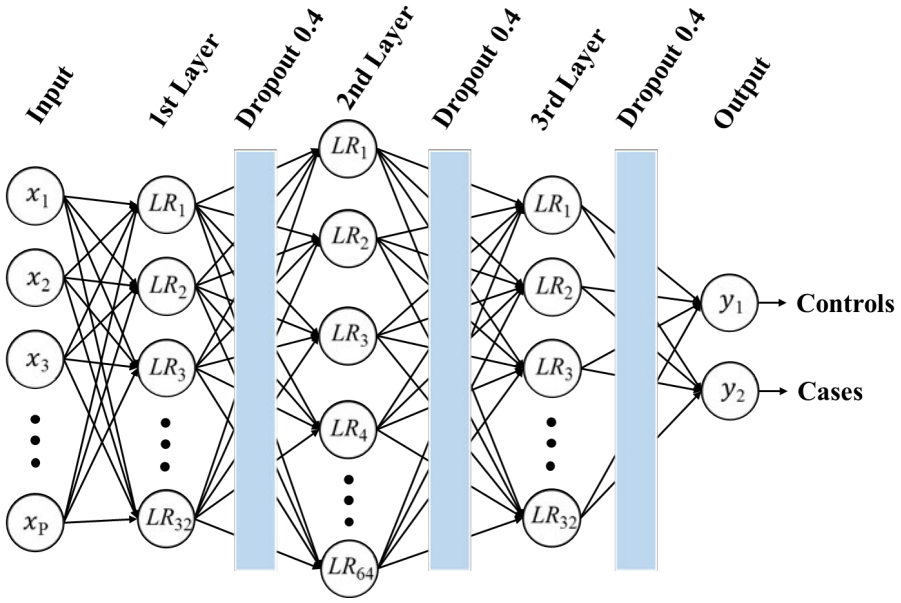

Figura 4.2: Esquema de la arquitectura del perceptrón multicapa utilizado en la experimentación con el procedimiento FNA. Presenta 3 capas ocultas con 32, 64 y 32 neuronas respectivamente, utilizando Leaky ReLU (LR) como función de activación. Las cajas azules representan el uso de dropout con 0.4 como valor de tasa de abandono.

4.3.2. Resultados experimentales

Para probar el rendimiento como método de aumento de datos del procedimiento FNA con el modelo clasificador expuesto en la subsección anterior, éste fue aplicado a nueve conjuntos de datos tomados de las bases de datos UCI (Dua y Graff, 2017) y PROBEN1 (Prechelt, 1994). Estos datos de referencia se utilizan con frecuencia en la literatura para comprobar el rendimiento en tareas de clasificación, y consisten en datos del mundo real que representan problemas binarios que podrían denominarse tareas de diagnóstico. La Tabla A.1 del Apéndice A.1 muestra algunas características de los datos de referencia utilizados para los experimentos.

Como está descrito en la sección 4.1, en la experimentación se realiza una división de los datos utilizando un procedimiento de validación cruzada estratificado de 10 folds, con 8 folds para el entrenamiento, uno para validación y el restante para test. Este procedimiento de validación cruzada está diseñado para que el conjunto de test no sea utilizado durante el entrenamiento y generación de datos, con el fin de que los resultados sean totalmente independientes. Los experimentos son realizados utilizando una sola semilla, debido al alto tiempo de computación requerido.

UNIVERSIDAD DE MÁLAGA

Después de experimentos preliminares en los que se probaron diferentes métricas, la decisión fue tomar la F_1 *score* como métrica para realizar el proceso de optimización de los niveles de ruido, siendo una métrica representativa del rendimiento en predicción con cierta independencia del balanceo del conjunto de datos. Esta métrica está descrita de forma más extendida en el capítulo 3 sección 3.2. Siendo este el primer acercamiento al trabajo desarrollado, y viéndolo en perspectiva, los resultados podrían haber sido acompañados de otras métricas para un análisis en mayor profundidad, si bien la métrica F_1 es suficiente para valorar el rendimiento alcanzado.

Tabla 4.1: Valores de F_1 referencia obtenidos con los datos originales, número de características seleccionadas para el proceso DA, F_1 usando los datos aumentados (100 % DA) y diferencia relativa (RD).

Conjunto	F_1 referencia	Características DA	F_1 DA	RD
heart cleveland	0.7805	7	0.8262	5.845
horse colic	0.7642	3	0.7703	0.801
wdbc	0.9575	3	0.9655	0.835
card	0.8346	1	0.8418	0.869
breast cancer	0.9705	3	0.9716	0.113
pima diabetes	0.8139	2	0.8227	1.090
gene	0.7514	2	0.7534	0.262
waveform	0.9051	3	0.9114	0.699
thyroid	0.9693	6	0.9727	0.355
promedio	0.8608	3.3	0.8708	1.234

$$\mathrm{RD} = \frac{(\mathrm{F_1_DA} - \mathrm{F_1_Ref})}{\mathrm{F_1_Ref}} \times 100 \tag{4.2}$$

Con el propósito de comprobar, no solo la efectividad del aumento de datos, sino también el efecto del número de muestras generadas, se realizaron tres pruebas con diferentes porcentajes de datos creados. Así, se comprobó el efecto aumentando un porcentaje del 100 % de muestras de entrenamiento, del 200 % y del 500 %. Esto implica el duplicar el tamaño del conjunto de entrenamiento (100 %), triplicarlo (200 %) y sextuplicarlo (500 %). En la Tabla 4.1 se muestran los resultados obtenidos para los 9 conjuntos de datos utilizados: la medida F_1 cuando no se aplica ningún aumento de datos y que sirve de referencia; el número de características del conjunto de datos para las que el procedimiento seleccionó un nivel de ruido diferente de 0 y, por tanto, fueron modificadas en el proceso de DA; la medida F_1 usando el conjunto de datos de entrenamiento aumentado al 100 %; y la diferencia relativa porcentual (RD %), cuyo cálculo aparece descrito en la Ecuación 4.2, entre la medida F_1 cuando se aplica DA y la medida obtenida cuando no se aplica (F_1_Ref). Las Tablas 4.2 y 4.3 muestran la misma información, pero utilizando 200 % y 500 % como porcentajes de aumento del conjunto.

UNIVERSIDAD DE MÁLAGA

Tabla 4.2: Valores de F_1 referencia obtenidos con los datos originales, número de características seleccionadas para el proceso DA, F_1 usando los datos aumentados (200 % DA) y diferencia relativa (RD).

Conjunto	F_1 referencia	Características DA	F_1 DA	RD
heart cleveland	0.7805	8	0.8269	5.945
horse colic	0.7642	4	0.7778	1.789
wdbc	0.9575	4	0.9683	1.130
card	0.8346	8	0.8428	0.978
breast cancer	0.9705	6	0.9724	0.201
pima diabetes	0.8139	5	0.8331	2.367
gene	0.7514	3	0.7537	0.303
waveform	0.9051	3	0.9146	1.050
thyroid	0.9693	5	0.9738	0.464
promedio	0.8608	5.1	0.8737	1.580

Tabla 4.3: Valores de F_1 referencia obtenidos con los datos originales, número de características seleccionadas para el proceso DA, F_1 usando los datos aumentados (500 % DA) y diferencia relativa (RD).

Conjunto	F_1 referencia	Características DA	F_1 DA	RD
heart cleveland	0.7805	10	0.8321	6.606
horse colic	0.7642	4	0.7798	2.044
wdbc	0.9575	6	0.9769	2.024
card	0.8346	8	0.8397	0.607
breast cancer	0.9705	3	0.9716	0.113
pima diabetes	0.8139	4	0.8242	1.275
gene	0.7514	2	0.7552	0.499
waveform	0.9051	2	0.9157	1.180
thyroid	0.9693	5	0.9740	0.484
promedio	0.8608	4.9	0.8743	1.648

Los resultados obtenidos indican un aumento en la métrica F_1 con los datos aumentados para los 3 porcentajes utilizados. Los valores promedios de RD obtenidos son mayores de 1.2 con los tres niveles de aumento estudiados. Sin embargo, es necesario tener en cuenta que estos valores se ven altamente influenciados por el gran aumento alcanzado para el conjunto *heart cleveland*, cuyo valor de RD es mayor de 5.7 para todos los porcentajes estudiados. Al respecto del número de características para las que el procedimiento seleccionó un nivel de ruido diferente de 0, es decir, que han sido modificadas durante el procedimiento FNA, el valor promedio está entre el 10 % y el 16 % de características totales. Se puede observar como este valor también ve incrementado con los porcentajes de muestras generadas de 200 % y 500 % respecto al obtenido con 100 %. Para este último la media de características modificadas es de 3.3, mientras que para porcentajes de 200 % y 500 % son de 5.1 y 4.9 respectivamente, siendo estos valores similares.

UNIVERSIDAD DE MÁLAGA

Tabla 4.4: Diferencias relativas (RD) de la métrica F_1 obtenidas medidas para los nueve conjuntos de datos de referencia utilizando diferentes niveles de aumento de datos (100 %, 200 % y 500 %).

Conjunto	RD 100 % DA	RD 200 % DA	RD 500 % DA
heart cleveland	5.845	5.945	**6.606**
horse colic	0.801	1.789	**2.044**
wdbc	0.835	1.130	**2.024**
card	0.869	**0.978**	0.607
breast cancer	0.113	**0.201**	0.113
pima diabetes	1.090	**2.367**	1.275
gene	0.262	0.303	**0.499**
waveform	0.699	1.050	**1.180**
thyroid	0.355	0.464	**0.484**
promedio	1.234	1.580	**1.648**

La Tabla 4.4 resume los resultados para los diferentes niveles de aumento de datos (DA) que se muestra en las tablas anteriores, indicando con la fuente en negrita los mejores valores obtenidos. Para 6 de los 9 conjuntos de referencia, los mejores valores se obtienen con un porcentaje de muestras generadas del 500 %, mientras que el porcentaje del 200 % obtiene los mejores valores para los 3 conjuntos restantes. Aun así, los resultados no se diferencian estadísticamente entre sí, siendo 0.0035 la diferencia absoluta entre el valor de la métrica F_1 obtenido con los aumentos del 500 % y del 100 %.

Para comparar la efectividad de los modelos de DL, se llevó a cabo un experimento adicional utilizando un MLP tradicional. Esta red tenía 32 neuronas en la única capa oculta que presenta, con una arquitectura completamente conectada. En promedio, los resultados obtenidos con los tres niveles de aumento de datos estudiados mostraron un valor para la medida F_1 de 0.854 cuando no se aplica aumento de datos y de 0.86 cuando se aplica aumento de datos. El valor para la diferencia relativa porcentual entre aplicar o no aumento de datos, es de 0.729. Como era de esperar, estos resultados son peores que los obtenidos utilizando arquitecturas profundas (0.861, 0.873 y 1.49 respectivamente), pero en este caso también se puede observar el efecto de realizar DA.

La Figura 4.3 muestra la relación entre la medida F_1 obtenida con los conjuntos de entrenamiento aumentados al 500 % y la medida F_1 con el caso de referencia sin aumento de datos aplicado, para cada uno de los conjuntos de datos analizados. De esta forma, se puede observar el aumento logrado en F_1 respecto a la función de identidad (línea continua). El análisis no muestra una tendencia clara de que exista una dependencia del efecto de DA respecto al valor de referencia.

También se analiza la dependencia entre los resultados obtenidos de medida relativa F_1 para cada conjunto de datos según el número de característi-

UNIVERSIDAD DE MÁLAGA

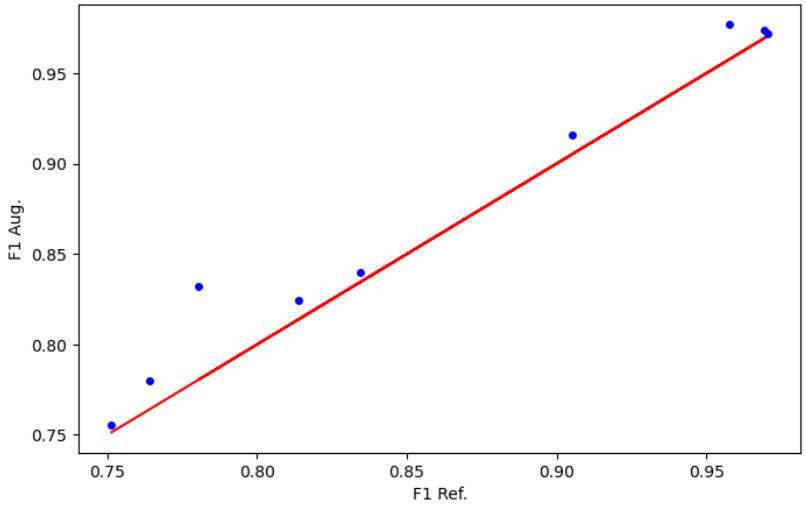

Figura 4.3: Relación entre la medida F_1 para los conjuntos de datos de entrenamiento aumentados (500 % DA) y los valores de referencia para los conjuntos de test (la línea continua es la función de identidad).

cas, el número de instancias y el grado de desequilibrio (medido por la clase mayoritaria %). Aclarar que la medida relativa F_1 (rF_1) es el valor de F_1 obtenido con la aplicación de aumento de datos, dividido por el valor de F_1 obtenido con los conjuntos originales. Los resultados mostrados en la Figura 4.4 pueden sugerir alguna tendencia hacia poder alcanzar mejores resultados mediante el proceso de DA cuando se consideran conjuntos de datos que presentan valores bajos de los tres aspectos analizados. Es ligeramente más clara la relación con el número de instancias presentes en los datos, pues el proceso de DA incrementa el número de muestras de los que el clasificador puede aprender, sin embargo, el análisis indica que es necesario realizar más experimentos con otros conjuntos de datos para verificar esta relación.

El aumento de datos no solo conduce a un mejor rendimiento de predicción como se muestra antes, sino que también acelera el rendimiento de entrenamiento de los modelos como se ilustra en la Figura 4.5, que muestra la pérdida obtenida al ajustar el modelo a los conjuntos de datos '*card*'y '*wdbc*' sin aumento de datos y con aumento de datos para los tres niveles considerados: 100 %, 200 % y 500 %. El proceso de aumento de datos para el conjunto de datos '*wdbc*' (resultados promedio mostrados en la Tabla 4.4) corresponde a un nivel de ruido de 0.4, 0.8 y 0.7 para los atributos 22, 27 y 20 respectivamente; y para el conjunto de datos '*card*' corresponde a un nivel de ruido de 0.2 y 0.8 para los atributos 9 y 8 respectivamente. Una limitación para el número de experimentos realizados fue el tiempo computacional involucrado en las simulaciones numéricas, que necesita aproximadamente 5 días para cada conjunto de parámetros (combinación de niveles de ruido)

UNIVERSIDAD DE MÁLAGA

para un nivel de aumento del 100 % para todo el conjunto de datos de referencia. Este tiempo se ve incrementado considerablemente para los niveles de aumento de datos mayores (200 % y 500 %). Los tiempos de ejecución de los experimentos han sido medidos utilizando un PC Intel Core 7 bajo Windows 10 usando una GPU NVIDIA ® Titan Xp.

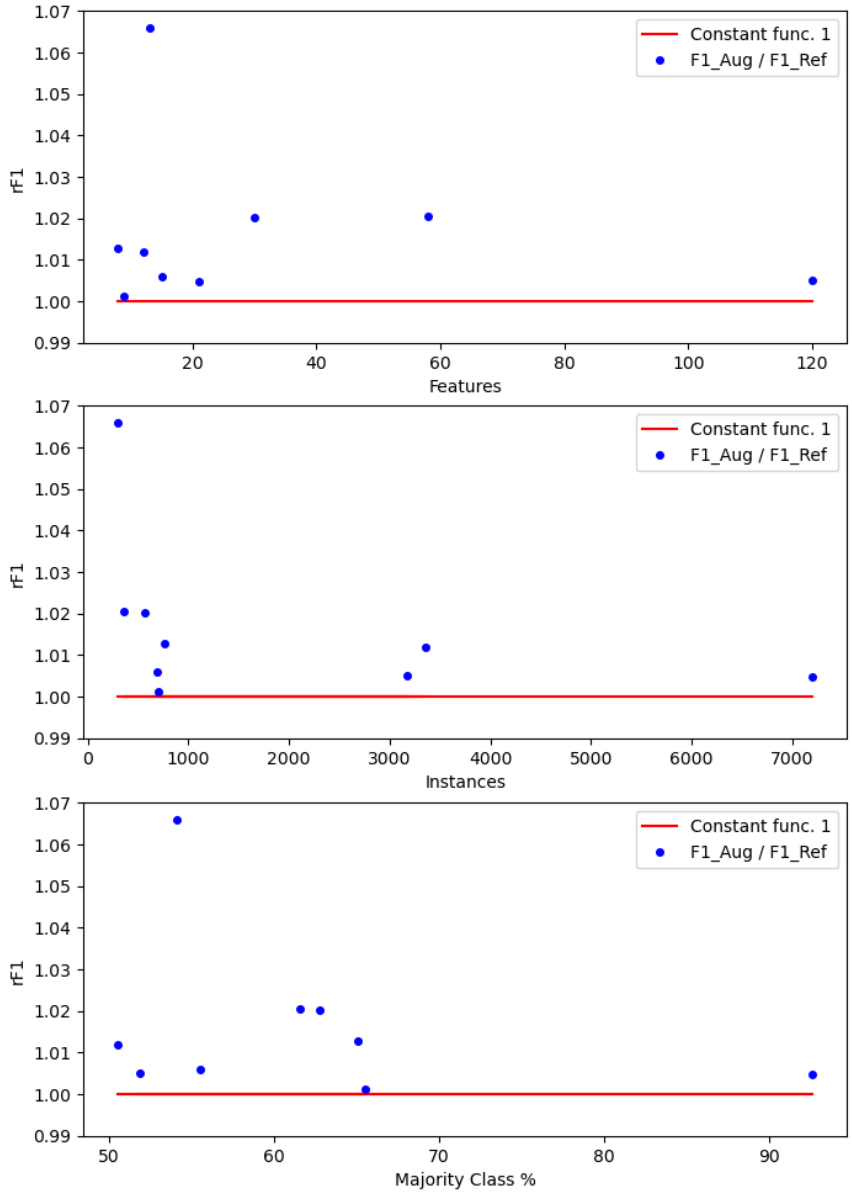

Figura 4.4: Medida relativa de F_1 en función del número de características (gráfico superior), número de instancias (gráfico medio) y grado de desbalanceo (gráfico inferior) obtenido para cada conjunto de datos de referencia.

UNIVERSIDAD DE MÁLAGA

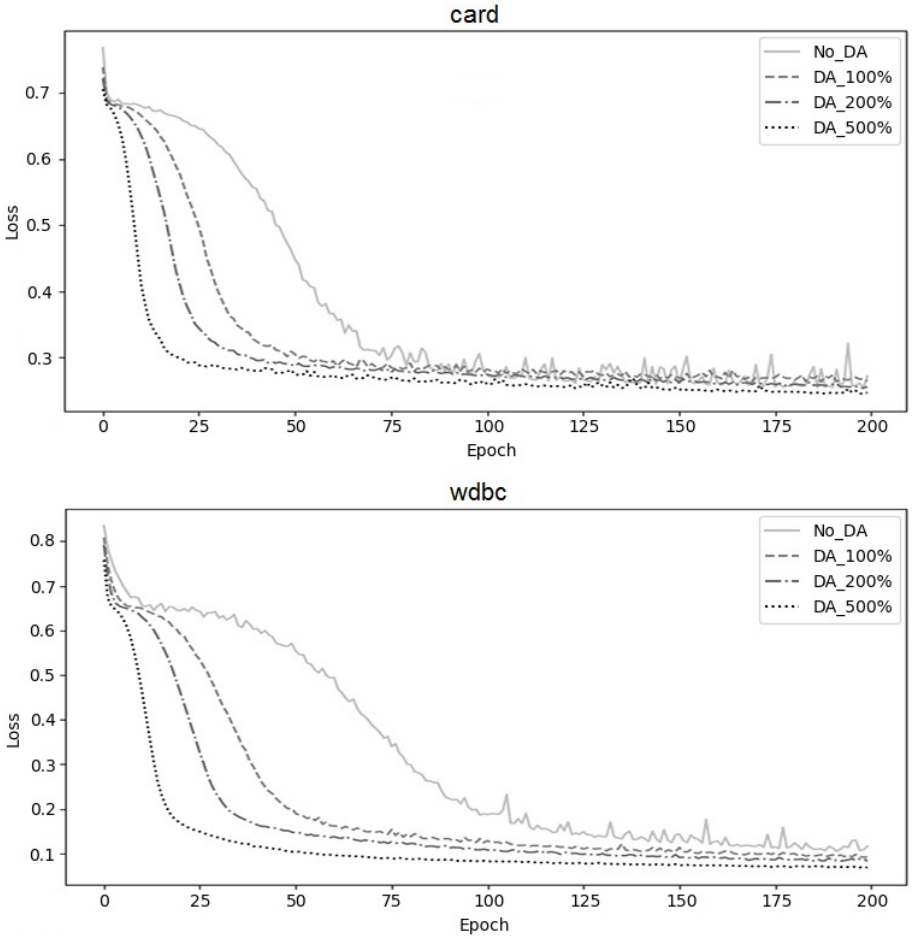

Figura 4.5: Rendimiento en entrenamiento del modelo DL respecto al número de épocas para los conjuntos '*card*'y '*wdbc*'. Las líneas muestran la pérdida obtenida cuando no se aplica ningún aumento de datos, y con el proceso de aumento de datos teniendo en cuenta el número de patrones generado según un porcentaje de 100 %, 200 % y 500 %.

4.4. Conclusiones

En este capítulo se han presentado varios métodos de DA basados en adición de ruido, dos basados en adición aleatoria de ruido gaussiano y un método denominado FNA. Este último método es un esquema de ajuste de niveles de ruido para la tarea de aumento de datos con el objetivo de mejorar la métrica F_1 en la predicción de problemas de clasificación. La propuesta consiste esencialmente en establecer el nivel de ruido adecuado para las características individuales en un proceso progresivo en el que se consideran en primer lugar las características más relevantes. De este modo,

UNIVERSIDAD DE MÁLAGA

una modificación de estas características debería tener asociado un impacto mayor, bien de forma positiva o negativa, en el rendimiento de predicción alcanzado. Al finalizar, se tiene una colección de grados de ruido a aplicar para cada característica, siendo un procedimiento totalmente personalizado para el conjunto de datos.

En el capítulo se muestran los resultados obtenidos de realizar un extenso número de simulaciones experimentales comprobando la aplicación de este método FNA en un proceso de aumento del conjunto de datos disponibles, utilizando nueve conjuntos de datos bien conocidos, usados de forma habitual en aprendizaje computacional y provenientes de las bases de datos UCI (Dua y Graff, 2017) y PROBEN1 (Prechelt, 1994). Estos conjuntos representan específicamente problemas binarios con un número de entradas entre 8 y 120 muestras. Los análisis indican en el mejor de los casos un aumento relativo promedio de 1.65 % en la métrica F_1, correspondiente a la situación en la que el conjunto de datos se incrementó seis veces (500 % de DA). El comportamiento del método propuesto se analizó en función de parámetros particulares de los datos, como son el número de características e instancias en el conjunto, o el balanceo de clases presente. Los análisis realizados no manifestaron una tendencia clara sobre la relación de estas características con el rendimiento alcanzado, imposibilitando una conclusión certera sobre este aspecto. También se visualizó el rendimiento alcanzado por el modelo durante el proceso de entrenamiento, demostrando que la aplicación de DA no solo conduce a un mejor rendimiento de predicción, sino que también acelera la convergencia de los modelos de aprendizaje profundo. El DA disminuye las épocas necesarias para que el error de la red se vea reducido, alcanzando antes una estabilidad del error.

Estos resultados muestran pues los beneficios del DA mediante la aplicación del método FNA propuesto. Sin embargo, este método presenta la gran desventaja de requerir un tiempo de computación muy elevado respecto a los otros métodos propuestos. Además, aunque los resultados obtenidos con el método son satisfactorios, la mejora en predicción alcanzada no es equivalente al logrado con métodos DA en su aplicación con imágenes. Este análisis hizo pensar en considerar un enfoque más novedoso e interesante, como es el uso de modelos generativos profundos, especialmente los modelos GAN.

UNIVERSIDAD DE MÁLAGA

UNIVERSIDAD
DE MÁLAGA

Capítulo 5

Adaptación de Modelos Generativos Profundos

Resumen: Es conocido que el aumento de datos (DA) es un elemento clave en el éxito de los modelos de aprendizaje profundo, ya que su uso puede conducir a mejores valores de capacidad de predicción cuando se utilizan conjuntos de datos de gran tamaño. Especialmente, los modelos de última generación basados en *Variational Autoencoders* (VAEs) y *Generative Adversarial Networks* (GANs), son los modelos del aprendizaje profundo que están liderando el camino de la generación de datos sintéticos, especialmente en el procesamiento y la generación de imágenes. El impresionante rendimiento de estos modelos ha dado lugar a una mayor atención sobre cómo se pueden aplicar en DA. En este capítulo se desarrolla un esquema adaptativo para estos modelos, investigando el efecto del aumento mediante estos métodos al aplicarlos a conjuntos de pequeño tamaño, y analizando los resultados en términos de capacidad predictiva obtenida según las diferentes características de las muestras de entrenamiento. Además, se introducen dos modificaciones al esquema adaptativo para alterar el equilibrio de las clases. Los resultados generales indican que, con un esfuerzo computacional sustancial, se puede obtener un aumento significativo en la capacidad predictiva cuando se consideran conjuntos de datos pequeños.

La gran desventaja de los modelos generativos profundos es su uso en procesos de generación de muestras para aprendizaje supervisado, como las tareas de clasificación/predicción. Salvo versiones de estos modelos denominados condicionales (Conditional VAE (Kingma et al., 2014; Sohn et al., 2015) y Conditional GAN (Mirza y Osindero, 2014)) que incluyen las etiquetas de los datos en el proceso de entrenamiento y generación, el resto de modelos no estaban diseñado con este propósito, empezando por los modelos básicos VAE (Kingma y Welling, 2013) y GAN (Goodfellow et al., 2014a),

UNIVERSIDAD DE MÁLAGA

o sus versiones mejoradas Denoising VAE (Im et al., 2017) y Wasserstein GAN-GP (Gulrajani et al., 2017). De este modo, teniendo en cuenta que los modelos básicos se utilizan esencialmente en generación de imágenes sintéticas partiendo únicamente de una entrada ruidosa, a la hora de asignarle una posible etiqueta es necesario una inspección visual de la imagen generada por parte de un experto. Incluso en el caso de los modelos GAN, es necesario que un experto evalúe la calidad de las imágenes generadas, aun teniendo en cuenta que la red discriminadora ya realiza un proceso de filtrado de muestras que considera falsas. Es necesario mencionar que este último hecho se ha vuelto menos necesario conforme los modelos GAN han sido perfeccionados, desarrollando los investigadores multitud de modelos diferentes, y mejorando con ello la calidad de las imágenes generadas. Sin embargo, no hay experto que pueda etiquetar una muestra generada para conjuntos de datos que no tienen una estructura espacial, como ejemplariza la Figura 5.1.

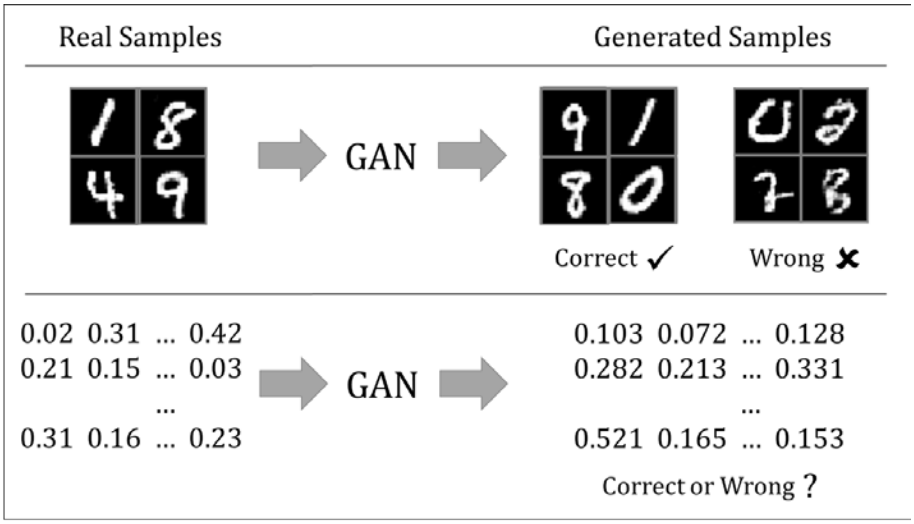

Figura 5.1: Ejemplo de generación de muestras sintéticas con un modelo GAN cuando se aplica con un conjunto de imágenes (MNIST) y con un conjunto no estructurado. Existe un impedimento a la hora de realizar una evaluación visual de la corrección de las muestras generadas.

5.1. Modelo clasificador generativo

Debido a las razones expuestas previamente, se propuso un método en el cual se utilice como experto un modelo clasificador, adaptable a cualquier método o modelo que realice una tarea de predicción (Moreno-Barea et al., 2020). A este respecto, hay que señalar el modelo *Auxiliary Classifier* GAN (ACGAN) (Odena et al., 2017), desarrollado y presentado a mediados de

UNIVERSIDAD DE MÁLAGA

2017 para su uso en imágenes, y que ganó difusión a la vez que desarrollábamos nuestro método clasificador generativo. El principio de este modelo, explicado más en profundidad en el Capítulo 2 sección 2.3.4, es utilizar la red discriminadora para realizar la predicción de si corresponde a una muestra real o falsa, y a la vez realizar una predicción de la etiqueta que debería tener asignada. Este es similar al proceso adaptable desarrollado, sin embargo, ACGAN sufre de ciertos fallos a la hora de generar muestras y asignarles etiquetas por el hecho de compartir una misma red para dos propósitos. Además, no es un modelo que pueda ser adaptado, por ejemplo, a una VAE o una WGAN.

Por estas razones el desarrollo de este modelo adaptativo es interesante y novedoso. Todo el proceso que realiza el modelo clasificador generativo se muestra en la Figura 5.2, donde se puede ver que incluye dos modelos representados por redes neuronales: un "Generador" (G) y un clasificador denominado "Clasificador Generativo" (C). Como ya se ha mencionado, en la investigación tratamos con problemas supervisados que involucran muestras etiquetadas, de forma que el proceso de DA incluye dos pasos en relación con esto: primero el Generador (G) para la creación de nuevos datos sintéticos y, en segundo lugar, el uso de un clasificador generativo (C) para etiquetar las nuevas muestras creadas. Se puede ver en la Figura 5.2 que todo el proceso consiste esencialmente en un bucle entre estos dos subprocesos principales, donde la sentencia de control IF incluida hace las funciones de verificador, comprobando si el Clasificador Generativo decide que la muestra creada es considerada como ruido o como muestra real; y en el caso de ser considerada real, le asigna de forma directa una etiqueta que permitirá su aplicación en un esquema de clasificación supervisada.

En primer lugar, el entrenamiento del modelo de Clasificador Generativo utiliza muestras del conjunto de entrenamiento (Train) para que sea capaz de aprender a otorgar una clase partir de las muestras reales, pero también incorpora muestras ruidosas que se generan de dos fuentes diferentes: a) mediante el uso de una distribución aleatoria uniforme; b) mediante la adición de ruido gaussiano (con media igual a la media de la muestra y desviación estándar igual a 0.3). Las muestras creadas por ambos métodos de generación de ruido, tienen asignada una etiqueta que indica su consideración como ruido. Se utilizan estas dos fuentes diferentes de ruido para, con la primera, crear una distribución normal que no está relacionada con la distribución original de muestras y, por lo tanto, es una forma de enseñar al clasificador una distribución que se trata de ruido "puramente"; y con la segunda fuente, en la que se utiliza un esquema de perturbación gaussiano, crear muestras relacionadas con la distribución original. La adición de estas muestras ruidosas se realiza con el propósito de que el Clasificador Generativo sea robusto al ruido que puedan llevar consigo las muestras sintéticas generadas, haciendo las veces de discriminador a la vez que otorga etiquetas de clase.

UNIVERSIDAD DE MÁLAGA

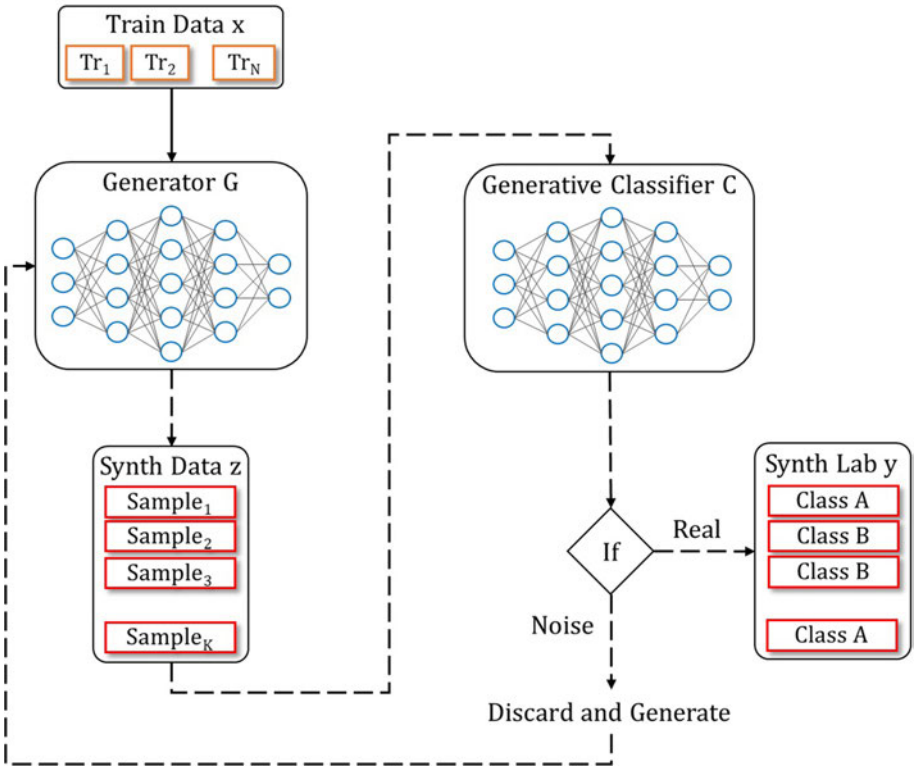

Figura 5.2: Proceso del Modelo clasificador generativo para la creación de datos sintéticos. El modelo generativo G tiene como entrada el conjunto train. Cuando se entrena, se generan las muestras sintéticas z y el clasificador generativo C predice las etiquetas apropiadas y. Si la etiqueta label_i representa ruido, sample_i se descarta y G genera otra nueva muestra.

La Figura 5.3 muestra un ejemplo de un conjunto de datos original con dos clases $\{A, B\}$, que con la adición de las muestras ruidosas se convierte en un problema de 3 clases $\{A, B, \text{Noise}\}$. El conjunto de validación se utiliza en el modelo para realizar una parada temprana en el modelo, además de servir de evaluación para su comportamiento. Por el diseño del esquema general del proceso, como modelo clasificador puede utilizarse cualquier algoritmo de aprendizaje supervisado, desde una regresión logística a una red neuronal profunda. Incluso puede variar el entrenamiento del modelo para adaptar el nivel de ruido que recibe, o realizar un aumento de datos previo en el conjunto para mejorar la estabilidad del mismo.

Por otro lado, el entrenamiento del modelo Generador (G) se lleva a cabo utilizando únicamente las muestras del conjunto de entrenamiento, manteniendo independencia total con los conjuntos de validación y test. Como se ha mencionado, este modelo Generador puede ser sustituido por cualquier modelo generativo profundo, o por cualquier otro método de creación de

UNIVERSIDAD DE MÁLAGA

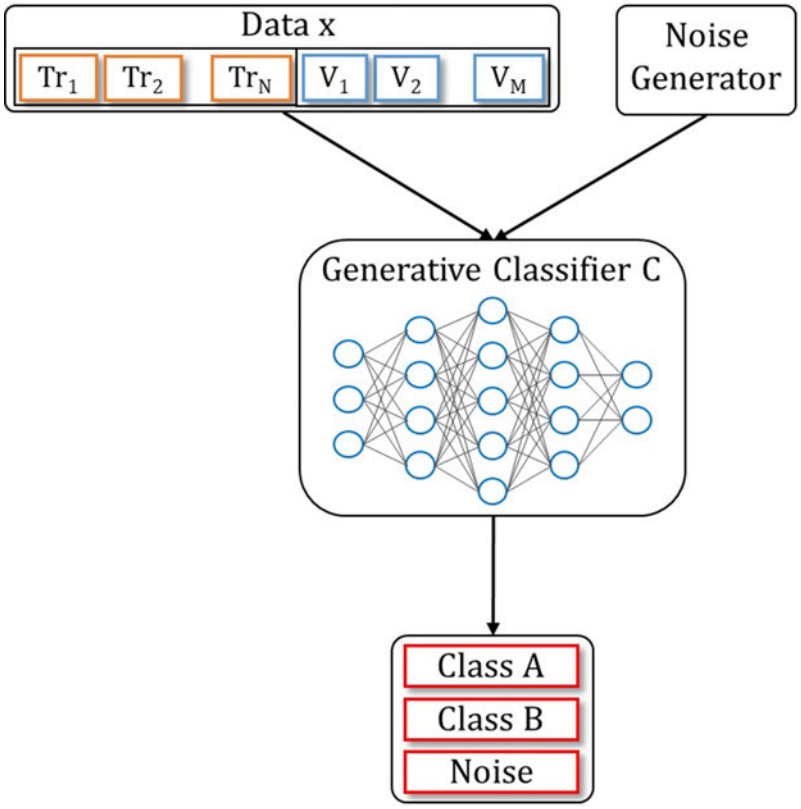

Figura 5.3: El proceso de entrenamiento del clasificador generativo (C) implica el uso de entrenamiento (Tr_i) y conjuntos de validaciones (V_i) de los datos de entrada, junto con el uso de fuentes de ruido. El objetivo del entrenamiento es que C discrimine entre las muestras originales y las ruidosas.

muestras sintéticas. Debido a esto, los detalles de entrenamiento y medición de desempeño específicos aplicados para cada modelo generador son independientes del resto del modelo, incluidas las modificaciones implementadas para el tratamiento de conjuntos de datos multiclase no balanceados explicadas en la siguiente sección. Una vez entrenado el Generador, se puede aplicar la generación de datos sintéticos seguida de la generación de las etiquetas correspondientes utilizando el modelo de Clasificador Generativo.

Como se muestra en la Figura 5.2, el Generador (G) produce un conjunto de muestras sintéticas z que se introducen en el Clasificador Generativo para obtener una etiqueta asignada a las nuevas muestras. Los procesos de entrenamiento del Clasificador Generativo y Generador se pueden realizar en cualquier orden, o bien en un proceso paralelo, ya que ambos son totalmente independientes. El único requisito es que el entrenamiento de ambos procesos se haya completado antes de iniciar la generación de etiquetas para los datos sintéticos.

UNIVERSIDAD DE MÁLAGA

5.2. Modificaciones para tratar distribuciones desbalanceadas

Cuando las diferentes variaciones de los modelos generativos profundos VAE y GAN se aplicaron directamente a la generación de datos sintéticos y se usaron para entrenar un modelo predictivo supervisado, se observó que la mayoría de las veces generan datos sintéticos solo para una de las clases, generalmente para la clase mayoritaria. Este hecho puede provocar un desbalanceo mayor en el conjunto aumentado, llevando consigo un posible colapso en el modelo clasificador posterior, incapacitando la predicción correcta de las muestras pertenecientes a la clase minoritaria. Para evitar el problema mencionado anteriormente, y después de experimentar con diferentes opciones, se introducen y describen a continuación dos modificaciones denominadas 'Multiclass' y 'Balanced Multiclass'.

Lo primero que hay que tener en cuenta de ambas modificaciones, es que utilizan tantos modelos generativos como clases estén presentes en el conjunto de datos original, en lugar de un solo modelo generativo para todo el conjunto de datos. De esta forma, el primer paso del proceso consiste en dividir los datos de entrada según las etiquetas del problema (clases). Para cada una de las n clases, se entrena un modelo generador tomando todas las muestras que pertenecen a la categoría del modelo junto con una selección aleatoria de muestras de las otras clases, tratando de alcanzar una proporción del 20 % de muestras que no pertenecen a la categoría del modelo (en los casos en que el número total de muestras de las clases restantes sea menor que esta proporción, se utilizan todas las muestras existentes).

La primera modificación implementada, denominada como 'Multiclass', esencialmente funciona manteniendo la proporción de muestras de cada clase como en los datos originales. Para la segunda modificación introducida, denominada 'Balanced Multiclass', se generan más muestras para las clases menos representadas, es decir, después de la aplicación del proceso de DA, el conjunto de entrenamiento aumentado será más equilibrado. El número de muestras L que el modelo generador de la clase i tiene que producir viene dado por las siguientes ecuaciones para ambos enfoques:

$$\mathrm{L} = \mathrm{x} \cdot \mathrm{K}, \quad \text{para el caso “Multiclass”} \tag{5.1}$$

$$\mathrm{L} = [(1 - \mathrm{x})/(\mathrm{n} - 1)] \cdot \mathrm{K}, \quad \text{para el caso “Balanced Multiclass”} \tag{5.2}$$

donde x es la proporción de la clase i en los datos originales, n es el número de clases y K es el total número de muestras que se desea generar. Las modificaciones introducidas tienen un doble objetivo, especialmente en la modificación 'Balanced Multiclass'. En primer lugar, tratar de modificar el sesgo de los modelos generadores, de forma que tener un generador que observa más muestras de una clase que de otras, provoque que este se centre

UNIVERSIDAD DE MÁLAGA

mejor en generar muestras de más calidad para esta clase, pero observando a su vez la diferencia con el resto de clases. En segundo lugar, no se pretende obligar al modelo a generar solo muestras sintéticas de una clase, sino adaptar mejor la distribución de muestras de esta clase, tratando de evitar el colapso que ocurre cuando el modelo se alimenta con todos los datos.

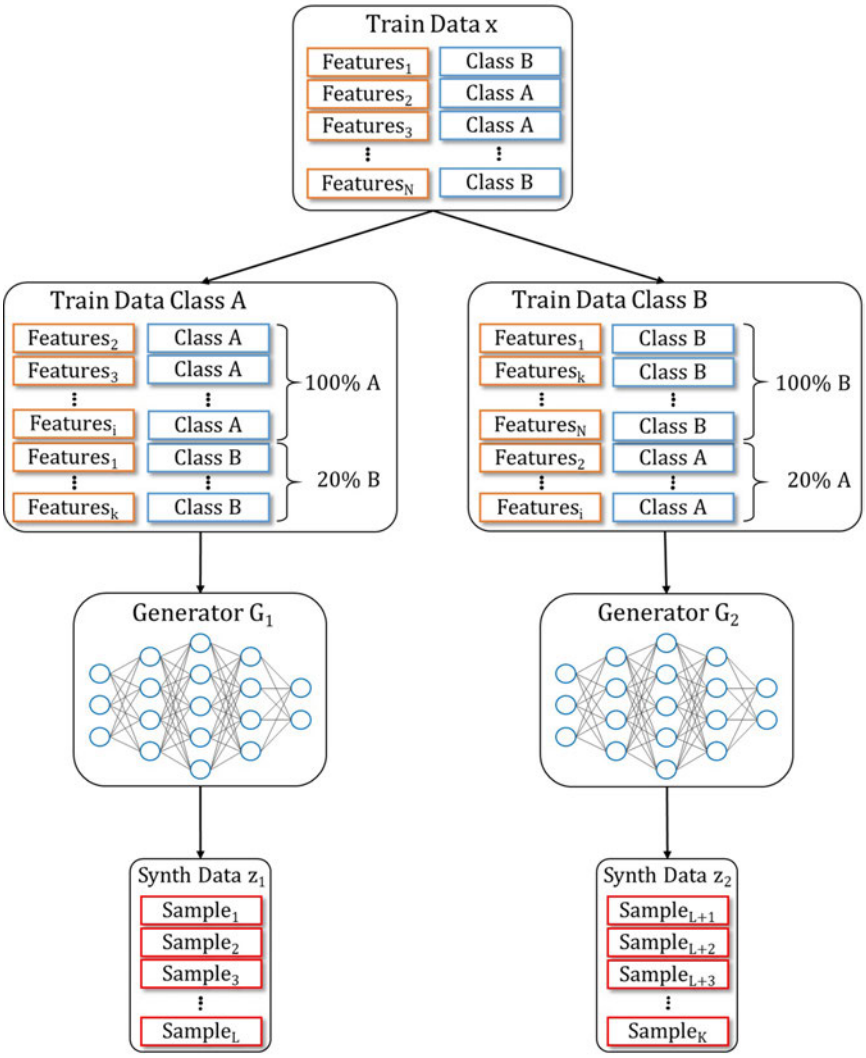

Figura 5.4: Esquema de las modificaciones introducidas para abordar el problema del aumento de datos.

El número de muestras generadas por cada uno de los generadores depende del método implementado utilizado, del número de muestras deseadas y de la proporción de clases en el conjunto original. Por ejemplo, si se considera un conjunto de datos con tres categorías que originalmente contienen 60, 20 y 20 muestras para cada clase, y el objetivo es aumentar el tamaño en un

UNIVERSIDAD DE MÁLAGA

100 %, para el caso de la modificación 'Multiclass' el objetivo es generar un número de muestras cercano a 120, 40 y 40 respectivamente para cada clase. Mientras que la distribución de clases objetivo será de 80, 60 y 60 respectivamente para la modificación 'Balanced Multiclass'. Dentro de la aplicación del proceso del modelo clasificador generativo, si una muestra sintética generada por el modelo generador i se clasifica como ruido, el mismo modelo i genera una nueva muestra. El conjunto final de datos sintéticos z resulta de la unión de los diferentes conjuntos de muestras generadas por cada generador.

El proceso descrito anteriormente se muestra esquemáticamente en la Figura 5.4. Una vez que se tienen los datos de entrenamiento x, en este ejemplo con datos de clase binaria, A y B, se dividirán en dos conjuntos. Uno de los subconjuntos está compuesto por todas las muestras de la clase A y el 20 % de muestras seleccionadas al azar de la clase B, y el otro subconjunto está compuesto por todas las muestras de B y 20 % de las muestras de A. Tras esta división, los modelos generativos G_1 y G_2 se entrenan con su subconjunto respectivo. Ya en el proceso de generación de muestras sintéticas, las muestras sintéticas z_1 y z_2 se generan siguiendo una de las modificaciones diseñadas para ello, hasta que tengan el tamaño deseado K.

5.3. Modelos generativos profundos estudiados

A modo de resumen comparativo de todos los modelos que han sido aplicados con el anterior modelo clasificador generativo, se muestra en la Tabla 5.1 una descripción de las principales características de los diferentes modelos utilizados, incluyendo también las modificaciones de balanceo introducidas para la generación de datos sintéticos. La tabla presenta el nombre de los modelos y su abreviatura formal, que en el caso de modificaciones de modelo indica el sufijo que se agregará a la abreviatura, '_M' para 'Multiclass' y '_BM' para 'Balanced Multiclass'. En este también se muestra una breve descripción de la estructura y operación que presenta cada modelo, y la función de costo objetivo que aplican.

Las explicaciones detalladas de cada modelo generador estudiado pueden verse en el Capítulo 2 sección 2.3.3 (modelos VAE) o en la sección 2.3.4 (modelos GAN). En estas secciones se detallan pues las motivaciones, funcionamiento, objetivo y desventajas de cada uno de los modelos.

5.3.1. Implementaciones de los modelos

Los detalles de todos los modelos usados en la experimentación desarrollada en este capítulo, son expuestos en esta sección. La disposición de la arquitectura (número de capas y organización de las neuronas) de los modelos VAE y GAN se ha seleccionado tras una amplia consulta de la literatura y un proceso de optimización, siendo el número de neuronas seleccionadas

UNIVERSIDAD DE MÁLAGA

UNIVERSIDAD
DE MÁLAGA

Tabla 5.1: Breve descripción de los diferentes modelos utilizados para los experimentos de DA.

Modelo	Abrev.	Resumen Arquitectura	Función
Variational Autoencoder	VAE	Dos redes neuronales:codificador y decodificador. La salida del codificador se transforma en una variable latente que se convierte en la entrada del decodificador, intentando recrear los datos originales.	Evidence Lower Bound y Kullback-Leibler Divergence.
Denoising VAE	DVAE	Similar a VAE pero con ruido agregado a la entrada del codificador.	Igual que VAE.
Conditional VAE	CVAE	Similar a VAE pero con etiquetas de muestra agregadas a las entradas del codificador y decodificador.	Igual que VAE.
Generative Adversarial Network	GAN	Dos redes neuronales: generador y discriminador. Una distribución aleatoria es la entrada del generador, mientras que las muestras sintéticas son la salida. El discriminador recibe muestras reales y sintéticas intentando predecir si son reales o no.	Jensen−Shannon Divergence.
Wasserstein GAN con penalización de gradiente	WGAN-GP	Similar al modelo GAN, pero en este caso el discriminador se llama Critic y se entrena varias veces para cada iteración del generador.	Distancia Wasserstein.
Conditional GAN	CGAN	Similar al modelo GAN pero con etiquetas de muestra agregadas a la entrada del generador y discriminador.	Igual que GAN.
'Multiclass' GANs y VAEs	_M	Esta modificación crea n modelos, cada uno para cada clase existente de los datos originales. La entrada del modelo i consiste en todas las muestras de la clase i, más un 20 % de otras clases. El número de muestras sintéticas generadas por cada modelo sigue la proporción de clases de los datos originales.	Igual que el modelo original.
'Balanced Multiclass' GANs y VAEs	_BM	Similar a la variación _M. Hay n modelos, cada uno recibe como entrada todas las muestras de una clase y un 20 % del resto. La diferencia con el método _M es el número de muestras sintéticas generadas por cada modelo, que en el presente caso sigue una proporción inversa de las clases en los datos, y así intenta equilibrar la distribución de las muestras.	Igual que el modelo original.

el suficiente para no tener problemas de capacidad de cómputo en su ejecución. Se realizaron pruebas previas a la experimentación para afinar la arquitectura, el número de neuronas y el resto de parámetros utilizados en la configuración de los modelos.

Todos los modelos VAE (VAE estándar, DVAE y CVAE) comparten las mismas características: el codificador presenta 4 capas ocultas con 512, 256, 128 y 64 neuronas respectivamente, y el decodificador tiene una estructura similar en orden inverso, con 4 capas ocultas que contienen 64, 128, 256 y 512 neuronas respectivamente. Además, el codificador tiene dos salidas (vector de medias y vector de covarianzas) con 32 neuronas cada una, siendo este también el tamaño del espacio latente que es la entrada del decodificador. Cada capa oculta de la red del codificador utiliza ReLU (Xu et al., 2015) como función de activación, ya que esta representa el estado del arte de facto de las funciones de activación en el aprendizaje profundo. Sin embargo, para el decodificador la Leaky ReLU (Maas et al., 2013) se utiliza como función de activación, ya que proporciona más estabilidad para la reconstrucción de los datos. La capa de salida del decodificador utiliza una función de activación sigmoidea para devolver como salida las muestras reconstruidas. En el caso específico del modelo CVAE, se incluye una neurona de entrada adicional para incorporar la información de la etiqueta.

La arquitectura utilizada para implementar los modelos GAN también presenta 4 capas ocultas. En este caso, las redes generadoras presentan 4 capas ocultas con 1024, 512, 256 y 128 neuronas con función de activación ReLU, y se utilizó una capa de salida con el tamaño de los datos y la función de activación de tangente hiperbólica (tanh) para la generación de las muestras. La dimensión del vector de ruido de la entrada del generador, z, se estableció en 32, teniendo en cuenta que la mayoría de conjuntos de datos probados presentan un número bajo de características. La red discriminadora (o red *critic* en el caso de WGAN-GP) también presenta 4 capas ocultas, pero con 64, 128, 256 y 512 neuronas, respectivamente, y Leaky ReLU como función de activación. La capa de salida del discriminador que decide si una muestra creada es real o falsa, contiene una sola neurona una función de activación sigmoide en el caso de los modelos GAN y CGAN. Para el caso del modelo WGAN-GP, no se aplica una función de activación a la salida. El modelo WGAN-GP también incluye un parámetro adicional durante la implementación del entrenamiento del modelo, estableciendo el número de iteraciones que debe realizar el *critic* por cada iteración del generador, al cual se le asigna un valor de 5, un valor común en su aplicación. A la entrada de las redes que componen el modelo CGAN, se le agrega una neurona adicional para incluir la información sobre la etiqueta de la muestra.

Tanto los modelos VAE como GAN utilizados incluyen la normalización por lotes como técnica de regularización (Ioffe y Szegedy, 2015), siendo utilizada en todas las capas ocultas de las redes. Para el entrenamiento de los

UNIVERSIDAD DE MÁLAGA

modelos, se utiliza Adam (Kingma y Ba, 2014) como algoritmo de optimización con tasa de aprendizaje adaptativo. Los parámetros utilizados son: 2e-4 como tasa de aprendizaje, 0.5 y 0.9 para la tasa de caída exponencial para el primer y segundo momento respectivamente, y 1e-8 para $\hat{\epsilon}$.

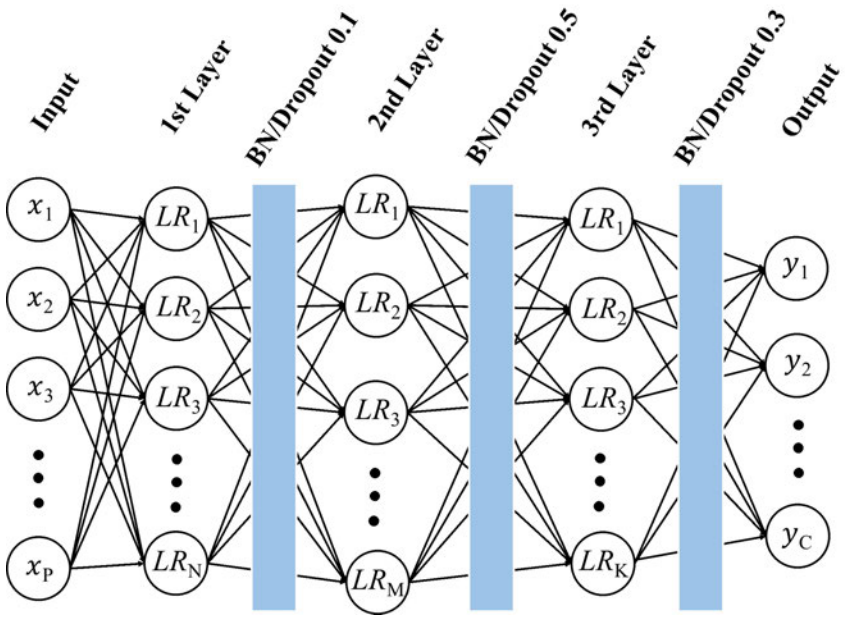

Figura 5.5: Esquema de la arquitectura del perceptrón multicapa con 3 capas ocultas con N, M y K neuronas respectivamente, utilizando Leaky ReLU (LR) como función de activación. Las cajas azules representan el uso de normalización por lotes y dropout.

En relación a los modelos clasificadores utilizados tanto para el proceso de generación de muestras sintéticas (Clasificador Generativo), como para la experimentación final de los modelos cuando se utilizan los datos aumentados, el modelo implementado es una arquitectura de tres capas ocultas. La Figura 5.5 muestra un esquema de dicha arquitectura con $(\mathrm{N}, \mathrm{M}, \mathrm{K})$ neuronas en la primera, segunda y tercera capas ocultas. El número de neuronas incluidas en cada capa depende del conjunto de datos. En la Tabla A.4 del Apéndice A.2 se presentan el número de neuronas empleado para cada conjunto de datos, valores que fueron obtenidos a partir de una experimentación explicada en el mencionado apéndice. Se utiliza Leaky ReLU como función de activación en las neuronas de las capas ocultas y la función de activación sigmoidea para las neuronas de salida que clasifican los patrones en las diferentes clases. Se utiliza una combinación de dropout (Srivastava et al., 2014), normalización de lotes y norma L^2 (también conocida como *weight decay*) como técnicas de regularización para prevenir efectos de sobreajuste. La tasa de abandono fue de 0.1, 0.5 y 0.3 en la primera, segunda y tercera capa

UNIVERSIDAD DE MÁLAGA

oculta, respectivamente. El algoritmo de Adam, mencionado anteriormente en el caso de los modelos generativos, se utiliza con la misma configuración de parámetros para el entrenamiento de las redes.

5.4. Resultados experimentales

Para realizar una comprobación del rendimiento de los modelos generadores expuestos anteriormente y con el uso del modelo clasificador generativo, incluyendo las modificaciones de balanceo en la generación de los datos, se llevaron a cabo experimentos con 19 conjuntos de datos tomados de la base de datos de UCI (Dua y Graff, 2017) y PROBEN1 (Prechelt, 1994). La Tabla A.3 del Apéndice A.2 muestra algunas características de los datos de referencia utilizados para los experimentos. El esquema completo del enfoque utilizado para realizar el DA y la experimentación para evaluar los resultados se puede ver en la Figura 5.6, donde se muestran los dos procesos principales involucrados: 1) el proceso DA en sí, y 2) el clasificador utilizado para obtener el error de predicción en datos no utilizados previamente.

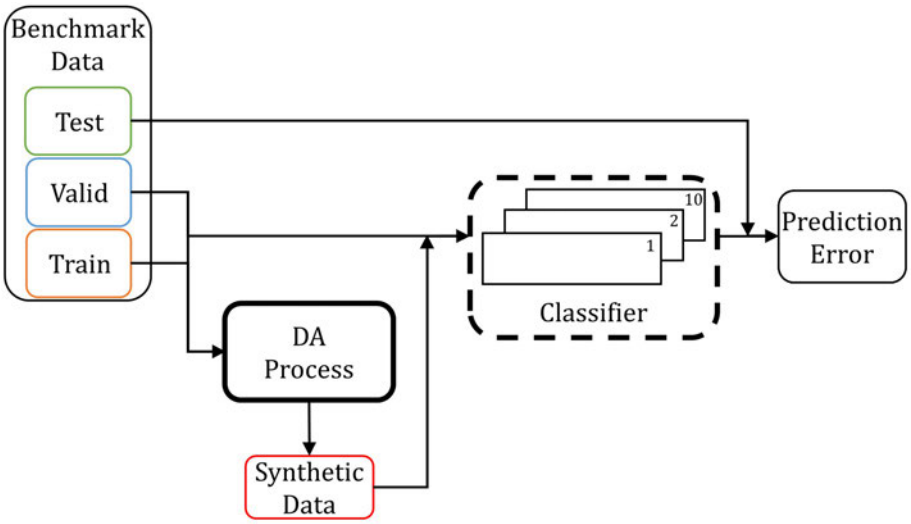

Figura 5.6: Diagrama de flujo del proceso desarrollado para realizar el DA y evaluar los resultados en un conjunto de problemas de referencia.

De este modo, en primer lugar, se realiza el proceso de generación de datos sintéticos para cada uno de los modelos, siguiendo los procedimientos descrito en el presente capítulo, y dependiendo de cada uno de los modelos generadores estudiados. Una vez finalizado este proceso, los datos sintéticos generados z se agregan en el porcentaje deseado a los datos de validación y entrenamiento previamente separados y se implementa un procedimiento de validación cruzada. Los datos de prueba no se utilizan en el esquema de

UNIVERSIDAD DE MÁLAGA

validación cruzada, ya que se guardan por separado para una prueba externa honesta del rendimiento en predicción. Como se muestra en la Figura 5.7, los conjuntos de entrenamiento y validación inicialmente separados para su uso en la generación de datos, se unen de nuevo y se emplean para hacer una división a través de un procedimiento estratificado de validación cruzada de 10-folds, de modo que se dedican 9 folds para el entrenamiento, 1 para realizar una parada temprana mediante validación en el modelo clasificador empleado en la experimentación. Es necesario mencionar que los subconjuntos de validación no incluyen datos sintéticos para no afectar al rendimiento, y también mencionar que el subconjunto de datos sintéticos que se es agregado a los subconjuntos de entrenamiento es el mismo para todos, con el fin de minimizar los posibles efectos aleatorios.

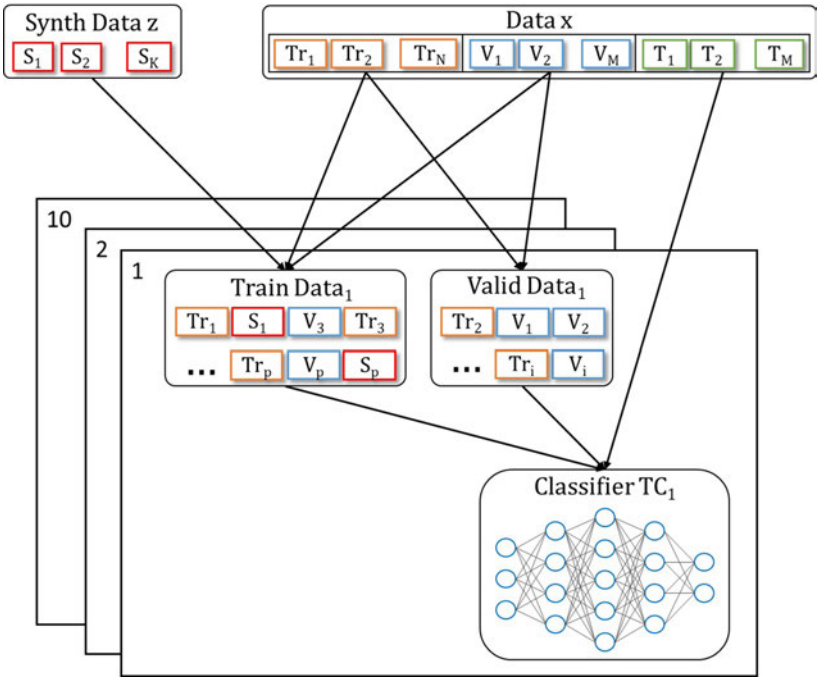

Figura 5.7: Proceso de clasificación para la realización de los experimentos. Los datos reales x se utilizan para realizar un proceso de validación cruzada de 10-folds. Los conjuntos de entrenamiento se incrementan agregando muestras sintéticas, implementando el aumento de datos. Se realiza la predicción con el conjunto Test y se mide la capacidad de predicción.

Una vez finalizado el entrenamiento, se procede a la predicción con el conjunto de Test. El resultado del proceso de clasificación es entonces el promedio de los resultados de capacidad predictiva obtenidos para los 10 folds. Este proceso se repite 10 veces más con diferentes semillas para reducir los efectos aleatorios. Como métrica de evaluación en esta experimentación se

UNIVERSIDAD DE MÁLAGA

utiliza la capacidad en predicción (*accuracy*). Con esto se pretende comprobar si el aumento mediante estos modelos lleva a una mejora en el número total de muestras bien clasificadas. No se tienen en cuenta otras métricas de las expuestas en la sección 3.2, pues se trabaja con problemas multiclase. Igualmente, dado que los conjuntos se encuentran bien balanceados, siendo el valor de equilibrio mayor que 0.9 para 15 de 19 conjuntos y para 18 de 19 mayor que 0.8 (totalmente balaceado igual a 1), la capacidad predictiva no es una métrica errónea. En cuanto a los niveles de aumento de datos se comprobó el efecto con un porcentaje del 100 % de muestras de entrenamiento, del 200 % y del 50 %. Esto implica el duplicar el tamaño del conjunto de entrenamiento (100 %), triplicarlo (200 %) y aumentarlo en la mitad (50 %). Estos niveles de DA han sido probados siguiendo las conclusiones obtenidas tras las experimentaciones realizadas en el Capítulo 4.4.

Tabla 5.2: Capacidad predictiva en test utilizando 50 %, 100 % y 200 % como porcentajes de datos aumentados, y la diferencia relativa máxima ($\widehat{RD}$) para todos los problemas de referencia ordenados según el número de instancias. Se aplicaron todos los métodos estudiados y, para el que produjo el menor error de validación, se muestran los resultados de test obtenidos.

		Aumento de Datos			
Conjunto	Original	50 %	100 %	200 %	$\widehat{RD}$
iris	0.9126	0.9323	0.9327	**0.9427**	3.295
wine	0.9633	**1.0000**	**1.0000**	**1.0000**	3.810
sonar	0.6245	0.7519	**0.7781**	0.7676	24.595
seeds	0.8917	**0.8938**	0.8921	0.8933	0.237
glass	0.6351	0.6572	**0.6749**	0.6658	6.264
cleveland	0.8493	0.8636	0.8690	**0.8769**	3.248
ionosphere	0.8862	0.8945	0.9051	**0.9118**	2.892
horse colic	0.6574	0.6593	0.6660	**0.6697**	1.875
wdbc	0.9705	**0.9789**	0.9778	0.9769	0.870
non bal scale	0.9835	0.9869	0.9855	**0.9886**	0.521
bal scale	0.9258	0.9252	0.9250	**0.9441**	1.975
soybean	0.9241	**0.9283**	0.9281	0.9256	0.457
card	0.8536	0.8591	0.8541	**0.8618**	0.962
breast cancer	0.9736	**0.9754**	0.9754	0.9741	0.188
pima diabetes	0.7537	0.7566	**0.7578**	0.7555	0.543
geneN	0.8499	0.8833	**0.8839**	0.8768	3.996
gene	0.9008	**0.9051**	0.9028	0.8985	0.479
waveform	0.8525	**0.8634**	0.8630	0.8632	1.273
thyroid	0.9780	**0.9808**	0.9806	0.9767	0.281
promedio	0.8624	0.8787	0.8817	**0.8826**	3.040

La Tabla 5.2 muestra los resultados obtenidos para los 19 conjuntos de datos aplicados en la experimentación. La primera columna ('Original') muestra la capacidad de predicción con el conjunto de test cuando no se aplica

UNIVERSIDAD DE MÁLAGA

ningún aumento de datos (DA), mientras que las siguientes columnas muestran los resultados obtenidos para diferentes valores del tamaño de los datos de entrenamiento aumentados (50 %, 100 %, 200 %). Para cada conjunto de datos, se selecciona el modelo generador que obtiene el valor de validación más alto (error más bajo), manteniendo la independencia con el conjunto de test. De este modo, los resultados mostrados en la Tabla 5.2 para el DA, corresponden con la evaluación de la capacidad predictiva en el conjunto de prueba con el modelo generador seleccionado. La última columna de la tabla ($\widehat{RD}$) muestra la diferencia relativa calculada sobre los resultados máximos obtenidos anteriormente (la fuente en negrita indica los mejores valores para cada conjunto de datos). La diferencia relativa (RD) ya ha sido utilizada en la investigación, e indica el aumento porcentual cuando el conjunto de datos aumentados se usa en comparación con los datos originales (Ecuación 5.3).

$$\mathrm{RD} = \frac{(\mathrm{Acc_Aug} - \mathrm{Acc_Ref})}{\mathrm{Acc_Ref}} \times 100 \tag{5.3}$$

Los resultados indican una diferencia relativa media de 3.04 teniendo en cuenta los mejores resultados posibles obtenidos (indicados en negrita en la Tabla 5.2). La mejora en capacidad de predicción es sustancial para algunos de los conjuntos de datos, donde 9 de ellos obtienen un valor de $\widehat{RD}$ superior al 1.8 %. Si únicamente se tienen en cuenta los resultados para cada nivel de validación, para 8 de 19 conjuntos se obtiene el mejor resultado con un porcentaje de aumento del 50 %, e idéntico resultado con un porcentaje de 200 %, si bien la mejora media es mayor con este último alcanzando un 88.26 % de porcentaje de aciertos. Un análisis del número de instancias de los conjuntos de datos correspondientes a los mejores valores obtenidos muestra que para el 50 % de DA la media de instancias es de 2214.3 (std. 2476.2), para 100 % DA es de 908.6 instancias (std. 1154.4), y para 200 % es de 404.6 instancias (std. 190.5). Esto indica que los conjuntos con menor número de instancias se benefician más al usar mayores porcentajes de DA.

Para analizar más a fondo la influencia del tamaño de los conjuntos de datos en la capacidad predictiva obtenida, se muestra en la Figura 5.8 la diferencia relativa máxima $\widehat{RD}$ (valores mostrados en la última columna de la Tabla 5.2) *versus* el número de instancias en una escala logarítmica en el eje x. Se ajustó un modelo de regresión lineal a los resultados para observar la tendencia, obteniendo la siguiente ecuación: $y = -0.59x + 5.69$, con un coeficiente de correlación de -0.38, lo que indica una correlación negativa moderada entre el número de instancias y la ganancia de capacidad de la predicción. Esto apoya la hipótesis de que la aplicación de DA es más beneficiosa en aquellos problemas con un número menor de muestras disponibles. Los resultados del conjunto '*sonar*' son excluidos de este análisis debido a que los valores de RD son excesivamente grandes, hecho que puede explicarse por la baja capacidad de generalización obtenida con los datos originales.

UNIVERSIDAD DE MÁLAGA

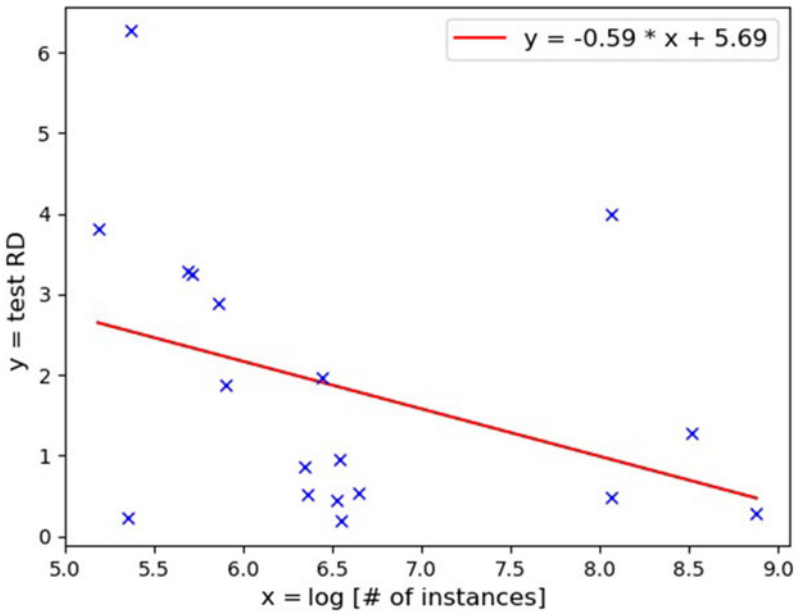

Figura 5.8: Diferencia relativa (RD) de capacidad de predicción frente al logaritmo del número de instancias utilizando los valores de la Tabla 5.2. La línea continua es una regresión lineal ajustada a los datos.

Este tipo de resultado se observó previamente para este conjunto de datos, tomando como referencia el trabajo de Gorman y Sejnowski (1988). La explicación es que algunos patrones no se encuentran bien representados en los conjuntos de entrenamiento o test según la división realizada.

De forma complementaria, se realiza un análisis para observar cuál de los modelos conduce a los mejores resultados para cada conjunto de datos. Estos resultados se muestran en la Tabla 5.3 para los diferentes porcentajes de aumento de los datos de entrenamiento (50 %, 100 %, 200 %) y para todo el conjunto de problemas de referencia ordenados según el número de instancias. Gracias a esta tabla se puede observar como difiere el modelo seleccionado según el conjunto de datos, por lo que no parece haber un acuerdo en el mejor modelo a aplicar. Sin embargo, para cada conjunto de datos si se observa cierta tendencia similar. Para 9 conjuntos de datos, el modelo generador seleccionado es el mismo, teniendo en cuenta alguna de sus variantes (estándar, o 'Balanced Multiclass'), y para 6 conjuntos adicionales, el modelo seleccionado se repite al menos una vez.

Para poder visualizar mejor la elección de modelo con independencia del conjunto de datos, un resumen de estos se muestra en la Tabla 5.4. Para cada uno de los seis modelos, se analizaron las tres variaciones diferentes, donde "S" se refiere al modelo estándar, "M" se refiere al modelo modificado denominado 'Multiclass', mientras que "BM" se refiere al modelo modifica-

UNIVERSIDAD DE MÁLAGA

Tabla 5.3: Modelos que conducen a los mejores resultados de validación cuando se utilizan un 50 %, 100 % y 200 % de datos aumentados.

	Aumento de Datos		
Conjunto	50 %	100 %	200 %
iris	GAN_BM	WGAN-GP_M	**WGAN-GP**
wine	**WGAN-GP_M**	**WGAN-GP_M**	**WGAN-GP_M**
sonar	GAN_M	**WGAN-GP_M**	WGAN-GP_M
seeds	**CVAE_M**	CVAE_BM	CVAE_M
glass	CVAE_BM	**CVAE_BM**	CVAE_M
cleveland	WGAN-GP_M	WGAN-GP_M	**WGAN-GP_M**
ionosphere	CGAN_M	GAN_BM	**GAN_BM**
horse colic	DVAE_BM	CVAE_BM	**GAN_BM**
wdbc	**GAN_M**	CGAN_BM	CVAE_M
non bal scale	CVAE_M	CVAE_M	**CVAE_BM**
bal scale	DVAE_M	VAE_M	**CVAE**
soybean	**CGAN_BM**	WGAN-GP_M	CVAE_M
card	GAN	DVAE	**GAN_M**
breast cancer	**DVAE**	DVAE	DVAE
pima diabetes	CVAE_M	**CGAN_M**	CVAE_BM
geneN	CVAE	**CVAE**	CVAE
gene	**CGAN_BM**	CGAN_BM	CGAN_BM
waveform	**GAN_M**	GAN_M	GAN_M
thyroid	**CVAE_BM**	CGAN	CVAE

do denominado 'Balanced Multiclass'. La primera columna de la tabla corresponde a los diferentes modelos probados y los esquemas de aumento de muestreo, mientras que la última columna informa de la capacidad predictiva del modelo elegido promediada en los 19 problemas. El modelo WGAN-GP con el esquema 'Multiclass' conduce a 10 mejores resultados de los 57 casos analizados y a la mejor capacidad predictiva general con una media en la predicción de 0.877. La segunda opción más seleccionada es el modelo CVAE, también con el uso del esquema 'Multiclass', que condujo a la obtención de 8 mejores casos, pero destacando que en promedio la capacidad predictiva obtenida es mucho menor (0.869).

De esto último cabe destacar que la selección de los modelos no implica un mejor rendimiento, obteniendo el modelo WGAN-GP con el esquema 'Balanced Multiclass' una capacidad de predicción media de 0.876, pero no siendo seleccionada ninguna vez. Esto puede estar relacionado con el hecho de que la selección se realiza sobre resultados de validación, cuando el resultado sobre test puede ser mayor con otro modelo. Otro resultado a mencionar es el hecho de que para los modelos condicionales (CVAE y CGAN) que incluyen la etiqueta de clase en la generación de los datos, los modelos balanceados no tienen casi efecto positivo. Los valores con el esquema 'Multiclass' es menor en ambos casos, y con el esquema 'Balanced Multiclass' se obtienen valores

UNIVERSIDAD DE MÁLAGA

Tabla 5.4: Número de veces que los diferentes modelos probados se eligen como mejores modelos (menor error de validación). El número total de casos (57) corresponde a los 19 conjuntos de datos considerados para los tres diferentes porcentajes de DA.

Modelo		Aumento de Datos 50 %	100 %	200 %	Mejor caso	Acc Test
VAE	S	0	0	0	0	0.867
	M	0	1	0	1	0.871
	BM	0	0	0	0	0.870
DVAE	S	1	2	1	4	0.869
	M	1	0	0	1	0.870
	BM	1	0	0	1	0.869
CVAE	S	1	1	3	5	0.870
	M	3	1	4	8	0.869
	BM	2	3	2	7	0.870
GAN	S	1	0	0	1	0.871
	M	3	1	2	6	0.874
	BM	1	1	2	4	0.874
WGAN-GP	S	0	0	1	1	0.871
	M	2	5	3	10	0.877
	BM	0	0	0	0	0.876
CGAN	S	0	1	0	1	0.874
	M	1	1	0	2	0.872
	BM	2	2	1	5	0.874

similares, 0.87 para CVAE y 0.874 para CGAN. Esto podría indicar que los modelos condicionales sí son capaces de realizar una generación buena de las muestras para las diferentes clases, sin necesidad de que el conjunto de entrenamiento esté balanceado para las clases.

Tabla 5.5: Modelo seleccionado como mejor opción y sus resultados para tres casos: considerando todos los conjuntos de datos juntos o seleccionando el modelo para dos grupos de problemas según el número de instancias.

Datos	Método	$\overline{\#inst}$	Acc Val	RD	Acc Test	RD
Sets 1-19	CVAE_BM_{50}	1271	0.92	0.40	0.87	1.03
Sets 1-9	WGAN-GP_BM_{100}	283	0.90	0.73	0.85	4.12
Sets 10-19	VAE_M_{100}	2259	0.94	0.43	0.91	0.77

Un análisis cuidadoso de los resultados que se muestran en la Figura 5.8 y los de la Tabla 5.4 lleva a la conclusión de que, para los problemas analizados, los conjuntos con menor número de instancias se comportan de manera diferente a aquellos con mayor número de patrones. Por lo tanto, se han considerado diferentes escenarios para intentar llegar a una conclusión sobre cuál puede considerarse el método de mejor rendimiento.

UNIVERSIDAD DE MÁLAGA

Si solo se debe aplicar un método a todo el conjunto de datos, nuestros resultados indican que la mejor opción es el modelo CVAE con el esquema 'Balanced Multiclass' y 50 % de DA. La primera fila de la Tabla 5.5 indica, para este caso, el número promedio de instancias para todo el conjunto de problemas, los resultados de validación basados en los cuales se seleccionó el método elegido y los resultados de test. Para ambos casos (validación y test) se indican la capacidad predictiva y los valores RD %.

Se pueden obtener mejores resultados si el conjunto de problemas se divide en dos grupos según el número de instancias disponibles, y en este caso WGAN-GP con el esquema 'Balanced Multiclass' con 100 % DA funciona mejor que el resto para problemas con un número medio de instancias de 283, lo que lleva a un RD relativamente grande de 4.12 %. Para problemas con un mayor número de instancias (media igual a 2259), los VAE estándar parecen ser la mejor opción, en este caso utilizando el esquema 'Multiclass 'y el 100 % de DA, lo que lleva a una prueba RD % de 0.77. El resultado promedio para la estrategia combinada conduce a una capacidad predictiva en test de 0.88 y 2.36 de RD %, un poco mayor que en el caso de elegir un solo método (0.87 y 1.03).

Todos los resultados anteriores se obtuvieron utilizando la arquitectura neuronal que se muestra en la Figura 5.5. A modo de comparación, también se implementó un clasificador de modelo de regresión logística (LR). Los resultados obtenidos para los 19 conjuntos de datos se muestran en la Tabla 5.6. La tabla muestra los valores de capacidad predictiva en test medidos con el conjunto original sin DA, los resultados para aquellos modelos seleccionados por obtener los con valores de validación más altos, teniendo en cuenta los diferentes niveles de DA aplicado al conjunto de entrenamiento (50 %, 100 %, 200 %), y la diferencia relativa máxima ($\widehat{RD}$) obtenida.

En comparación con los resultados obtenidos con la red neuronal, el modelo LR conduce a un valor promedio de RD mayor (4.146), lo que indica una mejora mayor gracias a la aplicación de DA. Sin embargo, si se observa la capacidad predictiva promedio máxima de test, esta se obtiene aplicando un 50 % de DA con un valor de 0.8642, menor que el obtenido de los modelos de aprendizaje profundo cuando se utiliza DA, y solo ligeramente superior (0.8624) al valor de capacidad predictiva obtenido con el conjunto original. Con respecto a la comparación entre los modelos con los que se han obtenido mejores resultados, el uso del modelo WGAN-GP arrojó 18 mejores resultados, seguido de CGAN y GAN con 14 y 12 mejores resultados para cada modelo, por lo que las variaciones de los modelos GAN representan las mejores opciones en 44 de los 57 casos analizados. En cuanto a las modificaciones, el uso de modelos 'Balanced Multiclass' conduce a 22 mejores resultados, y los modelos 'Multiclass' conducen a 21 mejores resultados para cada modelo, por lo que nuestra propuesta para lidiar con problemas de desequilibrio de clases representa las mejores opciones en 43 de 57 casos.

UNIVERSIDAD DE MÁLAGA

Tabla 5.6: Capacidad predictiva en test obtenida cuando se utiliza como clasificador un modelo de regresión logística con los datos originales y los diferentes casos de datos aumentados.

		Aumento de Datos			
Conjunto	Original	50 %	100 %	200 %	$\widehat{RD}$
iris	0.7500	0.9100	0.9207	**0.9317**	24.222
wine	0.9667	0.9894	**0.9964**	0.9950	3.075
sonar	0.7457	**0.7629**	0.7590	0.7533	2.299
seeds	0.8333	**0.8802**	0.8681	0.8433	5.629
glass	0.5163	0.6263	**0.6442**	0.6347	24.775
cleveland	0.9230	**0.9328**	0.9301	0.9293	1.066
ionosphere	0.8803	0.8890	0.8855	**0.8901**	1.120
horse colic	0.6836	0.6914	**0.7130**	0.7096	4.309
wdbc	0.9430	0.9462	0.9451	**0.9495**	0.688
non bal scale	0.9371	0.9351	**0.9366**	0.9379	0.092
bal scale	0.8608	**0.8996**	0.8772	0.8718	4.507
soybean	0.9241	0.9258	0.9300	**0.9335**	1.019
card	0.8703	**0.8883**	**0.8883**	0.8823	2.073
breast cancer	0.9629	0.9641	**0.9649**	0.9644	0.215
pima diabetes	0.7630	**0.7795**	0.7738	0.7734	2.170
geneN	0.7005	**0.7049**	0.6741	0.6778	0.630
gene	**0.8976**	0.8898	0.8864	0.8946	-0.342
waveform	0.8592	**0.8618**	0.8617	0.8610	0.301
thyroid	0.9347	**0.9434**	0.9425	0.9416	0.929
promedio	0.8396	**0.8642**	0.8630	0.8618	4.146

Adicionalmente, se llevó a cabo un experimento utilizando solo datos sintéticos para realizar el entrenamiento, con el fin de analizar la calidad de las muestras generadas. Específicamente, para obtener la capacidad de predicción para estos casos, se utilizaron datos sintéticos para entrenar y validar la red neuronal feedforward utilizando una distribución de muestras igual a la presente en los datos originales. El conjunto de test es el mismo reservado para realizar las pruebas en los experimentos anteriores. La Tabla 5.7 muestra la capacidad predictiva obtenida en el conjunto de datos de test cuando en el entrenamiento solo se utilizan los datos originales, solo los datos sintéticos y cuando se utilizan los conjuntos aumentados. Estos últimos resultados son los valores máximos alcanzados usando los porcentajes 50 %, 100 % y 200 % de datos aumentados, que corresponden a los casos analizados previamente mostrados en la Tabla 5.2. La fuente en negrita indica el mejor valor de capacidad predictiva obtenida entre los datos originales, sintéticos y aumentados, mostrando claramente que en la mayoría de los casos (15 de 19) es beneficioso usar los conjuntos aumentados. De forma inesperada, para 4 conjuntos de datos se obtuvieron valores más altos usando solo las muestras sintéticas creadas. Los análisis realizados a estos 4 casos no determinan la razón exacta de que obtengan una mejor capacidad de predicción.

UNIVERSIDAD DE MÁLAGA

Tabla 5.7: Capacidad predictiva en test usando los datos originales y las muestras sintéticas generadas, y la mejor capacidad predictiva usando un 50 %, 100 % y 200 % de datos aumentados.

Conjunto	Original	Sintéticos	Data Aug.
iris	0.9126	0.9337	**0.9427**
wine	0.9633	0.9936	**1.0000**
sonar	0.6245	0.5312	**0.7781**
seeds	0.8917	**0.9143**	0.8938
glass	0.6351	0.5979	**0.6749**
cleveland	0.8493	0.8115	**0.8769**
ionosphere	0.8862	0.8821	**0.9118**
horse colic	0.6574	**0.6718**	0.6697
wdbc	0.9705	0.9694	**0.9789**
non bal scale	0.9835	0.9762	**0.9886**
bal scale	0.9258	0.9154	**0.9441**
soybean	0.9241	0.9088	**0.9283**
card	0.8536	**0.8720**	0.8618
breast cancer	0.9736	0.9729	**0.9754**
pima diabetes	0.7537	0.7577	**0.7578**
geneN	0.8499	0.7533	**0.8839**
gene	0.9008	0.6878	**0.9051**
waveform	0.8525	**0.8657**	0.8634
thyroid	0.9780	0.9174	**0.9808**
promedio	0.8624	0.8386	**0.8851**

Tabla 5.8: Capacidad predictiva en test obtenida con los datos originales y el promedio de los porcentajes de DA utilizados para los diferentes modelos generadores estándar, sobre los conjuntos del ámbito médico analizados.

Datos	Original	VAE	DVAE	CVAE	GAN	WGAN	CGAN
cleveland	0.8493	0.8236	0.8492	**0.8609**	0.8422	0.8414	**0.8609**
horse colic	0.6574	**0.6866**	0.6842	0.6798	0.6621	0.6662	0.6817
wdbc	0.9705	0.9662	0.9704	0.9676	0.9748	0.9749	**0.9765**
card	0.8536	0.8579	0.8494	0.8561	0.8590	0.8440	**0.8706**
breast	0.9736	0.9747	0.9750	0.9742	0.9755	0.9730	**0.9756**
diabetes	0.7537	0.7598	**0.7609**	0.7555	0.7497	0.7458	0.7586
geneN	0.8499	0.8392	0.8433	**0.8813**	0.8468	0.8466	0.8800
gene	0.9008	0.8985	0.8995	0.8995	0.8963	0.8955	**0.9012**
thyroid	0.9780	0.9772	0.9767	0.9766	0.9533	0.9478	**0.9801**
promedio	0.8652	0.8648	0.8676	0.8724	0.8622	0.8595	**0.8761**

Dado que nuestro propósito es aplicar DA con los modelos generativos profundos estudiados sobre problemas del ámbito de la bioinformática, se realizó un análisis de resultados generales sobre los conjuntos analizados que están recogidos del ámbito médico. La Tabla 5.8 muestra los resultados de capacidad predictiva obtenidos para los 9 conjuntos relacionados con un

UNIVERSIDAD DE MÁLAGA

problema médico, con los datos originales sin aumentar y el promedio de capacidad predictiva obtenida con los modelos generativos en su esquema estándar para los tres porcentajes de aumento de muestras aplicados (50 %, 100 % y 200 %). Se han acortado los nombres de los conjuntos '*breast cancer*' y '*pima diabetes*' para que la longitud de la tabla no excediera los márgenes de la hoja de forma excesiva. Para 6 de 9 conjuntos, los mejores resultados se obtienen con el modelo CGAN, que a su vez obtiene el mayor valor de capacidad predictiva con 0.8761, lo que corresponde a un 1 % respecto a la capacidad de predicción obtenida con el conjunto original. El segundo mejor modelo es la CVAE, con un valor de 0.8724 de capacidad predictiva y los mejores valores para 2 conjuntos de datos. Esto nos indica que la ganancia con este tipo de problemas es más difícil, y que los modelos condicionales, gracias a la inclusión de la etiqueta de clase, obtienen de base mejores resultados para estos conjuntos.

La Figura 5.9 muestra los tiempos de cálculo (en segundos y en escala logarítmica) que han sido necesarios para los procesos de generación y clasificación de datos sintéticos, graficados en función del número de instancias más el número de características de los conjuntos de datos de referencia. Representado por una línea que consta de guiones y puntos, se encuentra el tiempo de cálculo necesario para realizar el proceso de clasificación utilizado en la experimentación si solo se toma un fold. Tomando mayores valores de

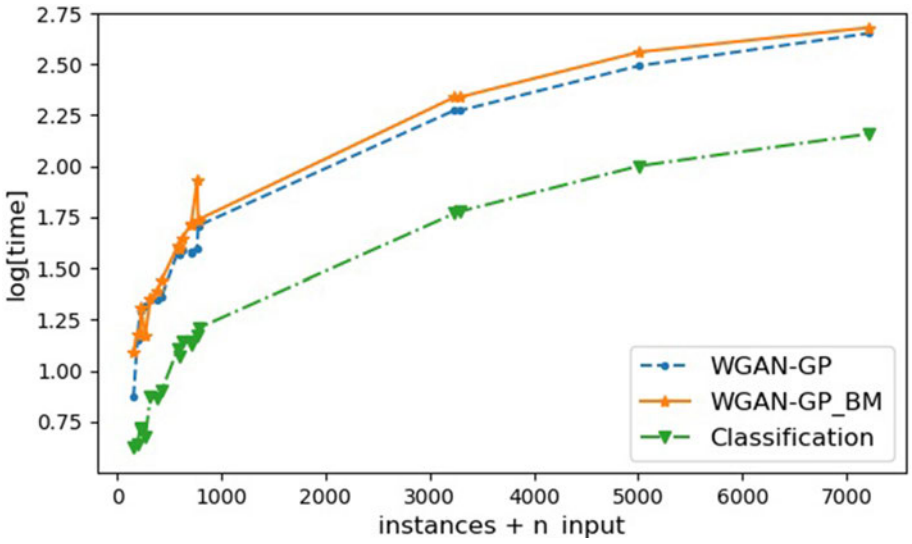

Figura 5.9: Tiempo de cálculo utilizado en los diferentes experimentos. La línea discontinua representa el proceso de generación usando el modelo WGAN-GP, y la línea continua el modelo WGAN-GP con el esquema 'Balanced Multiclass' con 100 % DA. El tiempo empleado en el proceso de clasificación se representa por una línea de guiones y puntos.

UNIVERSIDAD DE MÁLAGA

tiempo empleado, se muestra el tiempo de cálculo necesario para la generación de los datos sintéticos con el modelo WGAN-GP representado por una línea discontinua, y el tiempo necesario para la generación con el modelo WGAN-GP_BM representado por una línea continua. En ambos casos, no se incluye el tiempo empleado para entrenar el modelo clasificador generativo, por lo que el tiempo total para realizar la generación es aún mayor. Se ha utilizado el número de instancias más el número de características de los datos para poder distinguir los valores obtenidos para los conjuntos de datos '*gene*' y '*geneN*', ya que estos presentan el mismo número de muestras, pero un número de características diferente.

Para dar una idea más clara de los tiempos de cálculo involucrados, informamos que el tiempo promedio en los 19 conjuntos de datos aplicados en el estudio, para la ejecución de un experimento (1 fold) con un conjunto de parámetros fijos aplicando DA con el modelo WGAN-GP, es de 69 segundos. Además, si se quisieran repetir todos los experimentos que dan lugar a los resultados reportados en este capítulo (esto excluye diferentes pruebas realizadas para afinar los modelos de DA, optimización de parámetros utilizados en las redes neuronales, etc.) el tiempo total de cálculo es de 76.37 días. Todos los experimentos se ejecutaron en un PC con Windows 10 64-bits OS, equipado con una CPU Intel Core i7-6700K 4.00 GHz, 32 GB RAM, y una NVIDIA® Titan Xp GPU con 12 GB RAM. El software utilizado para realizar los experimentos fue CUDA® Toolkit 9.0, cuDNN v7.0, Python v3.6.2, y TensorFlow v1.8 con soporte GPU (Abadi et al., 2016).

5.5. Conclusiones

En este capítulo se han aplicado diferentes tipos de modelos VAE y GAN para la generación de datos aumentados mediante el proceso clasificador generativo desarrollado, siendo este el método más relevante, pues permite el uso de múltiples modelos generativos en una generación con etiquetado para tareas supervisadas. También se han introducido dos modificaciones a la distribución de clases presentes en las muestras originales, intentando aumentar la capacidad de clasificación (modificaciones denominadas 'Multiclass' y 'Balanced Multiclass'). Se realizó un extenso número de simulaciones experimentales utilizando diecinueve conjuntos de datos bien conocidos de problemas multiclase y de tamaño reducido (número de instancias por clase inferior a 1000).

Como resultado relevante, el DA puede conducir a un aumento en la capacidad de predicción en test de aproximadamente un 3 %, si se evalúan todos los métodos probados y se elige el mejor según el error de validación. Si solo se aplicara un método, los resultados muestran que el modelo de mejor desempeño entre todos los conjuntos de datos probados es el CVAE con el esquema 'Balanced Multiclass' y 50 % de DA. Sin embargo, se pueden obte-

UNIVERSIDAD DE MÁLAGA

ner mejores resultados si los problemas se agrupan de acuerdo con el número de instancias disponibles; en este caso, WGAN-GP con el esquema 'Balanced Multiclass' con 100 % DA es la mejor opción para problemas con un menor número de muestras que conducen a un RD de 4.12 %, mientras que el VAE estándar con el esquema 'Multiclass' y 100 % DA es la mejor alternativa para conjuntos de datos con un número de muestras superior a 600 (número medio igual a 2259) (Tabla 5.5). Sin embargo, vale la pena señalar que los resultados muestran que no todos los conjuntos de datos se benefician en el mismo grado, y se observaron resultados con valores más bajos cercanos al 0.2 % y valores muy altos hasta un aumento del 24.6 % en la capacidad de predicción. La Figura 5.10 resume los resultados principales para la capacidad predictiva en test obtenida con las muestras de entrenamiento originales y aumentadas, agrupadas en cuatro casos: todos los conjuntos de datos considerados juntos, los nueve conjuntos con el menor número de instancias, los diez restantes conjuntos con el mayor número de muestras, y los nueve conjuntos relacionados con problemas del ámbito de la medicina.

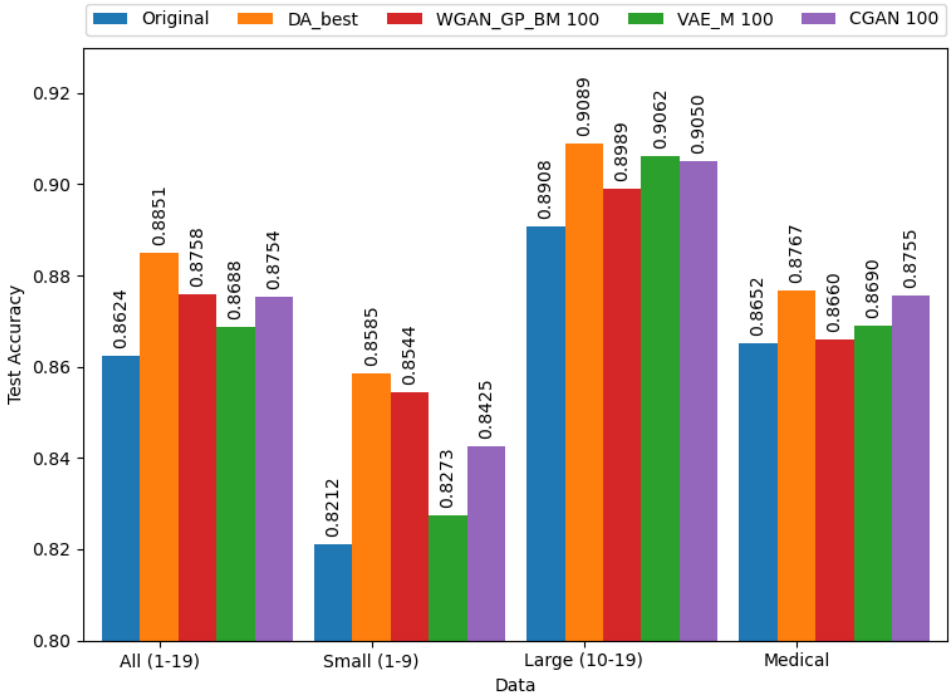

Figura 5.10: Capacidad predictiva utilizando los datos originales, la mejor opción para DA, el modelo WGAN-GP con 'Balanced Multiclass' 100 % DA, el modelo VAE con 'Multiclass' 100 % DA y el modelo CGAN con 100 % DA. Los resultados se indican para cuatro grupos: todos los conjuntos; agrupados según el número de instancias; y del ámbito médico.

UNIVERSIDAD DE MÁLAGA

La conclusión general que se puede extraer de los resultados obtenidos es que las técnicas de DA basadas en VAE y GAN constituyen enfoques útiles para aumentar la capacidad de predicción cuando se analizan pequeños conjuntos de datos, ya que generan datos de buena calidad, que conducen en promedio a un incremento relativo de predicción del 3 %. También que los modelos condicionales (CVAE y CGAN), gracias a la inclusión de la etiqueta de clase en el proceso de generación, parecen ser capaces de generar muestras de una calidad mayor, especialmente en su aplicación con problemas relacionados con la medicina, y se ven menos influenciados por las modificaciones del balanceo de los datos. En relación a esto, los experimentos sirvieron para plantear la aplicación de los métodos desarrollados a datos genómicos, un problema desafiante en biomedicina donde generalmente hay pocas muestras disponibles, y para el cual, hasta ese momento, no se habían obtenido grandes ventajas de la aplicación de problemas de aprendizaje profundo (Liu et al., 2019).

UNIVERSIDAD DE MÁLAGA

UNIVERSIDAD
DE MÁLAGA

Parte III

Aplicación en Bioinformática

UNIVERSIDAD
DE MÁLAGA

UNIVERSIDAD
DE MÁLAGA

Capítulo 6

Predicción de eventos en cáncer

Resumen: En el campo de la biomedicina, los modelos de aprendizaje profundo (DL) ofrecen resultados excepcionales en tratamientos relacionados con imágenes histológicas, escaneos y tomografías. Sin embargo, cuando se aplican técnicas similares a datos de expresión génica, su rendimiento es limitado, lo que se ve obstaculizado aún más por la complejidad de estos modelos que requieren una gran cantidad de muestras para proporcionar resultados precisos. Debido a la difícil y costosa recolección de datos médicos, la aplicación de técnicas de aumento de datos (DA) es un tema de gran importancia en los problemas que actualmente aborda la bioinformática. Mediante la aplicación de modelos de vanguardia basados en modelos GAN condicionales y las modificaciones adaptativas desarrolladas, se investiga el efecto del DA para predecir el estado vital de un paciente con cáncer a partir de datos de expresión génica de RNA-Seq. Los resultados experimentales en varios conjuntos de datos del mundo real demuestran la eficacia de los modelos propuestos, aumentando significativamente el rendimiento en predicción. Además, se analiza la calidad de las muestras sintéticas generadas mediante el uso de diferentes métricas de similitud entre distribuciones, y aplicando técnicas multivariantes como PCA.

6.1. Introducción al Problema

El cáncer es uno de los mayores problemas médicos a los que se enfrenta la sociedad. Según el Instituto Nacional del Cáncer (NCI) de los Estados Unidos de América, cáncer es el nombre con el que se designa a un conjunto de enfermedades relacionadas por el hecho de que en estas se produce un descontrol celular, donde algunas de las células del cuerpo empiezan a dividirse sin detenerse y se diseminan a los tejidos de alrededor. En todo el mundo,

UNIVERSIDAD DE MÁLAGA

según el estudio realizado por GLOBOCAN (Sung et al., 2021), se detectaron aproximadamente 19 millones de nuevos casos de cáncer y se produjeron casi 10 millones de muertes por esta enfermedad en 2020. El cáncer de mama femenino ha superado al cáncer de pulmón como el cáncer diagnosticado más común, con un estimado de 2.3 millones de nuevos casos (11.7 %), seguido del cáncer de pulmón (11.4 %), colorrectal (10.0 %), de próstata (7.3 %) y de estómago (5.6 %). Si bien, el cáncer de pulmón representa la principal causa de muerte por cáncer, con un estimado de 1.8 millones de fallecimientos (18 %), seguido de los cánceres colorrectal (9.4 %), de hígado (8.3 %), de estómago (7.7 %) y de mama femenino (6.9 %). Debido a los cambios demográficos, como el envejecimiento y el crecimiento de la población, así como cambios en la prevalencia y distribución de los principales factores de riesgo de cáncer, la incidencia y la mortalidad por cáncer está creciendo rápidamente en todo el mundo. Se espera que en 2040 haya en el mundo 28.4 millones de casos de cáncer, representando un aumento del 47 % con respecto a 2020, con un aumento mayor en los países en vías de desarrollo (64 % a 95 %) respecto a los países desarrollados (32 % a 56 %) debido a los posibles cambios demográficos. Estas estimaciones pueden agravarse más por el aumento de los factores de riesgo asociados con la globalización.

En 2020 y 2021, se han producido importantes demoras en el diagnóstico y el tratamiento del cáncer por la pandemia del COVID-19 (Kutikov et al., 2020; Maringe et al., 2020). Entre otras cuestiones, el acceso reducido a los sistemas de salud debido a los cierres de establecimientos de atención médica puede dar como resultado graves retrasos en el diagnóstico y el tratamiento que pueden conducir a una disminución a corto plazo en la incidencia de cáncer, pero seguida de un aumento de casos de la enfermedad en etapa avanzada y, en última instancia, un aumento de la mortalidad. Sin embargo, esta consecuencia secundaria de la pandemia tardará varios años en cuantificarse. Por eso, es importante la creación de nuevos sistemas de diagnóstico eficaces y la mejora de los sistemas actuales.

En este sentido, la predicción de eventos en cáncer, como la detección y la predicción de supervivencia, incluida la predicción del tiempo hasta la muerte de un paciente, siguen siendo problemas desafiantes en el dominio clínico incluso después de décadas de esfuerzo por parte de los investigadores (Nicholson et al., 2001; Kourou et al., 2015). En términos de supervivencia, de forma general suele realizarse la predicción de esta en relación al tiempo transcurrido desde el diagnóstico de la enfermedad. El modelo estadístico más comúnmente utilizado es el modelo de riesgo proporcional de Cox (Cox-PH) (Cox, 1972), que permite el manejo de muestras censurada, muestras cuyo estado es desconocido. Para lidiar con estas muestras, el método de análisis de supervivencia se centra en si un paciente sobrevive en un determinado momento, en lugar de cuando ha fallecido el paciente. En la investigación presentada en este capítulo, el propósito no es realizar este

UNIVERSIDAD DE MÁLAGA

tipo de predicción de supervivencia. En cambio, se centra en la predicción del evento en intervalo de tiempo fijo. De este modo, la aplicación se encuentra en un estado intermedio del método anteriormente explicado, tratando con una tarea de clasificación sin muestras censuradas.

Así pues, el foco está en predecir el estatus vital del paciente al finalizar el estudio, siendo este el evento. Se tienen dos posibilidades de estatus vital, el paciente se encuentra vivo o falleció por causas derivadas de la enfermedad. Se tiene de esta forma el objetivo adicional de predecir la agresividad del cáncer sin tener en cuenta el tiempo transcurrido de enfermedad. Con este propósito se analizan únicamente perfiles de expresión génica de mRNA secuenciado de pacientes con cáncer. El increíble progreso tecnológico de las últimas dos décadas ha derivado en la accesibilidad de datos transcriptómicos de alto rendimiento de pacientes con cáncer. Los investigadores han descubierto que los datos de transcriptómica, especialmente los perfiles de expresión génica, pueden ser muy útiles para el análisis del cáncer (Valdes-Mora et al., 2018). Los perfiles de expresión génica se obtienen de métodos de laboratorio que identifican todos los genes de una célula que están produciendo RNA mensajero (Wang et al., 2009). Las moléculas de RNA mensajero (mRNA) transportan la información genética necesaria para producir proteínas desde el ADN en el núcleo de la célula hasta el citoplasma donde se producen las proteínas. Gracias a esta información encapsulada, se puede usar un perfil de expresión genética para diagnosticar las enfermedades y para predecir la respuesta de un paciente a determinado tratamiento.

Dado el objetivo y los medios, como modelo para realizar la tarea de predicción no se utiliza el modelo Cox-PH, dado que este se basa en una combinación lineal de características determinadas, lo que impide un aprendizaje eficaz de los procesos biológicos no lineales subyacentes a partir de datos transcriptómicos para el pronóstico del cáncer. En su lugar, se utiliza de manera común modelos de redes neuronales. La primera red neuronal para este propósito fue desarrollado por Faraggi y Simon (1995), y hoy en día, gracias al aprendizaje profundo (DL), se pueden utilizar datos de entrada genómicos para la predicción de este tipo de eventos, aunque sigue siendo un desafío (Ching et al., 2018; Kim et al., 2020). Principalmente porque, como ya ha sido comentado y representa la razón del trabajo de tesis, la cantidad de datos disponibles para los modelos de aprendizaje profundo de pacientes en problemas de la bioinformática es insuficiente, lo que genera graves problemas de sobreajuste en los modelos. Para ello se va a estudiar el impacto generado por los métodos de DA para aumentar de tamaño los datos genómicos disponibles, intentando mejorar el rendimiento en predicción del estatus vital en intervalo de tiempo fijo en cáncer. Concretamente, se realizará un estudio con 18 tipos de cáncer, detallados a continuación.

- Carcinoma de vejiga (*bladder urothelial carcinoma*) (blca): El carcinoma urotelial de vejiga es el tipo más común de cáncer de vejiga. Este

UNIVERSIDAD DE MÁLAGA

cáncer invasivo se desarrolla en el revestimiento interno de la pared de la vejiga, y es más común entre los hombres que entre las mujeres. Si el cáncer se diagnostica después de la metástasis, la supervivencia después de cinco años es poco probable.

- Carcinoma de mama (*breast invasive carcinoma*) (brca): El cáncer de mama es el tipo de cáncer que se diagnostica con más frecuencia, y la segunda causa de muerte por cáncer en las mujeres. Debido a la detección temprana mediante el uso de mamografías y las mejoras en el tratamiento, las muertes por cáncer de mama han disminuido constantemente desde la década de 1990. Este tipo de cáncer invasivo incluye el carcinoma ductal y el carcinoma lobulillar.

- Adenocarcinoma de colon (*colon adenocarcinoma*) (coad): Los adenocarcinomas de colon y recto son los terceros diagnosticados con mayor frecuencia. Cuando el cáncer colorrectal se disemina a otras partes del cuerpo, la supervivencia de los pacientes es muy reducida. El número de fallecidos ha ido disminuyendo gracias a una detección temprana y la extirpación de los pólipos antes de convertirse en cáncer.

- Glioblastoma multiforme (gbm): Este cáncer es un tipo de tumor cerebral maligno de rápido crecimiento. Es el tumor cerebral más común en adultos. Los pacientes con este cáncer tienen un pronóstico desfavorable, con un tiempo de supervivencia reducido, ya que actualmente no existen tratamientos eficaces a largo plazo para esta enfermedad.

- Carcinoma de células escamosas de cabeza y cuello (*head and neck squamous cell carcinoma*) (hnsc): La mayoría de los cánceres de cabeza y cuello comienzan en las membranas mucosas. Estos cánceres afectan con mayor frecuencia a personas mayores de 50 años y es más del doble en los hombres que en las mujeres. Los factores de riesgo conocidos son fumar y beber en exceso.

- Carcinoma de células claras de riñón (*kidney renal clear cell carcinoma*) (kirc): Este es el tipo más común de cáncer de riñón. Cuando se detecta a tiempo, la mayoría de los casos se pueden tratar de forma eficaz. Sin embargo, las tasas de supervivencia son bajas cuando el cáncer se disemina a otras partes del cuerpo.

- Cánceres renales (*pan-kidney cohort*) (kipan): Estos incluyen los carcinomas de células claras (kirc), los de células papilares (kirp) y de células cromófobas (kich). Este último es un tipo poco común de cáncer de riñón que se forma en las células que recubren los pequeños túbulos del riñón. Dado que los diferentes tipos de cáncer de riñón son muy distintos, caracterizar y comprender cada tipo es importante.

UNIVERSIDAD DE MÁLAGA

- Glioma cerebral de grado inferior (*brain lower grade glioma*) (lgg): Este es un tipo de cáncer que se desarrolla en las células gliales del cerebro, las cuales apoyan a las células nerviosas del cerebro y las mantienen saludables. Los tumores se clasifican en grados I, II, III o IV según los estándares establecidos por la Organización Mundial de la Salud. El glioma de grado inferior consta de los grados II y III.

- Carcinoma hepatocelular de hígado (*liver hepatocellular carcinoma*) (lihc): Esta enfermedad surge en los hepatocitos, las células que componen la mayor parte del hígado. Los factores de riesgo más comunes son la hepatitis B o C y la cirrosis. Este tipo de cáncer, más común en hombres, presenta un pronóstico general desfavorable.

- Adenocarcinoma de pulmón (*lung adenocarcinoma*) (luad): El cáncer de pulmón es responsable de más muertes que cualquier otro cáncer tanto en hombres como en mujeres, aproximadamente el 18 % de todas las muertes por cáncer. El pronóstico del cáncer de pulmón es malo; la mayoría de los pacientes con cáncer de pulmón son diagnosticados con cáncer avanzado y su supervivencia es reducida.

- Carcinoma de células escamosas de pulmón (*lung squamous cell carcinoma*) (lusc): Este cáncer comienza generalmente en los bronquios, con un crecimiento más lento que otros tipos de cáncer de pulmón. El carcinoma de células escamosas está estrechamente relacionado con antecedentes de tabaquismo.

- Carcinoma de ovario (*ovarian serous cystadenocarcinoma*) (ov): El cáncer de ovario representa aproximadamente el 3 % de todos los cánceres en las mujeres. Concretamente, el adenocarcinoma seroso de ovario representa el 90 % de todos los cánceres de ovario. Como resultado de la falta de pruebas eficaces de detección, a la mayoría de las mujeres se les diagnostica cáncer avanzado.

- Adenocarcinoma de próstata (*prostate adenocarcinoma*) (prad): El cáncer de próstata se clasifica según la puntuación de Gleason, que se basa en el aspecto de las células al microscopio. Una puntuación de Gleason baja significa que el tejido canceroso es similar a las células normales, y una puntuación de Gleason alta significa que las células cancerosas son muy diferentes y es probable que se diseminen. Para los pacientes cuyo cáncer se ha diseminado, el tiempo de supervivencia es reducido.

- Melanoma cutáneo (*skin cutaneous melanoma*) (skcm): Los factores de riesgo del melanoma incluyen tez clara, antecedentes familiares y exposición prolongada a la luz solar natural o artificial. Concretamente, los melanomas metastásicos, aquellos que se ha diseminado a una parte del cuerpo distinta, son los que se descubren con mayor frecuencia.

UNIVERSIDAD DE MÁLAGA

- Adenocarcinoma de estómago (*stomach adenocarcinoma*) (stad): La incidencia de este tipo de cáncer varía ampliamente según la región geográfica y diferentes factores genéticos. Los factores de riesgo incluyen el tabaquismo, la obesidad e infecciones bacterianas o virales. El cáncer de estómago tiende a diagnosticarse en una etapa avanzada derivando en una baja tasa de supervivencia.

- Carcinoma de estómago y esofágico (*stomach and esophageal carcinoma*) (stes): Los dos tipos más comunes son el cáncer de células escamosas y el adenocarcinoma. Ambos presentan diferentes factores de riesgo, el cáncer de células escamosas aumenta con el tabaquismo y el consumo de alcohol, mientras que el adenocarcinoma se asocia con diversos trastornos gástricos. Al igual que el cáncer de estómago, tiende a diagnosticarse en una etapa avanzada, lo que lleva a una tasa de supervivencia reducida.

- Adenocarcinoma de tiroides (*thyroid adenocarcinoma*) (thca): El adenocarcinoma papilar de tiroides representa el 80 % de todos los casos de cáncer de tiroides. El cáncer de tiroides crece de forma lenta, de forma que, si se realiza un diagnóstico temprano, la posibilidad de supervivencia es muy alta.

- Carcinoma de endometrio (*uterine corpus endometrioid carcinoma*) (ucec): El cáncer de endometrio se desarrolla en las células que forman el revestimiento interno del útero. Este tipo de cáncer se diagnostica frecuentemente en una etapa temprana y, como resultado, tiene una tasa de supervivencia alta.

6.2. Implementación de los modelos

Con el objetivo de comprobar el efecto del DA en el rendimiento predictivo sobre el problema de predicción del estatus vital en cáncer, y realizar una comparación entre diversos métodos de DA, se utilizan la adición de ruido gaussiano, la técnica SMOTE, el modelo CGAN estándar y el modelo CGAN en el marco adaptativo del modelo clasificador generativo expuesto en el Capítulo 5, incluyendo las modificaciones de balanceo de datos. En esta sección se resume la implementación de estos modelos, incluyendo el modelo clasificador utilizado en la experimentación.

En primer lugar, se utiliza el método de adición de ruido gaussiano explicado en el Capítulo 4 sección 4.2. Este método selecciona aleatoriamente muestras del conjunto de entrenamiento y modifican aleatoriamente un 25 % de las características del mismo. El nuevo valor de la característica se calcula añadiendo ruido que surge de una distribución normal aleatoria, con un valor de 0.2 como desviación, al valor original de la característica. Se

UNIVERSIDAD DE MÁLAGA

realiza además un control de los nuevos valores, verificando que no excedan los límites establecidos (máximo y mínimo) para su característica. Gracias a este control, los nuevos valores no se ven afectados por la transformación logarítmica y la normalización realizada en los datos.

En cuanto a la técnica SMOTE aplicada, se utiliza la versión estándar (Chawla et al., 2002), detallada en el Capítulo 2 sección 2.1.3. Este genera como límite superior, tantas muestras como sean necesarias para lograr el balanceo de los datos, por lo que no se puede probar con diferentes niveles de porcentaje de DA. El algoritmo SMOTE utiliza la interpolación de varias instancias vecinas dentro de la clase minoritaria para la creación de nuevas instancias. Para la implementación de esta técnica se utiliza el módulo *imbalanced-learn* (Lemaître et al., 2017) con los valores por defecto (crear muestras para balancear el conjunto, $sampling_strategy =$ '$auto$'; número de vecinos para interpolar, $k_neighbors = 5$).

Los modelos generativos profundos aplicados a esta experimentación se basan en el modelo CGAN (Mirza y Osindero, 2014). Las modificaciones realizadas en el proceso de generación de datos, y que dan lugar tanto al modelo adaptativo con clasificador generativo, como las variaciones de control de balanceo tienen la misma implementación que el modelo CGAN estándar, con la excepción de la implementación específica del clasificador generativo. El modelo adaptativo que se encuentra desarrollado en el Capítulo 5 sección 5.1, se denomina en los resultados de esta investigación como ModCGAN, para diferenciar este del modelo CGAN sin clasificador generativo. Debido a que el modelo CGAN ya realiza la asignación de clases a las muestras generadas, se desea realizar una comprobación para ver las diferencias existentes con nuestro modelo. La Tabla 6.1 muestra los detalles de implementación y arquitectura del modelo CGAN utilizado para el proceso generativo de muestras sintéticas, haciendo una distinción entre la red generador y discriminador. Las columnas muestran el tipo de capa (*"Fully connected"* representa una capa de neuronas totalmente conectada), el tamaño de salida de cada una, que se identifica con el número de neuronas presentes en cada capa, el uso de la normalización por lotes y la función de activación utilizada. En la parte inferior también se muestra el algoritmo de optimización utilizado en ambas redes para entrenar los modelos, mostrando los valores usados como parámetros de aprendizaje. También se muestra el valor usado como pendiente en la activación de Leaky ReLU y el *Backend* usado, que es Tensorflow (Abadi et al., 2016). En este caso, se aplica la convención práctica de que las funciones Leaky ReLU y Parametric ReLU (He et al., 2015) son equivalentes. Si bien se está utilizando una PReLU, pues el valor de pendiente es diferente a 0.01, valor que corresponde a Leaky ReLU.

Por último, se describe la implementación realizada para los modelos clasificadores utilizados tanto para la prueba final de los modelos cuando se utilizan datos aumentados, como para el proceso de generación de muestras

UNIVERSIDAD DE MÁLAGA

Tabla 6.1: Características del generador y discriminador que componen los modelos CGAN y ModCGAN utilizados para generar muestras sintéticas.

Capa	Salida	Norm.	Activación
Generador			
Entrada = [z, y]	128 + 1	-	-
Fully connected	1024	Sí	ReLU
Fully connected	512	Sí	ReLU
Fully connected	256	Sí	ReLU
Fully connected	128	Sí	ReLU
Fully connected	Tamaño Datos	No	Sigmoide
Discriminador			
Entrada = [x, y]	Tamaño Datos + 1	-	-
Fully connected	64	Sí	Leaky ReLU
Fully connected	128	Sí	Leaky ReLU
Fully connected	256	Sí	Leaky ReLU
Fully connected	512	Sí	Leaky ReLU
Fully connected	1	No	Sigmoide
Generador optimizador	Adam (lr = 0.0004, beta1 = 0.5, beta2 = 0.9)		
Discriminador optimizador	Adam (lr = 0.0002, beta1 = 0.5, beta2 = 0.9)		
Leaky ReLU	$\alpha = 0.2$		
Backend	Tensorflow		

sintéticas cuando se utilizan los modelos modificados. El modelo utilizado es una red neuronal profunda, cuyas características se muestran en la Tabla 6.2, el tipo de capa, el tamaño de salida, el uso de la normalización por lotes, el uso de dropout y la función de activación. Se señalan dos vectores de tasa de abandono usada para dropout haciendo una distinción entre el Clasificador Generativo y el clasificador utilizado para los experimentos. Como se puede observar, el número de neuronas seleccionadas es el mismo para todos los conjuntos. Para seleccionar esta arquitectura, neuronas y otros parámetros, se realizó una experimentación previa y un repaso de modelos de la literatura. Se probaron arquitecturas de 3 y 4 capas ocultas, con 64, 128, 256, 512, 1024 y 2048 neuronas posibles para cada capa oculta y tasa de abandono en el rango 0.1/0.7. Para la selección se tuvo en cuenta el rendimiento en predicción sobre validación y el tiempo de cómputo requerido. A este respecto, una práctica común en el diseño de redes profundas es el uso de una arquitectura en pirámide invertida. Se argumenta que la reducción paulatina del número de neuronas obliga a la red a ceder información, intentando eliminar el "ruido" en los datos, mientras trata de mantener tanta información relevante como sea posible. Es probable que esto obligue a la red neuronal a generalizar frente a memorizar, si bien no se ha demostrado con precisión. También sucede comúnmente que esta tipología obtiene un rendimiento en predicción similar a otras arquitecturas, pero a un menor costo computacional al tener una reducción sucesiva de las interconexiones en la red, lo cual sí fue demostrado

UNIVERSIDAD DE MÁLAGA

en la experimentación para su selección.

Tabla 6.2: Características de la red neuronal utilizada como clasificador.

Capa	Salida	Norm.	Dropout	Activación
Entrada = [x]	Tamaño Datos	-	-	-
Fully connected	1024	Sí	Sí	Leaky ReLU
Fully connected	512	Sí	Sí	Leaky ReLU
Fully connected	256	Sí	Sí	Leaky ReLU
Fully connected	2	No	No	Sigmoide
Optimizador	Adam (lr = 0.0002, beta1 = 0.5, beta2 = 0.9)			
Dropout ratio	Clasificador Generativo (0.1, 0.5, 0.3)			
Dropout ratio	Clasificador para test (0.3, 0.6, 0.4)			
Leaky ReLU	$\alpha = 0.2$			
Regularización	regularización L^2 en pesos (scale = 1e-6)			
Backend	Tensorflow			

6.3. Resultados Experimentales

El proceso de experimentación realizado para la aplicación de DA en el problema de predicción de eventos en intervalo de tiempo fijo en cáncer, es similar a la experimentación realizada en el capítulo anterior. Está se encuentra descrita en el inicio del Capítulo 5 sección 5.4. Los experimentos se llevaron a cabo con 18 conjuntos de datos tomados del sitio web *The Cancer Genome Atlas* (TCGA) (Tomczak et al., 2015), proporcionados por el *International Cancer Genome Consortium* (ICGC). La Tabla A.5 del Apéndice A.3 muestra algunas características de los conjuntos de datos utilizados. Una selección parcial de los resultados obtenidos se encuentra publicada en Moreno-Barea et al. (2022c).

Resumiendo la experimentación realizada, se realiza en primer lugar una división del conjunto de datos en conjuntos de entrenamiento y test, para mantener una independencia total entre la generación de datos, el entrenamiento de los modelos de clasificador y la evaluación del rendimiento en clasificación. El conjunto de test se mantiene por separado hasta la experimentación final para obtener un resultado honesto del rendimiento. Seguidamente, se lleva a cabo el proceso de generación de datos sintéticos con el modelo generativo correspondiente, sobre los conjuntos preprocesados con un método de selección de características. Una vez finalizado este proceso, los datos sintéticos se agregan en el porcentaje necesario a los datos de entrenamiento y se implementa un procedimiento de validación cruzada para tener una mejor estimación de los resultados. Se utiliza un procedimiento estratificado para obtener 10 folds, 9 para el entrenamiento y 1 para validación. Los subconjuntos de entrenamiento se incrementan agregando muestras sintéticas, implementando el aumento de datos. Luego se procede con el en-

UNIVERSIDAD DE MÁLAGA

trenamiento del clasificador, y una vez finalizado, se realiza la predicción con los datos de test. El resultado del proceso es el promedio de los resultados obtenidos para los 10 folds. Este proceso se repite 10 veces con diferentes semillas para reducir los efectos aleatorios.

Como métricas para el rendimiento de la predicción del estatus vital se han utilizado la capacidad predictiva (*accuracy*) y el área bajo la curva ROC (AUC). Ambas métricas se encuentran descritas de forma extendida en el Capítulo 3 sección 3.2. Se midió la capacidad predictiva para analizar cómo afectan los modelos de DA de forma directa en el porcentaje de muestras bien clasificadas. Además, la mayoría de los conjuntos están balanceados, para 11 de 18 conjuntos su valor es mayor que 0.9, o no presentan un desbalanceo excesivo, para 14 de 18 es mayor que 0.7, siendo un valor de 1 equivalente a un balanceo perfecto. Igualmente, en el análisis y conclusiones se tiene en cuenta la particularidad de esta métrica, siendo complementada con la medida del AUC. En cuanto a los niveles de DA se comprobó el efecto con porcentajes de creación de muestras del 25 %, 50 %, 100 % y 200 %.

6.3.1. Rendimiento en predicción

La Tabla 6.3 muestra los resultados obtenidos para los 18 conjuntos de datos utilizados como referencia. La primera columna muestra la capacidad predictiva en test obtenida con los datos originales (20531 variables) y la segunda columna ('FS') muestra la capacidad predictiva en test cuando se aplica la selección de características, pero no se aplica el aumento de datos (DA). El siguiente grupo de columnas muestra los resultados obtenidos cuando se aplica DA a los conjuntos de datos después de la selección de características, mostrando la capacidad predictiva en test y el modelo con el que se obtiene, además del porcentaje de aumento realizado en los datos de entrenamiento. Los resultados que se muestran para cada conjunto de datos, corresponden al modelo generativo para el que se evalúa la mayor capacidad predictiva en el conjunto de validación. La última columna ($\widehat{RD}$) muestra el valor máximo de diferencia relativa sobre los resultados obtenidos anteriormente. Si el modelo incluye la terminación "M" se refiere a la modificación denominada 'Multiclass', mientras que "BM" se refiere a la modificación denominada 'Balanced Multiclass'.

Los resultados de la Tabla 6.3 indican que se alcanza una mejora media de 3.15 de $\widehat{RD}$ en la capacidad de predicción en test con el uso de técnicas de DA en comparación con el obtenido con el conjunto de datos sin aumento. Cabe destacar que esta mejora es con respecto a los resultados obtenidos tras realizar el proceso de selección de características, proceso con el cual se obtienen una mejora sustancial con respecto a los datos originales. Utilizando el proceso FS y las técnicas de DA, se logra una mejora de aproximadamente un 11 % con respecto a los conjuntos de datos de referencia originales.

UNIVERSIDAD DE MÁLAGA

Tabla 6.3: Capacidad de generalización obtenida con el conjunto original, tras haber realizado la selección de características y tras realizar DA, el modelo seleccionado y el porcentaje de muestras generadas, y el valor de $\widehat{RD}$ obtenido de acuerdo con esta selección.

			Aumento de Datos			
Datos	Original	FS	Acc.	Modelo	Porc.	$\widehat{RD}$
gbm	0.6353	0.7794	0.8185	CGAN	200	5.02
coad	0.7015	0.7421	0.7529	CGAN	50	1.45
ucec	0.7250	0.7955	0.8405	NOISE	200	5.66
lgg	0.6802	0.8313	0.8413	CGAN_M	50	1.20
ov	0.5192	0.7370	0.7921	ModCGAN_M	200	7.47
luad	0.6071	0.6368	0.6694	CGAN_M	100	5.13
lihc	0.6248	0.6690	0.6918	NOISE	200	3.40
blca	0.5566	0.7412	0.7636	CGAN_M	50	3.03
stad	0.6093	0.6623	0.6708	SMOTE	None	1.28
skcm	0.5213	0.6872	0.7377	ModCGAN_BM	200	7.34
prad	0.8918	0.9772	0.9810	SMOTE	None	0.39
lusc	0.5485	0.5624	0.5857	CGAN_BM	200	4.15
hnsc	0.5919	0.7000	0.7388	ModCGAN_BM	200	5.54
thca	0.9106	0.9640	0.9655	NOISE	200	0.16
kirc	0.7239	0.7836	0.8041	ModCGAN_BM	100	2.62
stes	0.5747	0.5895	0.6033	CGAN_M	50	2.35
kipan	0.7402	0.8265	0.8278	ModCGAN	200	0.17
brca	0.8072	0.8531	0.8567	ModCGAN	100	0.42
promedio	0.6667	0.7521	0.7745			3.15

La Figura 6.1 muestra la relación entre la capacidad predictiva obtenida para los conjuntos de datos procesados con FS y DA, y la obtenida para los conjuntos de datos de referencia originales. Los resultados muestran que la mejora en la capacidad predictiva obtenida de los conjuntos DA con respecto a los conjuntos FS y los resultados originales, tiende a ser mayor para aquellos conjuntos cuya capacidad de predicción para los datos originales es menor. También se puede observar que existen unos pocos conjuntos para los que apenas hay una mejora en la capacidad predictiva de los conjuntos procesados, en comparación con el original, lo que nos indica que el efecto del procesamiento de datos depende del conjunto de datos analizados.

La Tabla 6.4 muestra con más detalle los resultados obtenidos al realizar DA. Esta tabla muestra una comparación entre la diferencia relativa RD calculada con modelos basados en CGAN (incluidos ModCGAN y todas las variantes), la técnica SMOTE y la adición de ruido. La fuente en negrita indica el mejor valor para cada conjunto de datos. La columna $\widehat{RD}$ muestra el mismo valor que la columna homónima que se muestra en la Tabla 6.3, con los mejores resultados de RD cuya mayor capacidad de predicción se evalúa en el conjunto de validación. Las columnas denominadas "Potencial" muestran

UNIVERSIDAD DE MÁLAGA

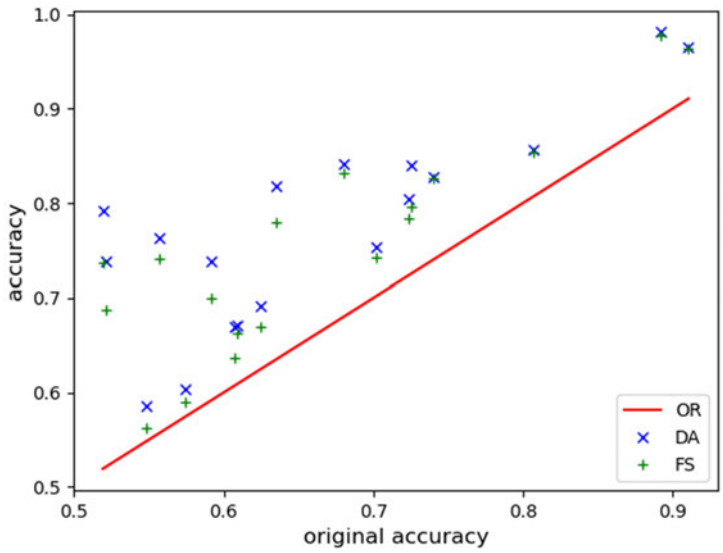

Figura 6.1: Relación entre la capacidad predictiva obtenida con los conjuntos procesados y la obtenida con el conjunto de referencia original. Las cruces representan la selección de características y las aspas la incorporación del DA. La línea continua es la función de identidad.

los modelos seleccionados y el valor de $\widehat{RD}$ alcanzado si se seleccionan los mejores resultados de test posibles, independientemente de la validación. Obviamente, realizando un proceso de validación cruzada, la selección directa por mayor valor de test no es honesto, sin embargo, nos da una idea del potencial de los modelos de DA, además de mostrar que existen diferencias respecto a escoger de validación (el conjunto de validación se utiliza para realizar una parada temprana en la red neuronal, lo que introduce cierto sesgo a los resultados). Esta comparación nos muestra que, si bien es cierto que hay diversos modelos que cambian, especialmente el porcentaje de aumento, los resultados difieren de una forma moderada (3.15 frente a 3.40).

Los resultados mostrados en la Tabla 6.3 y la Tabla 6.4 en referencia al análisis de la selección de los métodos de DA con mejor rendimiento para cada conjunto de datos, indican que 13 de 18 conjuntos (14 si se tiene en cuenta el potencial en test) logran mejores resultados de rendimiento con modelos basados en CGAN, obteniendo también los mejores resultados promedio con un valor de RD de 2.79. La adición de ruido es el mejor método para 3 conjuntos y SMOTE para 2, sin embargo, ambos métodos obtienen valores de RD negativos. Además, 12 de 18 métodos generan un porcentaje de muestras mayor o igual a 100, por lo que los modelos al menos duplican el número de muestras presentes en el conjunto de entrenamiento. Si se tiene en cuenta que la aplicación de SMOTE es aproximadamente similar a aplicar un porcentaje de aumento del 25 %, el promedio del porcentaje de aumento realizado por

UNIVERSIDAD DE MÁLAGA

Tabla 6.4: RD obtenido con cada modelo de DA seleccionado de acuerdo con el mejor resultado de validación para cada conjunto de datos, el mejor valor de $\widehat{RD}$ obtenido de acuerdo con esta selección, y el modelo y $\widehat{RD}$ obtenidos si se selecciona por el mejor valor de test.

	RD por Modelo				Potencial	
Conjunto	CGAN	SMOTE	NOISE	$\widehat{RD}$	Modelo	$\widehat{RD}$
gbm	**5.02**	-0.45	-1.21	5.02	$CGAN_{200}$	5.02
coad	**1.45**	-5.82	-0.50	1.45	$CGAN_{200}$	3.72
ucec	1.98	1.29	**5.66**	5.66	$NOISE_{200}$	5.66
lgg	**1.20**	-1.03	0.70	1.20	$CGAN_M_{50}$	1.20
ov	**7.47**	2.62	-6.56	7.47	$ModCGAN_M_{200}$	7.47
luad	**5.13**	0.35	-1.48	5.13	$CGAN_M_{200}$	5.57
lihc	1.19	1.64	**3.40**	3.40	$NOISE_{200}$	3.40
blca	**3.03**	0.24	-0.76	3.03	$CGAN_M_{25}$	3.40
stad	0.79	**1.28**	-0.91	1.28	SMOTE	1.28
skcm	**7.34**	-0.37	-2.94	7.34	$ModCGAN_BM_{200}$	7.34
prad	0.31	**0.39**	0.04	0.39	SMOTE	0.39
lusc	**4.15**	0.15	-4.98	4.15	$CGAN_BM_{100}$	4.80
hnsc	**5.54**	1.50	-5.90	5.54	$ModCGAN_BM_{200}$	5.54
thca	0.05	-0.03	**0.16**	0.16	$ModCGAN_BM_{25}$	0.34
kirc	**2.62**	0.01	2.04	2.62	$ModCGAN_BM_{100}$	2.62
stes	**2.35**	-0.43	0.03	2.35	$CGAN_M_{25}$	2.85
kipan	**0.17**	-2.85	-1.31	0.17	$ModCGAN_{200}$	0.18
brca	**0.42**	-0.90	0.09	0.42	$ModCGAN_{50}$	0.52
promedio	**2.79**	-0.13	-0.80	3.15		3.40

los modelos seleccionados es del 130 %.

Para analizar más a fondo la influencia del tamaño de los conjuntos de datos en la capacidad predictiva obtenida, se muestra en la Figura 6.2 la diferencia relativa RD en test obtenida con cada técnica de DA (valores de las primeras columnas de la Tabla 6.4) *versus* el número de instancias en una escala logarítmica en el eje x. Se ajusta un modelo de regresión lineal a los resultados de cada método de DA, obteniendo para los modelos basados en CGAN la ecuación $y = -1.29x + 10.62$, con un coeficiente de correlación de -0.29, lo que indica una correlación negativa moderada entre el aumento del número de instancias y la ganancia de capacidad predictiva obtenida. Los resultados son similares con el modelo de regresión lineal ajustado a los valores de inyección de ruido, una ecuación $y = -0.81x + 4.09$ y un coeficiente de correlación de -0.14. Los resultados obtenidos con SMOTE muestran que se mantiene la relación entre el número de instancias y la ganancia de capacidad de predicción, con un coeficiente de correlación de 0.04. Estos resultados son similares a los obtenidos con los experimentos realizados en el capítulo anterior. La Figura 6.2 también sirve para comparar visualmente los resultados obtenidos con los modelos de DA estudiados.

UNIVERSIDAD DE MÁLAGA

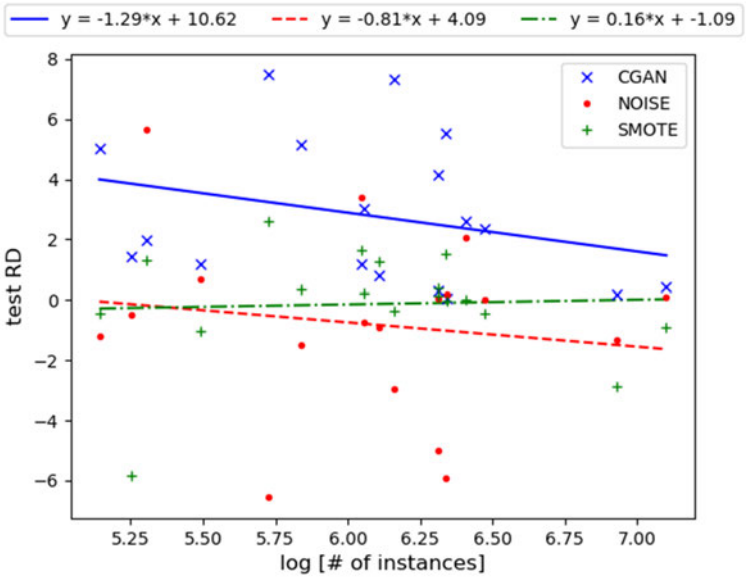

Figura 6.2: Diferencia relativa (RD) en capacidad de predicción frente al logaritmo del número de instancias. Las líneas son una regresión lineal ajustada a los datos. Las aspas y la línea continua representan los resultados obtenidos con CGAN, los puntos y la línea discontinua los obtenidos con NOISE, y las cruces y la línea punteada los obtenidos con SMOTE.

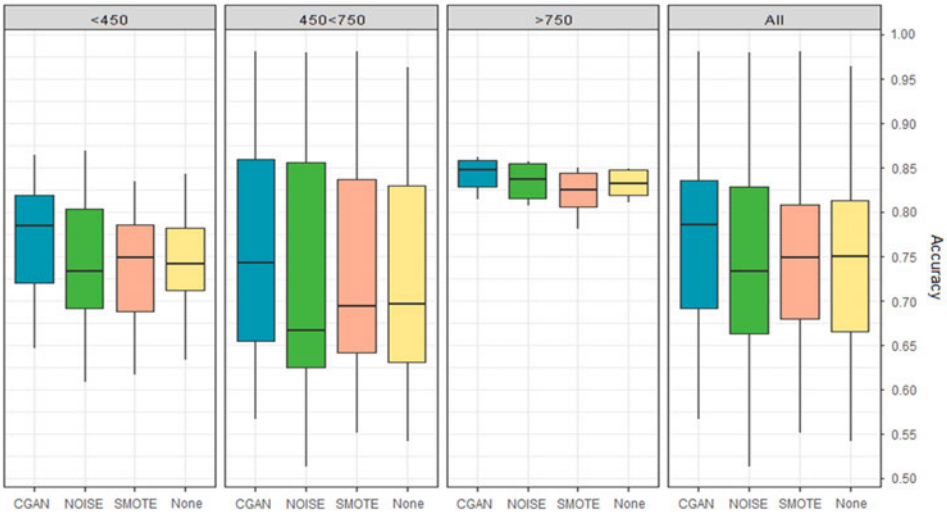

Figura 6.3: Comparación de la distribución de capacidad predictiva en test obtenida con diferentes modelos de DA agrupados según el número de instancias en el conjunto de datos original.

UNIVERSIDAD DE MÁLAGA

Teniendo en cuenta los resultados anteriores, la Figura 6.3 presenta un análisis de la simetría de la distribución de los resultados de las pruebas obtenidas por las diferentes técnicas de DA. Esta figura muestra los resultados agrupados según el número de muestras presentes en el conjunto, teniendo cuatro grupos: menos de 450 muestras; entre 450 y 750 muestras; más de 750 muestras; y todos los conjuntos. Cabe mencionar que los resultados se han obtenido con los resultados seleccionados en la Tabla 6.4 y el método mencionado como *"None"* se refiere a los resultados obtenidos con los conjuntos FS sin aumento de datos. La Figura 6.3 muestra una mayor dispersión de los resultados para los conjuntos de datos que presentan entre 450 y 750 muestras. La menor dispersión de resultados se produce para los conjuntos de más de 750 muestras, si bien comprende únicamente dos conjuntos de datos (*kipan* y *brca*). En cuanto a los métodos de DA, la generación de muestras con los modelos basados en CGAN obtiene mejores resultados de capacidad de predicción y con una dispersión similar, que si se utilizan los conjuntos sin aumento de datos. El método de adición de ruido es el que presenta una mayor dispersión para todos los agrupamientos realizados.

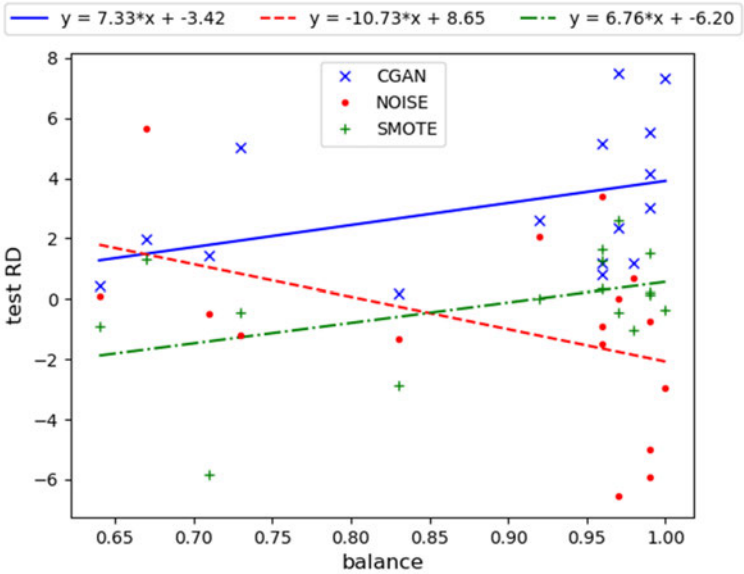

Figura 6.4: Diferencia relativa (RD) en capacidad predictiva frente al balanceo de los datos. Las líneas son una regresión lineal ajustada a los datos. Las aspas y la línea continua representan los resultados obtenidos con CGAN, los puntos y la línea discontinua los obtenidos con NOISE, y las cruces y la línea punteada los obtenidos con SMOTE.

Con el objetivo de realizar un análisis similar, pero teniendo en cuenta el balance de los datos en lugar del número de instancias, la Figura 6.4 mues-

UNIVERSIDAD DE MÁLAGA

tra la diferencia relativa obtenida con cada técnica DA *versus* el balanceo de clases presente en los datos. Esta figura no muestra los resultados para los conjuntos de datos *prad* y *thca* debido a su bajo valor de balanceo con respecto al resto (0.13 y 0.22 respectivamente), siendo atípicos e impidiendo una buena visualización de los resultados. Al igual que en el análisis del número de instancias, se ajusta un modelo de regresión lineal a los resultados de cada método de DA. Para los modelos basados en CGAN, la ecuación obtenida es $y = 7.33x - 3.42$, con un coeficiente de correlación de 0.4, lo que indica una correlación positiva entre el balanceo y la ganancia de capacidad de predicción. El modelo de regresión lineal ajustado a los resultados SMOTE es similar, $y = 6.76x - 6.20$ y coeficiente de correlación de 0.44. Por el contrario, los resultados con la adición del método de ruido indican una correlación negativa entre el balanceo y la ganancia de capacidad de predicción, con un coeficiente de correlación de -0.43.

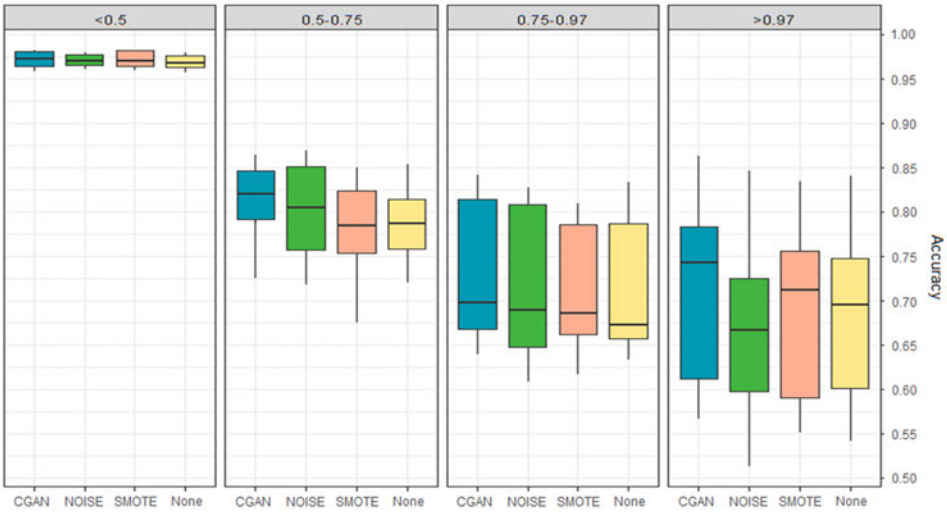

Figura 6.5: Comparación de la distribución de capacidad predictiva en test obtenida con diferentes modelos de DA agrupados según el balanceo de clases presente en el conjunto original.

La figura 6.5 presenta un análisis de la simetría de la distribución de los resultados de test obtenidos por las diferentes técnicas de DA, siendo el método mencionado como *"None"* los resultados obtenidos con los conjuntos habiendo aplicado selección de características, pero sin DA. Esta figura muestra los resultados agrupados según el balanceo de las clases en el conjunto, teniendo cuatro grupos: balanceo menor a 0.5; entre 0.5 y 0.75; entre 0.75 y 0.97; y superior a 0.97. En este punto recordamos que un balanceo de clases cercano a 1 indica que las clases del conjunto están bien equilibradas, y 0 indica totalmente desequilibrado. La Figura 6.5 muestra una distribución más pequeña de los resultados para los conjuntos de datos que presentan un

UNIVERSIDAD DE MÁLAGA

mayor desequilibrio (< 0.5 / $0.5 < 0.75$). En cuanto a los métodos de DA, la generación de muestras con modelos basados en CGAN también obtiene mejores resultados de capacidad predictiva en este caso, pero con una dispersión similar al resto de modelos. También se observa como el valor promedio de capacidad predictiva baja de forma general con respecto al aumento del balanceo, excepto el grupo con balanceo superior a 0.97. Esto podría ser debido a un mal funcionamiento de los clasificadores, que aplicados a conjuntos más desbalanceados pueden realizar una clasificación trivial mediante la predicción de las muestras como pertenecientes a la clase mayoritaria, aumentando de forma sencilla la capacidad predictiva. Cuando los datos se encuentran más balanceados, esta estrategia otorga menos beneficio y un valor de porcentaje de aciertos menor.

Tabla 6.5: Modelo seleccionado como mejor opción y sus resultados para cuatro casos: considerando todos los conjuntos de datos juntos o seleccionando el modelo para tres grupos de problemas según el número de instancias.

Datos	Método	$\overline{\#inst}$	Acc Val	RD	Acc Test	RD
Todos	ModCGAN_M_{25}	497	0.85	0.92	0.75	0.23
Sets 1-8	$CGAN_{25}$	288	0.87	0.62	0.75	0.81
Sets 9-16	ModCGAN_BM_{100}	551	0.84	1.90	0.75	0.96
Sets 17-18	$ModCGAN_{50}$	1116	0.88	0.47	0.84	0.16

Si solo se debe aplicar un método a todo el conjunto de datos, nuestros resultados indican que la mejor opción es el modelo ModCGAN con el esquema "Multiclass" y un 25 % de DA. Sin embargo, el análisis de los resultados mostrados hasta el momento lleva a la conclusión de que los conjuntos se comportan de manera diferente según las características que presentan. Si el conjunto de problemas se divide en tres grupos según el número de instancias disponibles, se pueden obtener mejores resultados. En la Tabla 6.5 se indica el número promedio de instancias para todo el conjunto de problemas, la capacidad predictiva obtenida en validación y los resultados de RD para el método seleccionado (siendo estos valores los más altos según la agrupación realizada), y la capacidad predictiva y RD obtenidos con el conjunto de test. En este caso, destaca el modelo CGAN estándar con un 25 % DA, con un rendimiento mayor para problemas con un número medio de instancias de 288, lo que lleva a un RD del 0.81 %; y el modelo ModCGAN con el esquema "Balance Multiclass" y 100 % de DA, siendo la mejor opción para un número medio de instancias (media igual a 551), con un RD de 0.96 %.

La Tabla 6.6 muestra que los resultados mejoran si el conjunto de problemas se divide en cuatro grupos según el equilibrio de las clases. En este caso, la adición de ruido con 200 % DA funciona mejor para conjuntos que presenten un equilibrio inferior a 0.5, lo que lleva a un valor de RD igual a 0.20 %; y un modelo ModCGAN con 25 % DA funciona mejor para conjun-

UNIVERSIDAD DE MÁLAGA

Tabla 6.6: Modelo seleccionado como mejor opción y sus resultados para cinco casos: considerando todos los conjuntos de datos juntos o seleccionando el modelo para cuatro grupos de problemas según el balanceo de clases.

Datos	Método	$\overline{\#bal}$	Acc Val	RD	Acc Test	RD
Todos	ModCGAN_M_{25}	0.81	0.85	0.92	0.75	0.23
< 0.5	$NOISE_{200}$	0.18	0.99	0.12	0.97	0.20
$0.5 - 0.75$	$CGAN_{200}$	0.69	0.92	0.94	0.80	1.12
$0.75 - 0.97$	$ModCGAN_{25}$	0.93	0.81	0.66	0.72	0.42
> 0.97	$ModCGAN_{50}$	0.98	0.81	1.96	0.70	1.25

tos cuyo equilibrio se encuentre entre 0.75 y 0.97, lo que lleva a un valor de RD igual a 0.42 %. Los mejores resultados de RD se obtienen con CGAN estándar con 200 % DA, con un RD de 1.12 para conjuntos con un equilibrio entre 0.5 y 0.75; y con ModCGAN estándar con 25 % DA, que lleva a un RD de 1.25 para conjuntos con un balanceo superior a 0.97.

Tabla 6.7: Valor de AUC obtenido con el conjunto de referencia (FS) y con cada modelo de DA, el mejor AUC obtenido de acuerdo con esta selección, y el valor $\widehat{RD}$ correspondiente.

Conjunto	FS	CGAN	ModCGAN	SMOTE	NOISE	Best	$\widehat{RD}$
gbm	0.6532	**0.6703**	0.6679	0.6138	0.6623	0.6703	2.62
coad	0.7212	0.7649	**0.7666**	0.6933	0.6836	0.7666	6.30
ucec	0.7432	0.7831	**0.7850**	0.7621	0.7751	0.7850	5.62
lgg	0.9136	**0.9312**	0.9263	0.9095	0.9053	0.9312	1.93
ov	0.7875	0.8063	**0.8134**	0.7765	0.7887	0.8134	3.29
luad	0.6703	**0.7124**	0.7120	0.6512	0.6742	0.7124	6.28
lihc	0.6716	0.7040	**0.7066**	0.6828	0.6763	0.7066	5.22
blca	0.8209	**0.8289**	0.8287	0.8211	0.8268	0.8289	0.98
stad	0.6718	0.6798	0.6827	0.6497	**0.6952**	0.6952	3.48
skcm	0.7519	0.7584	**0.7677**	0.7506	0.7624	0.7677	2.09
prad	0.5962	0.5941	**0.5999**	0.5946	0.5930	0.5999	0.62
lusc	0.5669	**0.5953**	0.5823	0.5453	0.5784	0.5953	5.00
hnsc	0.7710	0.7695	**0.7863**	0.7818	0.7722	0.7863	1.98
thca	0.8007	0.8230	**0.8256**	0.7920	0.8182	0.8256	3.10
kirc	0.8361	0.8445	0.8478	0.8294	**0.8481**	0.8481	1.44
stes	0.5986	**0.6080**	0.6020	0.5905	0.5996	0.6080	1.57
kipan	0.8515	0.8596	**0.8671**	0.8339	0.8354	0.8671	1.82
brca	0.7615	0.7719	**0.7741**	0.7398	0.7451	0.7741	1.66
promedio	0.7326	0.7503	**0.7523**	0.7232	0.7355	0.7545	3.06

Para ratificar el aumento del rendimiento en predicción mediante la aplicación de los diferentes modelos de DA, a parte de la capacidad predictiva, sesgada normalmente por la distribución de las clases, se calculó el valor de AUC. La Tabla 6.7 muestra los valores de AUC obtenidos para los 18 conjuntos de datos utilizados como referencia. La primera columna (FS) muestra el

UNIVERSIDAD DE MÁLAGA

AUC obtenido cuando se aplica únicamente la selección de características, y sirve como referencia. El siguiente grupo de columnas muestra los resultados obtenidos cuando se aplica DA con el modelo CGAN (incluidas las modificaciones de balanceo, 'M' y 'BM'), con el modelo adaptativo ModCGAN (incluidas variantes), la técnica SMOTE y el método de adición de ruido. La fuente en negrita indica el mejor valor para cada conjunto de datos. Las últimas columnas representan el mejor valor de AUC alcanzado y la diferencia relativa $\widehat{RD}$ sobre los resultados obtenidos anteriormente.

Estos resultados presentan cierto paralelismo con la Tabla 6.4, ratificando el aumento del rendimiento gracias al DA. Los mejores valores se alcanzan con los modelos generativos profundos (CGAN y ModCGAN) para 16 de los 18 conjuntos de datos. Para los 2 conjuntos restantes, el mayor valor se obtiene con la adición de ruido. Esto se ve reflejado en el valor promedio, alcanzando mediante la aplicación del modelo adaptativo un valor de 0.7523 en AUC, cercano al AUC obtenido si se seleccionan los mejores valores (0.7545). El valor de RD alcanzado es igual a 3.06, similar al obtenido para la capacidad predictiva (3.15). Para visualizar gráficamente el efecto del DA, se muestran en la Figura 6.6 las curvas ROC (*Receiver Operating Characteristic*) obtenidas para una selección de los conjuntos estudiados. Concretamente, para cada conjunto se muestra la curva ROC obtenida con el conjunto de referencia comparada con la obtenida mediante la aplicación de DA, siendo el método aplicado aquel que obtiene un mayor valor basándonos en los resultados de la Tabla 6.7.

Tabla 6.8: Modelo seleccionado como mejor opción y sus resultados considerando diferentes agrupaciones de los problemas: todos los conjuntos, según el número de instancias, y según el balanceo de clases.

Datos	Método	AUC Val	RD	AUC Test	RD
Todos	ModCGAN_BM$_{200}$	0.83	7.05	0.74	1.57
Instancias					
Sets 1-8	ModCGAN_BM$_{200}$	0.86	6.16	0.77	3.36
Sets 9-16	CGAN$_{200}$	0.80	6.87	0.71	1.80
Sets 17-18	ModCGAN$_{100}$	0.81	5.41	0.82	1.50
Balanceo					
< 0.5	ModCGAN_BM$_{200}$	0.84	7.53	0.71	1.76
$0.5 - 0.75$	CGAN_M$_{50}$	0.86	9.10	0.74	3.30
$0.75 - 0.97$	ModCGAN_M$_{50}$	0.81	5.50	0.76	2.83
> 0.97	ModCGAN_BM$_{100}$	0.81	4.14	0.75	1.31

Si se debe aplicar un método según los resultados de AUC, la mejor opción es el modelo ModCGAN con el esquema "Balance Multiclass" y un 200 % de DA. Sin embargo, al igual que en el caso de la capacidad predictiva, se pueden obtener mejores resultados si la selección se realiza tras agrupar los conjuntos

UNIVERSIDAD DE MÁLAGA

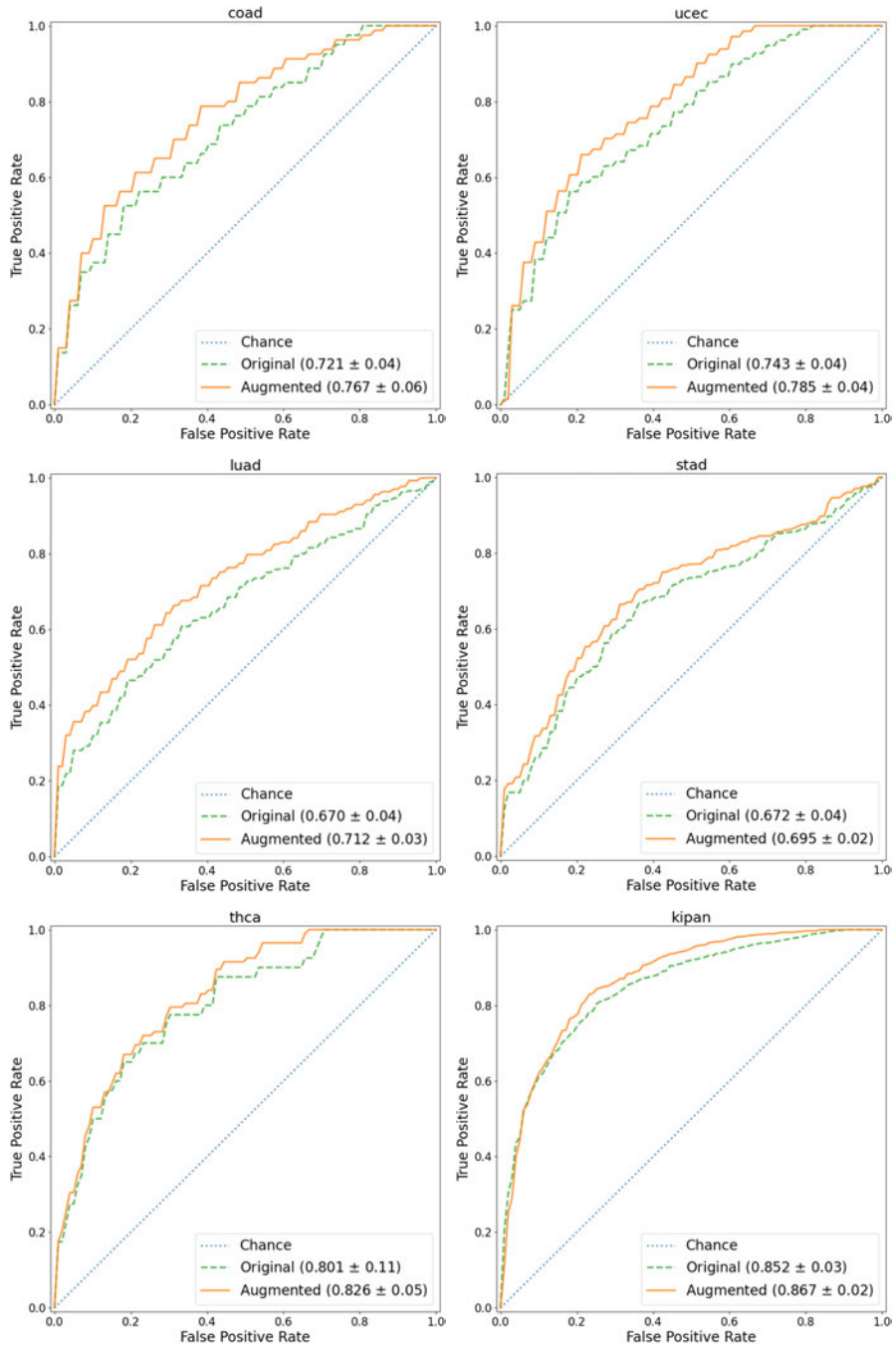

Figura 6.6: Curvas ROC obtenidas utilizando el conjunto de datos original y aplicando aumento de datos. La línea punteada representa el azar; la línea discontinua representa el enfoque con los datos originales; y la línea continua representa el aumento de datos.

UNIVERSIDAD DE MÁLAGA

según las características que presentan. La Tabla 6.8 muestra los resultados aplicando las mismas agrupaciones anteriores según número de instancias y balanceo de clases. Destaca el rendimiento del modelo ModCGAN_BM con 200 % DA con un número reducido de instancias (RD de 3.36) y mayor balanceo (RD de 1.76); y el modelo CGAN_M con 50 % DA, con un RD igual a 3.30 en conjuntos con un equilibrio entre 0.5 y 0.75.

6.3.2. Análisis del conjunto generado

Con el propósito de analizar cómo los métodos de DA utilizados en la experimentación replican la información presente en el conjunto de datos de expresión génica, se llevó a cabo un análisis de componentes principales (PCA). Mediante este método se puede visualizar la configuración de las muestras en un espacio bidimensional. Esto proporciona un medio para verificar la distribución de las muestras sintéticas generadas y compararla con la distribución de las muestras originales. La Figura 6.7 muestra la distribución de las muestras sintéticas creadas usando el método de adición de ruido con respecto a las muestras originales para los conjuntos de datos *ov*, *skcm*, *luad* y *thca*. Para no sobrecargar el gráfico, se presenta una selección de muestras generadas en similar número que las muestras originales. Debido al funcionamiento de este método, modificando las muestras originales mediante la adición de ruido gaussiano, se puede observar cómo las muestras sintéticas se mantienen alrededor de las muestras originales que modifican. Las muestras sintéticas presentan una distribución similar a la original, con zonas sin la presencia de muestras sintéticas debido al procedimiento de selección aleatoria de las muestras a modificar.

Los resultados de PCA después de agregar las muestras generadas con SMOTE se muestran en la Figura 6.8. Como ya ha sido explicado, SMOTE es una técnica de sobremuestreo, de forma que solo genera datos sintéticos para las clases minoritarias. En el caso del conjunto de datos *ov*, se han creado muestras de la clase 0 ("vivo") y para los otros tres conjuntos de datos se han creado muestras de la clase 1 ("muerto"). Las muestras sintéticas creadas por SMOTE se encuentran dentro de la distribución de las muestras originales, debido a su generación por interpolación. Esto implica una menor variabilidad en el espacio de características de las muestras.

La Figura 6.9 muestra la distribución de las muestras sintéticas creadas por los modelos basados en CGAN con respecto a las muestras originales. Estos resultados presentan algunas diferencias claras respecto a los obtenidos con el método de adición de ruido, debido al funcionamiento de los modelos GAN. Esencialmente, la mayoría de las muestras sintéticas están influenciadas por la densidad en la distribución que presentan los datos originales. Teniendo en cuenta que las muestras reales con las que se entrena el discriminador en los modelos GAN presentan unas características simila-

UNIVERSIDAD DE MÁLAGA

res (agrupadas en el núcleo de la distribución), el generador tiende a crear un número mayor de muestras con características semejantes a estas, siendo más fácil su objetivo de engañar al discriminador. Se puede observar para todos los conjuntos de datos como las muestras sintéticas para las diferentes clases se concentran donde hay la mayor cantidad de muestras reales de sus respectivas clases, siendo este el núcleo de la distribución. Sin embargo, una diferencia observable respecto a la generación mediante adición de ruido, es el margen que otorgan los modelos GAN al proceso de creación de muestras. Esto resulta en la creación de algunas muestras sintéticas que exceden los límites de las distribuciones reales. Estas muestras han sido generadas porque

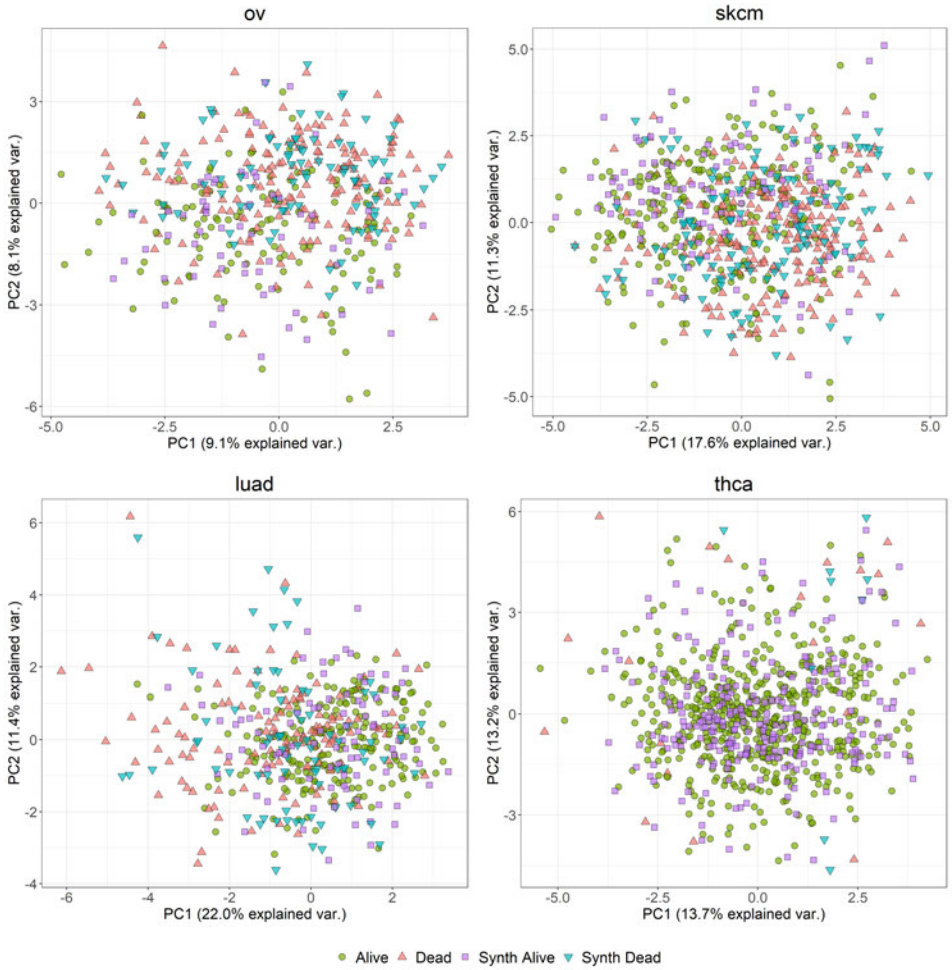

Figura 6.7: Gráfico de PCA con las muestras originales y las muestras generadas por el método de adición de ruido para los conjuntos *ov*, *skcm*, *luad* y *thca*. Códigos de color: círculo verde, "vivo"; triángulo rojo, "muerto"; cuadrado morado, "vivo" sintético; triángulo invertido azul, "muerto" sintético.

UNIVERSIDAD DE MÁLAGA

se encuentran ubicadas espacialmente donde los modelos generativos profundos creen que podría encontrarse una muestra real, expandiendo levemente la distribución real y pudiendo otorgar mayor variabilidad a los conjuntos de datos aumentados.

Para no depender únicamente de una inspección visual de los resultados de PCA para analizar las muestras generadas, se emplea la *Fréchet Inception Distance* (FID) (Heusel et al., 2017). La métrica FID está diseñada para medir la calidad de las imágenes generadas por los modelos GAN, pero puede ser aplicada para evaluar el rendimiento de estos modelos en cualquier aplicación de generación de datos. Concretamente, FID se presentó como una mejora sobre la métrica *Inception Score* existente (Salimans et al., 2016). En

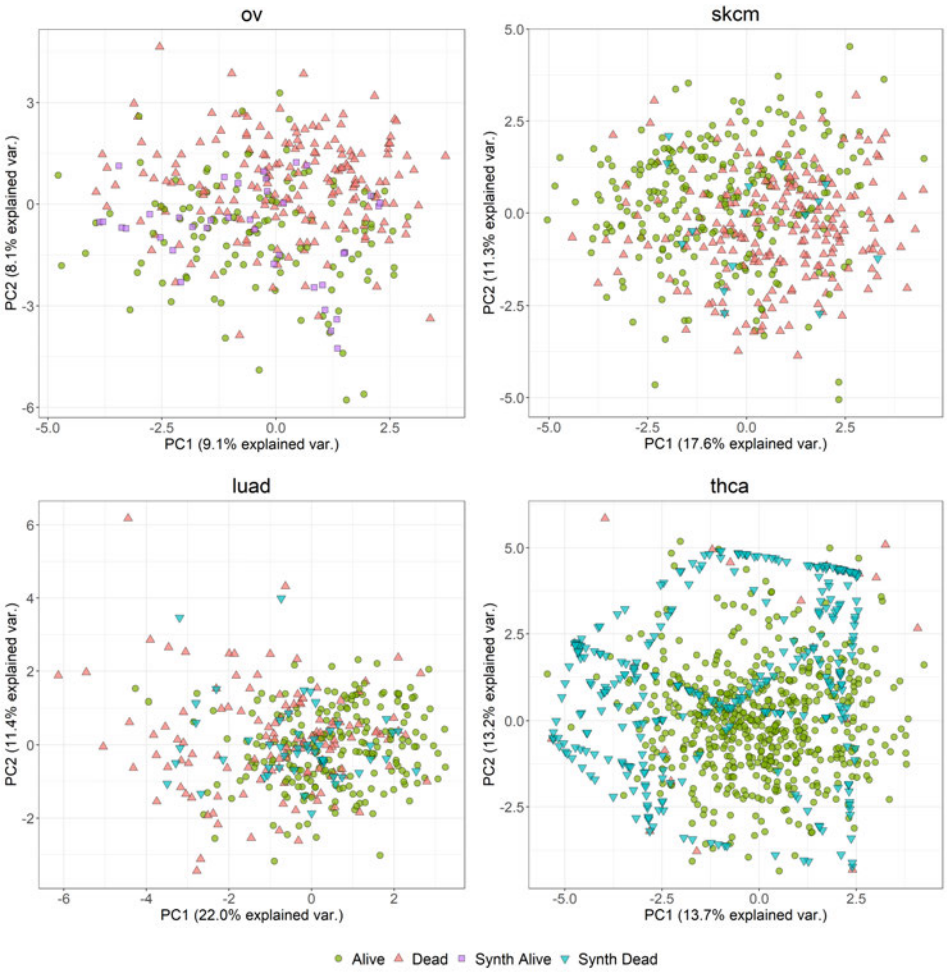

Figura 6.8: Gráfico de PCA con las muestras originales y las muestras generadas con SMOTE para los conjuntos de datos *ov*, *skcm*, *luad* y *thca*. Códigos de color: igual que la Figura 6.7.

UNIVERSIDAD DE MÁLAGA

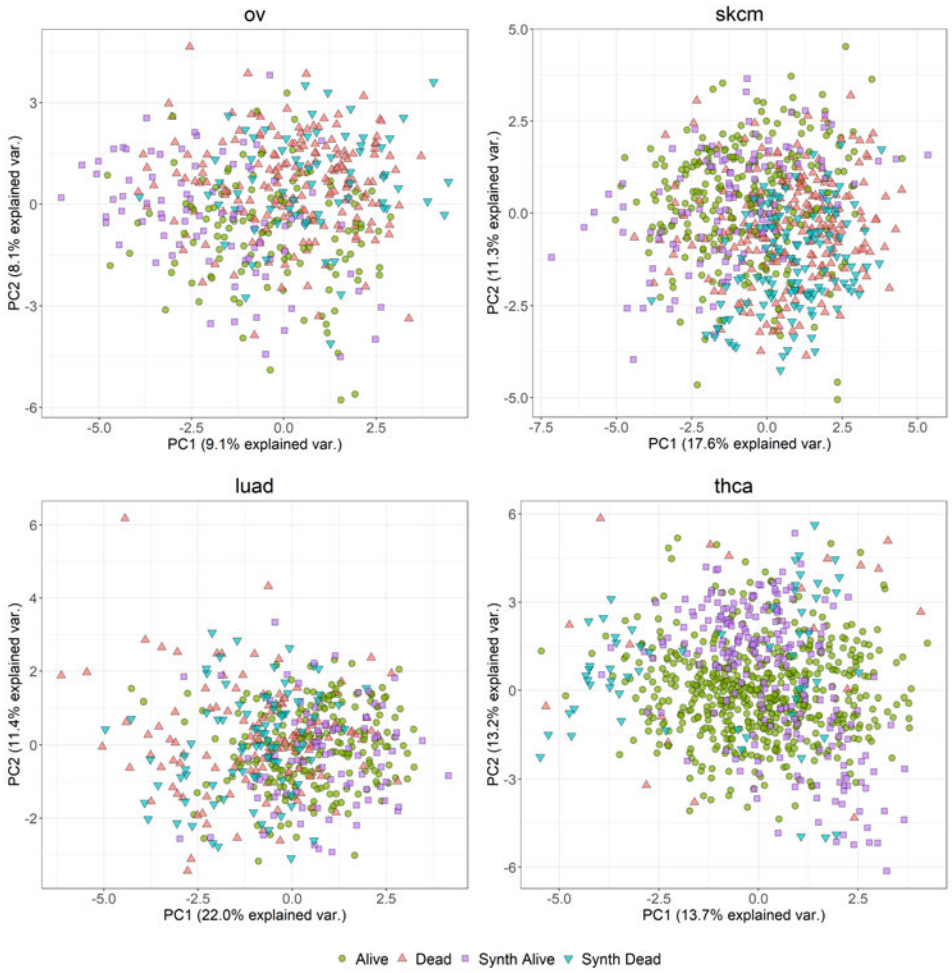

Figura 6.9: Gráfico de PCA con las muestras originales y las muestras generadas con los modelos basados en CGAN para los conjuntos de datos *ov*, *skcm*, *luad* y *thca*. Códigos de color: igual que la Figura 6.7.

lugar de necesitar de un modelo externo que realice la evaluación de calidad y diversidad, FID compara directamente la distribución de los datos generados con la distribución de los datos reales. La ecuación 6.1 muestra el cálculo de FID para la distribución normal "multivariante" a partir de las magnitudes del vector real (μ_r) y el vector generado (μ_g) , la traza de la matriz (Tr) y la matriz de covarianza de los vectores (Σ_r, Σ_g).

$$\text{FID} = \|\mu_r - \mu_g\|^2 + \text{Tr}(\Sigma_r + \Sigma_g - 2(\Sigma_r\Sigma_g)^{1/2}) \tag{6.1}$$

Para ser más precisos al medir la calidad de las muestras sintéticas mediante esta métrica, se obtiene el valor FID medio ($\widehat{FID}$) de realizar la comparación de la distribución sintética de la clase 0 con la distribución real de

UNIVERSIDAD DE MÁLAGA

la clase 0, y de realizar la comparación de la distribución sintética de la clase 1 con la distribución real de la clase 1. La Tabla 6.9 muestra los valores $\widehat{FID}$ obtenidos en la comparación de las distribuciones reales de los conjuntos de entrenamiento y test, con el conjunto sintético generado con los diferentes métodos de DA utilizados, haciendo también una comparación entre CGAN y ModCGAN. Para estos modelos se elige la variante (estándar, "Multiclass" o "Balanced Multiclass") que obtiene el mejor resultado de validación.

Tabla 6.9: Fréchet Inception Distance ($\widehat{FID}$) promedio obtenido con las muestras generadas por cada modelo de DA con respecto a las distribuciones reales de Train y Test, para cada conjunto de datos. Los valores más bajos representan una mayor similitud.

	SMOTE		NOISE		CGAN		ModCGAN	
Conjunto	Train	Test	Train	Test	Train	Test	Train	Test
gbm	0.150	0.893	0.179	0.602	0.407	0.769	0.206	0.644
coad	0.076	0.469	0.072	0.334	0.451	0.771	0.147	0.377
ucec	0.287	1.699	0.299	1.232	1.305	1.790	0.741	1.440
lgg	1.015	1.615	0.351	1.486	2.082	2.686	1.359	2.224
ov	0.264	0.490	0.103	0.434	0.745	1.099	0.256	0.554
luad	0.040	0.133	0.028	0.101	0.283	0.314	0.046	0.103
lihc	0.030	0.046	0.018	0.040	0.089	0.124	0.021	0.046
blca	1.441	2.037	0.541	2.060	3.170	3.820	2.057	2.471
stad	0.002	0.005	0.003	0.007	0.112	0.099	0.042	0.039
skcm	0.175	0.266	0.041	0.134	0.277	0.429	0.107	0.196
prad	0.269	0.000	0.047	0.094	1.020	1.795	1.142	2.687
lusc	0.086	0.135	0.029	0.093	0.280	0.326	0.051	0.105
hnsc	0.028	0.039	0.011	0.028	0.111	0.146	0.015	0.033
thca	0.223	1.083	0.222	0.623	0.801	1.265	0.411	0.804
kirc	0.378	0.949	0.341	1.127	3.413	3.657	1.359	1.708
stes	0.171	0.390	0.143	0.394	1.440	1.640	0.576	0.771
kipan	0.212	0.794	0.319	0.971	3.448	3.624	1.818	2.096
brca	0.128	0.700	0.206	0.682	2.064	2.336	1.052	1.291
promedio	0.276	0.652	0.164	0.580	1.194	1.483	0.634	0.977

La Figura 6.10 muestra la relación entre los valores $\widehat{FID}$ obtenidos en su comparación con la distribución real del conjunto de test, y los resultados de la diferencia relativa (RD) en capacidad predictiva. Los valores de $\widehat{FID}$ medidos para los modelos basados en CGAN se seleccionan de acuerdo con el valor de capacidad predictiva en validación más alto obtenido para cada conjunto. Por ejemplo, el $\widehat{FID}$ seleccionado para el conjunto de datos *ov* es el que se obtiene al compararlo con los datos sintéticos generados por ModCGAN_M, ya que este modelo presenta una capacidad predictiva en validación superior al resto de los modelos basados en CGAN. Los resultados indican que los datos generados con los modelos basados en CGAN obtienen una menor similitud con los datos reales, pero un mayor aumento en la capacidad de predicción. Concretamente, los modelos CGAN obtienen

UNIVERSIDAD DE MÁLAGA

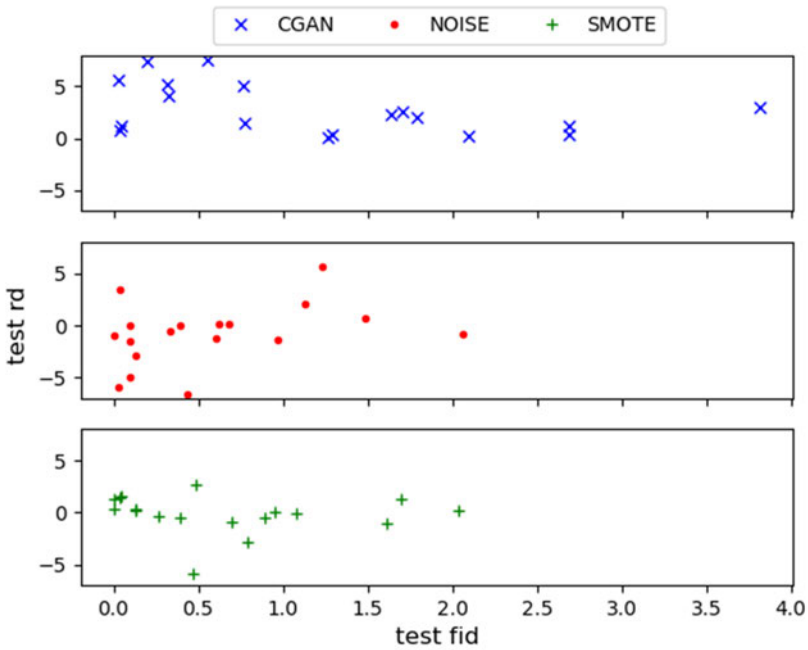

Figura 6.10: Fréchet Inception Distance ($\widehat{FID}$) *versus* la diferencia relativa en test. Las aspas (x) representan el modelo CGAN, los puntos representan el método NOISE y las cruces (+) representan SMOTE.

unos valores promedio de $\widehat{FID}$ de 1.194 y 1.483 con los conjuntos de entrenamiento y test, respectivamente. Las muestras generadas con los modelos modificados con el clasificador generativo (ModCGAN) obtienen unos valores significativamente más bajos que CGAN (0.634 y 0.977), pero son más altos que los obtenidos con Noise y SMOTE. Esto puede indicarnos que, gracias al clasificador generativo las muestras sintéticas generadas mantienen una mayor similitud con los datos reales. Con los modelos Noise y SMOTE se obtienen muestras de mayor calidad en comparación con las reales, pero con menor potencial de mejora en la predicción. Además, se observa que el método de adición de ruido obtiene los peores resultados de predicción con aquellas muestras que presentan un menor valor de $\widehat{FID}$ y, por tanto, una mayor similitud con las muestras reales de test. El análisis nos indica que una mayor variabilidad en el espacio de características de las muestras puede ser beneficioso para aumentar el rendimiento del modelo clasificador.

Otras medidas estadísticas que se tienen en cuenta para analizar las características de los datos sintéticos son la *Standard Distance Deviation* (SDD) y el desplazamiento del centro medio de la distribución (*Shift*). La métrica SDD es la distancia promedio de todas las muestras desde el centro medio y se usa para analizar la dispersión de una distribución. SDD proporciona una única medida como resumen de la distribución de muestras, de forma

UNIVERSIDAD DE MÁLAGA

similar a la desviación estándar, que mide la distribución de los valores de los datos alrededor del valor medio de la estadística. En este análisis, se utiliza previamente un proceso PCA para obtener un valor de SDD que abarque todas las variables presentes en los conjuntos de datos de referencia, además de reducir el tiempo de cálculo necesario. De esta forma, se realiza un PCA sobre los datos reales, obteniendo los vectores de puntuación de componentes y transformando todos los conjuntos de datos (reales y sintéticos). Con los conjuntos transformados, se utilizan los valores presentes en las muestras para la componente principal 1 y la componente principal 2, para realizar el cálculo de SDD. Aprovechando este proceso, el valor de desplazamiento del centro medio de la distribución se calcula mediante la distancia euclidiana entre el centro de la distribución del conjunto sintético y el centro de la distribución del conjunto original.

Tabla 6.10: Standard Distance Deviation (SDD) obtenida con los datos originales y con las muestras generadas por cada modelo de DA, y desplazamiento del centro medio de la distribución (Shift) obtenidos para las muestras generadas con respecto a las distribución real.

	Real	SMOTE		NOISE		CGAN		ModCGAN	
Conjunto	SDD	SDD	Shift	SDD	Shift	SDD	Shift	SDD	Shift
gbm	0.55	0.46	*0.076*	**0.60**	0.108	*0.44*	**0.292**	0.55	0.154
coad	0.43	0.34	*0.069*	0.30	0.104	**0.36**	**0.445**	*0.29*	0.172
ucec	0.56	*0.44*	*0.099*	0.53	0.131	**0.57**	**0.424**	0.54	0.316
lgg	0.56	0.56	*0.026*	*0.50*	0.092	0.52	**0.720**	**0.58**	0.105
ov	0.41	0.41	*0.049*	*0.40*	0.081	**0.52**	**0.158**	0.47	0.062
luad	0.34	*0.27*	0.077	0.33	0.066	**0.35**	**0.398**	0.35	*0.028*
lihc	0.32	*0.26*	*0.038*	0.33	0.047	**0.35**	**0.225**	0.27	0.107
blca	0.55	0.42	0.144	**0.61**	*0.075*	0.58	**0.381**	*0.40*	0.164
stad	0.24	0.21	*0.012*	**0.26**	0.028	*0.16*	0.108	0.21	**0.275**
skcm	0.41	*0.36*	0.104	0.40	*0.066*	**0.42**	**0.404**	0.38	0.154
prad	0.46	**0.39**	0.147	0.37	*0.028*	0.34	**0.210**	*0.24*	0.209
lusc	0.39	0.39	0.107	0.40	*0.087*	**0.41**	**0.172**	*0.37*	0.160
hnsc	0.29	*0.26*	0.031	**0.32**	*0.025*	0.28	**0.133**	0.28	0.066
thca	0.67	0.56	0.116	0.58	*0.069*	**0.69**	**0.655**	*0.52*	0.259
kirc	0.51	0.44	0.077	**0.59**	0.112	*0.41*	**0.660**	0.43	*0.056*
stes	0.50	*0.49*	0.061	0.57	*0.050*	**0.65**	**0.433**	0.60	0.083
kipan	0.51	0.47	*0.019*	**0.49**	0.070	*0.40*	**0.415**	0.45	0.114
brca	0.51	0.49	*0.060*	*0.49*	0.090	**0.58**	**0.716**	0.53	0.216
promedio	0.46	*0.40*	*0.07*	0.45	0.07	**0.45**	**0.39**	0.42	0.15

De manera similar a análisis realizado con FID, estas métricas se calculan para cada clase por separado en el conjunto de entrenamiento y, en este caso, se escoge el valor más alto como representación del modelo. La Tabla 6.10 muestra los resultados máximos de SDD obtenidos para los datos originales, y los valores de SDD y desplazamiento obtenidos para los datos sintéticos generados con cada método de DA. La fuente en negrita indica el

UNIVERSIDAD DE MÁLAGA

valor más alto logrado entre los resultados del conjunto de datos, y la fuente en cursiva el valor más bajo. Los resultados obtenidos para SDD nos indican que los modelos crean muestras con una dispersión similar a los datos originales (0.46), si bien el funcionamiento de SMOTE implica obtener el valor más bajo, pues la creación por interpolación reduce su dispersión, con un SDD igual a 0.40 y obteniendo el valor más bajo para 6 de 18 conjuntos. Los valores de desplazamiento indican que tanto SMOTE como la adición de ruido crean muestras que mantienen su centro medio cerca del presente en los datos originales, obteniendo con SMOTE 9 de 18 valores más bajos, y 7 de 18 para la inyección de ruido. Estos resultados junto al SDD, indican una superposición aproximada en el espacio de muestras. Las muestras generadas con los modelos CGAN presentan una dispersión similar a la original, y un desplazamiento mayor, con valores medios iguales a 0.45 para SDD y 0.39 para desplazamiento. Se obtienen valores de dispersión similares con el modelo ModCGAN, pero con un valor de desplazamiento más bajo. Un mayor valor de desplazamiento indica dos posibles circunstancias, la no superposición sobre la distribución original y la distribución balanceada de las muestras con otro foco de densidad. Si se analizan los resultados obtenidos con la inspección visual del PCA realizada anteriormente, la superposición del espacio de muestras sí parece darse de un modo similar al análisis con adición de ruido. El mayor desplazamiento del centro medio puede producirse entonces por la competición interna del modelo, creando un mayor número de muestras similares que engañan fácilmente al discriminador.

Por último, se utilizaron únicamente datos sintéticos para entrenar los modelos predictivos, con el propósito de analizar la calidad de las muestras generadas para obtener un rendimiento de predicción aceptable. En el experimento, los datos sintéticos se utilizan para entrenar y validar el clasificador utilizando un número de muestras igual al presente en los conjuntos originales. El conjunto de test utilizado para obtener la predicción es el mismo reservado para las pruebas en los experimentos iniciales y los resultados que se muestran son los obtenidos con los modelos que presentan el mayor valor de capacidad predictiva en el conjunto de validación.

La Tabla 6.11 muestra la capacidad de predicción en test obtenida cuando se utilizan los datos originales, los datos procesados con la selección de características, solo los datos sintéticos y cuando se utilizan conjuntos de datos aumentados (estos valores corresponden a los casos analizados mostrados en la Tabla 6.3). Para los datos sintéticos también se muestran los modelos con los que se han obtenido mejores resultados. La última columna $\widehat{RD}$ muestra el valor máximo de diferencia relativa sobre los resultados obtenidos anteriormente. La fuente en negrita indica los mejores valores de capacidad predictiva obtenida, mostrando claramente que en la mayoría de los casos (16 de 18) es beneficioso utilizar una técnica de DA para obtener un mejor desempeño en la predicción. Para 2 casos, se obtuvieron valores

UNIVERSIDAD DE MÁLAGA

Tabla 6.11: Capacidad predictiva usando los datos originales, procesados con selección de características, solo las muestras sintéticas generadas, y con aumento de datos, y el mejor valor de $\widehat{RD}$ obtenido acorde a estos resultados.

			Sintético			
Conjunto	Original	FS	Acc.	Modelo	DA	$\widehat{RD}$
gbm	0.6353	0.7794	0.7861	NOISE	**0.8185**	5.02
coad	0.7015	0.7421	0.7418	NOISE	**0.7697**	3.72
ucec	0.7250	0.7955	0.8093	NOISE	**0.8405**	5.66
lgg	0.6802	0.8313	0.8392	NOISE	**0.8413**	1.20
ov	0.5192	0.7370	0.7223	ModCGAN_M	**0.7921**	7.47
luad	0.6071	0.6368	0.6131	ModCGAN_M	**0.6722**	5.57
lihc	0.6248	0.6690	0.6756	ModCGAN_M	**0.6918**	3.40
blca	0.5566	0.7412	0.7169	NOISE	**0.7664**	3.40
stad	0.6093	0.6623	0.6476	NOISE	**0.6708**	1.28
skcm	0.5213	0.6872	0.7324	ModCGAN_BM	**0.7377**	7.34
prad	0.8918	0.9772	**0.9814**	NOISE	0.9810	0.43
lusc	0.5485	0.5624	0.5425	NOISE	**0.5894**	4.80
hnsc	0.5919	0.7000	0.7367	ModCGAN_BM	**0.7388**	5.54
thca	0.9106	0.9640	0.9633	NOISE	**0.9673**	0.34
kirc	0.7239	0.7836	0.7898	ModCGAN_BM	**0.8041**	2.62
stes	0.5747	0.5895	**0.6113**	NOISE	0.6063	3.71
kipan	0.7402	0.8265	0.8015	NOISE	**0.8279**	0.18
brca	0.8072	0.8531	0.8521	NOISE	**0.8576**	0.52
promedio	0.6667	0.7521	0.7535		**0.7763**	3.45

ligeramente más altos utilizando solo los conjuntos sintéticos creados. Cabe destacar que para 12 de los 18 conjuntos se ha seleccionado el método de adición de ruido, cuyo procedimiento a la hora de crear muestras las hace muy similares a las reales y obtienen un mayor valor de validación, por lo que son seleccionados. Los otros 6 modelos seleccionados son variaciones de balanceo del modelo ModCGAN (CGAN con asignación de etiquetas por clasificador generativo).

6.4. Conclusiones

En esta investigación se ha propuesto la aplicación de diferentes técnicas de DA, con el objetivo de mejorar el rendimiento en predicción del evento en intervalo de tiempo fijo, concretamente el estatus vital, en diferentes tipos de cáncer que se puede obtener a partir del análisis de conjuntos de datos con perfiles de expresión génica RNA-Seq. En concreto, se han estudiado métodos de DA tradicionales como la inyección de ruido y la técnica SMOTE, y métodos más novedosos como son los modelos CGAN y las variaciones mediante el modelo adaptativo diseñado y expuestos en el Capítulo 5, incluyendo modificaciones en el control del equilibrio de los datos generados.

UNIVERSIDAD DE MÁLAGA

Se realizó un extenso número de simulaciones experimentales analizando 18 tipos de cáncer para la predicción del estatus vital de los pacientes, presentando la mayoría de los conjuntos un tamaño reducido, con un número de instancias inferior a 1000.

Los resultados indican que el uso de técnicas de DA puede llevar a un aumento en la capacidad de predicción de aproximadamente un 3 %, evaluando todos los métodos probados y eligiendo el mejor según el error de validación. Esta mejora se ha logrado con respecto a los conjuntos tras realizar un proceso de selección de características, fundamental para incrementar el potencial de predicción en conjuntos de datos biomédicos (Peng et al., 2010). La mejora en la capacidad predictiva con respecto a los datos originales 'en bruto' es de aproximadamente un 11 %. Los resultados que se muestran en la Tabla 6.4 pueden considerarse los resultados más relevantes de este trabajo, mostrando que el uso de los modelos basados en CGAN mejoran los resultados de generalización que se pueden obtener con otras técnicas tradicionales como SMOTE y la adición de ruido. Con los métodos basados en CGAN, se obtiene un aumento en la capacidad predictiva del modelo clasificador (2.79 % RD), mientras que con SMOTE y la adición de ruido hay una disminución en el rendimiento medio. Estas conclusiones han sido corroboradas analizando el valor de AUC, obteniendo mediante el modelo adaptativo un aumento de aproximadamente un 3 %.

El análisis realizado respecto al número de instancias y el balanceo de los conjuntos, muestra una leve correlación negativa entre el aumento del número de instancias y la ganancia de capacidad predictiva para los modelos basados en GAN, y una correlación positiva entre el aumento del equilibrio de clases y la ganancia obtenida. Estas correlaciones pueden deberse a que los conjuntos con mayor número de instancias o más desequilibrados presentan un valor base de porcentaje de aciertos elevado, reduciendo el margen de mejora. Estos análisis muestran además una posible limitación en la aplicación del DA en entornos reales, la disminución del rendimiento si se selecciona solo un método. Esta selección debería hacerse dependiendo de las características de balanceo o número de instancias. Se demuestra que realizar divisiones según estas características permite una mejora del rendimiento alcanzado seleccionando un solo método, si bien no es tan elevada como sería deseable.

Respecto a los análisis gráficos y estadísticos de las muestras generadas por los métodos de DA, estos indican que los conjuntos sintéticos presentan una buena calidad, revelando también características de los mismos que explican el rendimiento alcanzado. Los resultados muestran que la técnica de adición de ruido genera muestras con un gran ajuste y similitud con las distribuciones originales del conjunto de entrenamiento, obteniendo el valor FID más bajo (0.164), y que SMOTE presenta un potencial para incrementar la variabilidad de muestras dentro de la clase minoritaria, pero con una gran restricción en la dispersión. De los métodos basados en GAN, el análi-

UNIVERSIDAD DE MÁLAGA

sis indica que agregan más variabilidad a los conjuntos, generando muestras que se ajustan a la distribución real pero también algunas que se posicionan fuera de estas distribuciones. Las muestras generadas con el modelo CGAN presentan una menor similitud con los conjuntos originales, alcanzando un valor de FID más elevado (1.19). Por otro lado, las muestras generadas con el modelo adaptativo ModCGAN, presentan valores inferiores, indicando una mayor similitud con las muestras reales. Los valores de FID son casi la mitad de los obtenidos con CGAN, lo que indica una mejora en la calidad de las muestras generadas por nuestro modelo desde un punto de vista estadístico.

Por último, el entrenamiento de los modelos predictivos solo con muestras sintéticas también indica que la calidad de las muestras generadas con los métodos de DA es alta, obteniendo una capacidad de predicción de 0.7535, ligeramente superior a la obtenida con los conjuntos de datos sin aumentar (0.7521). Estos resultados señalan que la adición de ruido implica la creación de muestras con mayor similitud y una distribución más ajustada a los conjuntos originales, alcanzando los mejores resultados en predicción con conjuntos puramente sintéticos, si bien su impacto con conjuntos aumentados es muy inferior. De estos análisis se puede concluir que no solo es necesario que las muestras sintéticas tengan calidad y presenten una buena similitud con los datos reales, sino que además deben brindar variabilidad a los conjuntos para incrementar la eficiencia de los modelos clasificadores. Esto podría permitir una mejor predicción de muestras de test que el modelo nunca ha visto, y que pueden no ser similares a las muestras de entrenamiento.

En conclusión, las técnicas de DA constituyen un enfoque adecuado para aumentar el rendimiento del análisis del estatus vital de un paciente en un intervalo de tiempo fijo a partir de perfiles de expresión génica RNA-Seq. Las técnicas de DA basadas en modelos CGAN son capaces de generar datos sintéticos de buena calidad que conducen en promedio a un aumento relativo de la capacidad predictiva y AUC del 3 %. Por estas razones, es interesante analizar la aplicación de los métodos de DA a otros conjuntos de datos de expresión génica y otros problemas del ámbito de la bioinformática, probando con diferentes procesos de experimentación y analizando otras medidas de rendimiento.

UNIVERSIDAD DE MÁLAGA

UNIVERSIDAD
DE MÁLAGA

Capítulo 7

Predicción Niemann-Pick C1

Resumen: La enfermedad de Niemann-Pick Clase 1 (NPC1) es una enfermedad de almacenamiento lisosomal (LSD) neurodegenerativa rara, para la cual los biomarcadores disponibles para su pronóstico clínico son actualmente limitados. Los conjuntos de datos metabolómicos de pacientes con NPC1 disponibles para realizar este tipo de análisis suelen presentar limitaciones en el número de muestras y un gran desequilibrio entre las clases. Para mejorar la capacidad predictiva e identificar nuevos biomarcadores de la enfermedad NPC1 en un conjunto de datos urinarios, se emplean en este capítulo técnicas de aumento de datos (DA) basadas en inteligencia computacional para crear muestras sintéticas, como son la adición de ruido, técnicas de sobremuestreo y GANs. Se experimenta con diferentes modelos de clasificación de aprendizaje automático y otras técnicas, como la validación cruzada de análisis discriminatorio, obteniendo resultados que muestran una mejora sustancial de la sensibilidad y la métrica F_1 mediante DA. Además, se emplean formas más convencionales de análisis de datos multivariados que permiten la detección de perfiles de metabolitos urinarios inusuales y la identificación de biomarcadores, así como demostrar la calidad de las muestras generadas mediante los modelos de DA.

7.1. Introducción al Problema

La enfermedad de Niemann-Pick tipo C (NPC, OMIM 257220) es una enfermedad hereditaria muy rara de almacenamiento lisosomal neurodegenerativa (LSD) causada por mutaciones en dos genes específicos, NPC1, cuya mutación representa un 95 % de los casos clínicos, y NPC2 (Vanier, 2010). La enfermedad afecta la capacidad de los pacientes para metabolizar las grasas (el colesterol y los lípidos) dentro de las células, las cuales no funcionan

UNIVERSIDAD DE MÁLAGA

adecuadamente y, a la larga, mueren. Desde una perspectiva clínica específica, la enfermedad de NPC implica el almacenamiento lisosómico alterado de esfingosina, un aminoalcohol formado por una cadena hidrocarbonada insaturada. Esto conduce a una pérdida de iones de calcio lisosómico, un proceso que viene acompañado por la acumulación de colesterol no esterificado y glucoesfingolípidos (Winkler et al., 2019; Platt et al., 2018), junto con niveles reducidos de calcio almacenado (Lloyd-Evans et al., 2008). El gen NPC1 codifica una proteína transmembranal que reside en la membrana limitante del lisosoma y se conoce como proteína NPC1 (Winkler et al., 2019). NPC2 codifica una proteína lisosomal soluble que se une al colesterol dentro del lumen del lisosoma, y se cree que transfiere el colesterol a la proteína NPC1. En Winkler et al. (2019) están disponibles los detalles completos sobre cómo estas proteínas pueden complementarse entre sí para permitir que el colesterol pase a través del glucocálix y la membrana del lisosoma.

Por lo general, la enfermedad de NPC se presenta más frecuentemente en niños. Los síntomas incluyen torpeza, ataxia, dificultades de aprendizaje, parálisis de la mirada vertical y disfagia, junto con cataplejía, epilepsia, hepatoesplenomegalia (aumento simultáneo del tamaño del hígado y del bazo) y, en casos graves, disfunción respiratoria. La enfermedad también se puede manifestar en la edad adulta, lo que puede estar asociado con la aparición de diversos trastornos psiquiátricos (Vanier, 2010). La enfermedad de NPC también implica neuroinflamación, apoptosis neuronal y estrés oxidativo dentro de su cascada patológica (Cougnoux et al., 2016).

Para el diagnóstico y seguimiento del pronóstico de este tipo de enfermedades lisosomales, los estudios metabolómicos son valiosos. Los sistemas de conjuntos de datos bioanalíticos pueden analizarse en condiciones preestablecidas determinadas por el diseño experimental, lo que permite también realizar análisis para estudiar la respuesta de los pacientes después de la exposición a estímulos, tratamientos o regímenes de ejercicio específicos. El carácter no invasivo de la metabolómica y la estrecha vinculación de este tipo de datos con el fenotipo, la convierten en una herramienta ideal para la salud farmacéutica y preventiva, entre otros. La metabolómica permite a los investigadores desarrollar una comprensión profunda de cómo el sistema explorado responde a tales estímulos. En los estudios clínicos, esto tiene aplicaciones potenciales para monitorear el diagnóstico y pronóstico del paciente, el manejo terapéutico e incluso el desarrollo de fármacos para las condiciones investigadas. Este último enfoque implica la identificación de actividades anormales en vías metabolómicas seleccionadas, de modo que cualquier enzima defectuosa pueda ser dirigida al fármaco. La metabolómica también es aplicable al descubrimiento de biomarcadores como soporte para la toma de decisiones. En metabolómica, los biomarcadores son moléculas pequeñas, conocidas como metabolitos. A modo de ejemplo, los metabolitos seleccionados y sus concentraciones se pueden utilizar para determinar el

UNIVERSIDAD DE MÁLAGA

estado de diferentes grupos de muestras en función de su detección en muestras del grupo de control, o en las recogidas de pacientes con una enfermedad específica. La espectroscopia de resonancia magnética nuclear (NMR) de protones (^{1}H) de alta resolución es una herramienta analítica útil para estudios de perfiles metabólicos en vista de su capacidad analítica multicomponente y alta sensibilidad (Grootveld y Silwood, 2005). Las muestras de orina ^{1}H NMR analizadas en la investigación, contienen metabolitos informativos que se pueden analizar fácilmente con el fin de descubrir nuevos biomarcadores.

Los estudios metabolómicos se realizan típicamente mediante técnicas analíticas de componentes múltiples especializadas y altamente reproducibles. El uso de técnicas bioanalíticas unidas a técnicas de inteligencia computacional y de estadística multivariante, está evolucionando y se está volviendo más popular (Marshall y Powers, 2017). Sin embargo, el rendimiento en predicción alcanzado se ve reducido normalmente debido a que los conjuntos de datos metabolómicos disponibles están limitados en el número de muestras. Además, debido a que generalmente se realizan estudios sobre enfermedades raras, los conjuntos presentan un desequilibrio entre las clases alto, ya que se tienen disponibles numerosas muestras de control respecto a muestras de pacientes enfermos, o en el caso de estudios farmacológicos, muestras de pacientes a las que se les ha administrado cierto fármaco. A este respecto, las enfermedades NPC causadas por ambas mutaciones afectan aproximadamente a una de cada 100000 personas (Geberhiwot et al., 2018) siendo considerada una enfermedad rara, por lo que su impacto mediático es menor que, por ejemplo, el cáncer. En general, los trastornos por almacenamiento lisosómico representan una serie de $\approx$ 41 enfermedades hereditarias genéticamente distintas y metabólicamente relacionadas. De hecho, Meikle et al. (1999) encontró que la prevalencia de estas enfermedades variaba sustancialmente. Estos valores se encuentran entre 1 por cada 57000 y 1 por cada 4.2 millones de personas para las enfermedades de Gaucher y la sialidosis respectivamente (Meikle et al., 1999). En tales casos, la recolección de las muestras y del consentimiento ético de los padres necesario para dicha recolección, son a menudo obstáculos muy difíciles de superar. Además, la obtención de una cantidad suficiente de muestras de biofluidos para NMR u otros análisis aumenta esta complejidad.

Por esta razón, el uso de técnicas de DA basadas en inteligencia computacional para crear más observaciones puede resultar muy beneficioso. Mediante el aumento del número de datos con la creación de nuevas muestras sintéticas, también se propone detectar patrones metabólicos inusuales en pacientes con enfermedad NPC1 mediante un análisis de los perfiles de ^{1}H NMR de sus muestras de orina, pudiendo realizar una comparación con los biomarcadores obtenidos utilizando únicamente el conjunto de datos original. Esto tiene a su vez un doble propósito, realizar una medida de la calidad de los datos generados e identificar nuevos biomarcadores gracias a estos

UNIVERSIDAD DE MÁLAGA

datos, o al cambio del equilibrio del conjunto. Para respaldar aún más estas investigaciones, y también con fines comparativos, se utilizaron métodos convencionales de análisis de datos multivariados. Estos incluyeron tanto el análisis de componentes principales (PCA) (no supervisado) como el análisis discriminatorio de mínimos cuadrados parciales (PLS-DA) (supervisado).

En la actualidad hay una clara falta de estudios metabolómicos globales centrados en investigaciones de enfermedades de almacenamiento lisosómico, y solo se pueden reportar al respecto una pequeña cantidad de estudios (Ruiz-Rodado et al., 2016; Probert et al., 2017; Percival et al., 2020, 2021). Estos estudios justifican el valor que ofrecen las técnicas de análisis de datos metabolómicos basadas en NMR, siendo la mayoría de las investigaciones sobre metabolómica enfocadas a la enfermedad de NPC1 (Ruiz-Rodado et al., 2014; Probert et al., 2017). De hecho, el análisis de ^{1}H NMR de alta resolución de muestras de orina ha buscado e identificado con éxito biomarcadores potenciales para el diagnóstico de la enfermedad NPC1. Estos incorporaron aminoácidos de cadena ramificada (BCAA) como son la valina, la leucina y la isoleucina, azúcares N-acetil, ácidos biliares y 3-aminoisobutírico (Ruiz-Rodado et al., 2014). Este último metabolito se origina a partir de perturbaciones metabólicas del catabolismo de los BCAA o de las vías de degradación de la timina. Los análisis disponibles en la literatura indican además que, junto con el lisosoma en sí, es importante enfocar el análisis en características clave que involucran al cerebro y el hígado, lo cual supone una observación totalmente congruente con los síntomas característicos de la enfermedad NPC1, incluidas las convulsiones y la hepatomegalia. Como referencia, los valores obtenidos por estos estudios de AUROC oscilan entre 0.81 y 0.91 para las biomoléculas más significativas, y la aplicación de una estrategia PLS-DA obtiene un valor significativo de 0.56 de bondad de predicción (Q^2) con una exactitud de 0.93 (Ruiz-Rodado et al., 2014).

7.2. Implementación de los modelos

Con el objetivo de comprobar el efecto del DA en el rendimiento predictivo sobre un conjunto metabolómico con muestras de pacientes para el estudio de la enfermedad Niemann-Pick Tipo C, y realizar una comparación entre diversos métodos de DA, se utilizan la adición de ruido gaussiano estándar y en un proceso de sobremuestreo, la técnica SMOTE, el modelo CGAN estándar y un par de estrategias combinatorias.

En primer lugar, se utilizan los métodos de adición de ruido gaussiano explicados en el Capítulo 4 sección 4.2. Estos seleccionan aleatoriamente muestras del conjunto de entrenamiento y modifican aleatoriamente un 25 % de las características, calculando el nuevo valor mediante la adición de ruido que surge de una distribución normal aleatoria, con una desviación de 1, pues los datos no se encuentran normalizados, y el valor de 0.2 utilizado

UNIVERSIDAD DE MÁLAGA

en otros experimentos es insuficiente para la creación de muestras que difieran significativamente de las originales. Se verifica que los nuevos valores no excedan los límites establecidos (máximo y mínimo) para su característica. En esta experimentación se introdujo un nuevo método de adición de ruido con un propósito de sobremuestreo. Este método, abreviado como 'Noise Bal', únicamente se diferencia del método de adición de ruido estándar en la selección aleatoria, pues Noise Bal solo selecciona sobre las muestras de entrenamiento de la clase minoritaria. En cuanto a la técnica SMOTE aplicada, se utiliza la versión estándar (Chawla et al., 2002), detallada en el Capítulo 2 sección 2.1.3. Para la implementación de esta técnica se utiliza el módulo *imbalanced-learn* (Lemaître et al., 2017) con los valores por defecto ($sampling_strategy$ = '$auto$'; $k_neighbors = 5$).

Los detalles de implementación y arquitectura del modelo CGAN utilizado para el proceso generativo de muestras sintéticas, son los mismos que los utilizados en el experimento de predicción de eventos en cáncer. Estos están presentes en la Tabla 6.1, mostrando una distinción entre la red generador y discriminador, y exponiendo para cada una el tipo de capa, tamaño de salida de cada una (número de neuronas presentes en cada capa), el uso de la normalización por lotes, la función de activación utilizada en la capa, los algoritmos de optimización utilizados, la parametrización de la función de activación y el software usado como backend. El único detalle que es diferente en la implementación del modelo CGAN utilizado para esta experimentación es la reducción del tamaño del vector latente de ruido que recibe la red generadora, pasando a tener un tamaño de 33, igual que el número de características presentes en el conjunto de datos. Las dos estrategias combinatorias diseñadas representan la unión de las muestras generadas utilizando el modelo CGAN y la técnica SMOTE, y la unión de las muestras generadas utilizando el modelo CGAN y el método Noise Bal. De esta forma se quiere observar si la inclusión de las muestras generadas por sobremuestreo a las muestras creadas por CGAN, es beneficioso para los modelos clasificadores.

Tabla 7.1: Número de muestras generadas y tamaño del conjunto de entrenamiento aumentado, según el porcentaje de DA aplicado. La aplicación de SMOTE siempre crea el mismo número de muestras.

	Porcentaje de DA							
Conjunto	25	50	100	200	500	1000	2000	SMOTE
Generado	9	18	36	72	180	360	720	20
Aumentado	45	54	72	108	216	396	756	56

Para esta investigación, los modelos generativos principales son probados con 7 niveles diferentes de porcentajes de aumento de datos (25 %, 50 %, 100 %, 200 %, 500 %, 1000 %, 2000 %). En este caso se han realizado experimentos con porcentajes tan elevados debido a que el número de muestras

UNIVERSIDAD DE MÁLAGA

disponibles era muy reducido. En la Tabla 7.1 se muestran el número de muestras generadas y presentes en el conjunto de entrenamiento aumentado, para cada porcentaje de aumento de datos que se aplica con los modelos generativos, incluyendo SMOTE que siempre genera el mismo número de muestras, todas pertenecientes a la clase minoritaria. De este modo, los conjuntos de entrenamiento tienen un tamaño medio de 36 muestras, generándose un máximo de 720 muestras sintéticas con un porcentaje de 2000 %, teniendo en cuenta que mediante el método Noise Bal se genera un número menor de muestras, todas de la clase minoritaria. La experimentación con estos porcentajes también influye en la selección de modelos clasificadores y generativos realizada. De los modelos generativos profundos solo se experimenta con la versión estándar de CGAN, por lo que no se tienen en cuenta versiones modificadas que conllevan un tiempo de cómputo más elevado que hubieran ralentizado la experimentación con porcentajes tan elevados.

7.3. Resultados Experimentales

El proceso de experimentación realizado para la aplicación de DA en el problema de predicción de la enfermedad Niemann-Pick Tipo C a partir de muestras metabolómicas se puede observar en la Figura 7.1. Esta experimentación, así como los resultados obtenidos, se encuentran publicados en los artículos Moreno-Barea et al. (2022b) y Moreno-Barea et al. (2022a). La Tabla A.6 del Apéndice A.4 muestra algunas características de los datos de referencia utilizados para los experimentos. A partir del conjunto de datos original, se realiza en primer lugar una división estratificada en conjuntos de entrenamiento y test para mantener la independencia total entre el proceso de generación de datos sintéticos y el entrenamiento de los modelos de clasificación y la evaluación de las métricas de predicción. En este contexto, el conjunto de test no se incluye en el proceso de generación de datos y se mantiene por separado hasta que se lleva a cabo la prueba final para obtener una evaluación honesta externa del rendimiento. Se establece una división del 60 % de los datos para entrenamiento y del 40 % para test en vista del pequeño número de muestras presentes en los conjuntos de datos de referencia. Esta proporción permite una mejor evaluación de los resultados de test. Además, todo el proceso de división de los datos y su aplicación se realiza mediante un procedimiento de validación cruzada para proporcionar una mejor estimación de los resultados adquiridos, reduciendo los efectos aleatorios que surgen del proceso. El proceso de generación de datos emplea el conjunto de entrenamiento para crear el número deseado de muestras sintéticas, a excepción del método SMOTE, que crea una cantidad suficiente de muestras para que el conjunto aumentado se equilibre. Cada modelo de DA aplicado sigue el procedimiento descrito en sus correspondientes capítulos de la presente tesis. En este proceso, la calidad de las muestras generadas puede

UNIVERSIDAD DE MÁLAGA

verse afectada por los datos de entrenamiento proporcionados, especialmente por el reducido número que exhiben; por lo que la implementación de una estrategia de validación cruzada se considera importante.

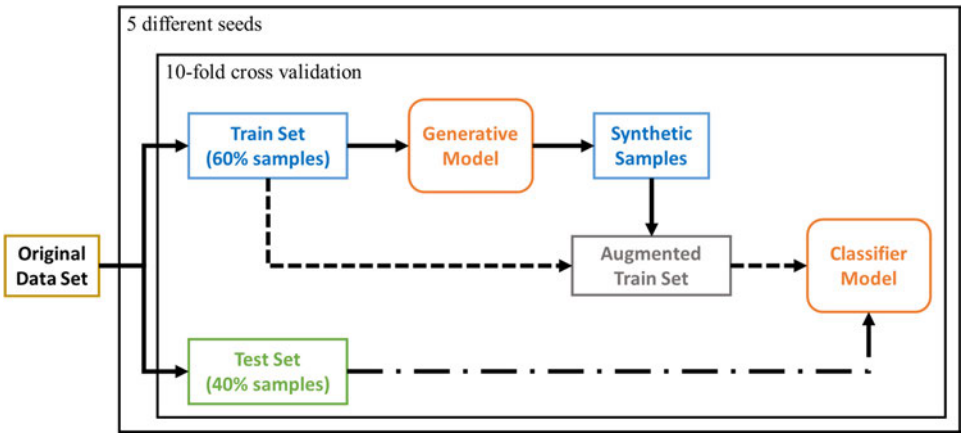

Figura 7.1: Diagrama de flujo de todo el proceso de experimentación. División de los datos en conjuntos de entrenamiento y test en un proceso de validación cruzada; aplicación del modelo generativo con el conjunto de entrenamiento y aumento con las muestras sintéticas; transformación de variables de metabolitos a componentes principales; entrenamiento del clasificador y predicción con el conjunto de test. El proceso se repite con 5 semillas.

Una vez creadas las muestras sintéticas, se realiza un análisis de componentes principales (PCA) sobre el conjunto de entrenamiento, obteniendo los vectores de puntaciones de las componentes y transformando todos los subconjuntos de datos (entrenamiento, prueba y sintéticos). Esta estrategia de PCA se aplica antes de cualquier experimentación realizada en el conjunto de datos. Para este enfoque se consideran los porcentajes de la varianza explicada por cada componente principal (PC), con el fin de seleccionar un número suficiente de componentes. En la Figura 7.2 se muestra la proporción de varianza explicada acumulada para el número de componentes principales seleccionadas. Se realizaron diversas pruebas preliminares seleccionando 3, 6, 10, 14 y 18 PCs para observar el rendimiento de clasificación que se podía alcanzar. Finalmente, el número satisfactorio de PCs seleccionado es de 14. Con los vectores de puntuación de las 14 PCs, la transformación de los subconjuntos modifica las muestras. De esta manera, las muestras presentan valores que están representados por vectores PC en lugar de los valores de las variables originales. Este proceso se realiza para eliminar cualquier nivel alto de correlación (*multicollinearity*) entre las variables que están presentes en el conjunto de datos metabolómico. Con los subconjuntos transformados, los datos sintéticos se agregan a los datos de entrenamiento, un proceso que da como resultado un conjunto de entrenamiento aumentado con el que se entrena el modelo clasificador. Una vez que se completa el entrenamiento, se

UNIVERSIDAD DE MÁLAGA

realiza la predicción del conjunto de prueba, obteniendo las métricas necesarias, que luego se promedian para la validación cruzada. El proceso se repite 5 veces con diferentes semillas para aliviar los efectos aleatorios.

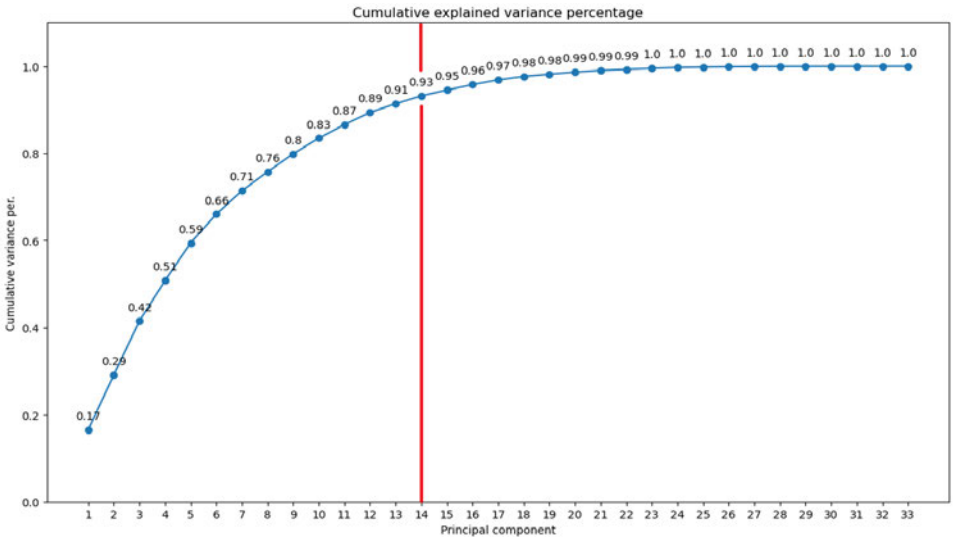

Figura 7.2: Proporción de varianza explicada acumulada con las diferentes componentes principales (PC) seleccionadas. Este proceso se realiza sobre el conjunto de datos original. El número de PC seleccionadas es de 14.

7.3.1. Rendimiento en predicción

La tabla 7.2 muestra los resultados de test obtenidos para los diferentes métodos y modelos DA aplicados, utilizando un enfoque de regresión logística como modelo clasificador en la experimentación. En esta tabla, "None" indica los resultados obtenidos con los datos originales y no se aumenta el número de muestras de datos, permitiendo observar si los resultados muestran una mejora del rendimiento en clasificación gracias a los conjuntos aumentados. Además, se incluyen los resultados obtenidos con los dos métodos combinatorios de muestras procedentes de dos modelos diferentes de DA. 'Comb 1' se refiere a los resultados obtenidos con una combinación entre las muestras creadas usando el modelo CGAN y la técnica SMOTE; y 'Comb 2' indica los resultados obtenidos con una combinación entre las muestras creadas usando el modelo CGAN y el método Noise Bal, cuando se aplica un nivel de porcentaje de DA del 500 % con Noise Bal.

La primera columna de la Tabla 7.2 ('Porcentaje') se refiere al tamaño de los datos sintéticos creados en comparación con el conjunto de entrenamiento original. Por lo tanto, un porcentaje de 100 indica que se crean tantas muestras como las del conjunto de entrenamiento. Un porcentaje de 50 indica que se crea la mitad del número de muestras. Finalmente, el valor 1000 indica que el número de muestras de entrenamiento se multiplica por diez.

UNIVERSIDAD DE MÁLAGA

Las siguientes columnas muestran los valores obtenidos para cada una de las métricas de test. Entre las métricas obtenidas se muestran:

- La capacidad predictiva (*accuracy*), que indica la proporción de predicciones correctas (verdaderos positivos y verdaderos negativos).
- La especificidad (*specificity*), relacionado con la capacidad del modelo clasificador para rechazar correctamente a los padres heterocigotos.
- La sensibilidad (*recall*), relacionada con la capacidad para detectar correctamente a los pacientes con la enfermedad. La sensibilidad también se conoce como la tasa de detección del modelo aplicado.
- La precisión (*precision*), que indica qué tan bueno es el modelo clasificador para predecir una muestra como positiva.
- La F_1 *score*, que permite una medida más confiable del desempeño del clasificador, particularmente bajo las circunstancias de este problema en la que la sensibilidad se vuelve más importante.

Tabla 7.2: Resultados de test obtenidos utilizando regresión logística como clasificador (fuente negrita indica mejor valor). Comb 1 es una combinación de muestras creadas con las estrategias CGAN y SMOTE. Comb 2 es una combinación de muestras creadas con los enfoques CGAN y NOISE Bal.

Método	Porcentaje	Accuracy	Specificity	Recall	Precision	F_1
None	None	88.42	**95.89**	60.00	79.37	68.34
CGAN	50	88.08	93.58	67.20	73.36	70.15
NOISE	1000	**89.33**	**95.89**	64.40	**80.50**	71.56
SMOTE	100	88.83	94.63	66.80	76.61	71.37
NOISE Bal	500	89.25	93.79	72.00	75.31	73.62
NOISE Bal	2000	88.58	91.16	**78.80**	70.11	**74.20**
Comb 1	50	87.58	91.89	71.20	69.80	70.49
Comb 2	100	87.17	90.63	74.00	67.52	70.61

Los resultados presentes en la Tabla 7.2 muestran que mediante la aplicación del método de adición de ruido se alcanza el mejor valor de capacidad de predicción, representando una mejora en comparación con el conjunto de datos sin aumento. Utilizando el método Noise con un porcentaje de aumento de 1000 % se consigue una capacidad predictiva del 89.33 % con una especificidad del 95.89 %, siendo este último el mismo valor que se obtiene con el conjunto de datos de referencia. El otro método destacado es el enfoque Noise Bal con un porcentaje de 2000 %. Este método obtiene los valores más altos de sensibilidad (78.8 %) y F_1 (74.2 %). Estos valores muestran una mejora sustancial en comparación con el análisis del conjunto de datos sin aumento, con el que se obtiene un 60 % de sensibilidad y un 68.34 % de F_1. Sobre el resto de modelos, mencionar que todos ellos mejoran los valores de sensibilidad y F_1 respecto a los datos originales, sin embargo, el modelo CGAN y las combinaciones obtienen valores más bajos de capacidad predictiva.

UNIVERSIDAD DE MÁLAGA

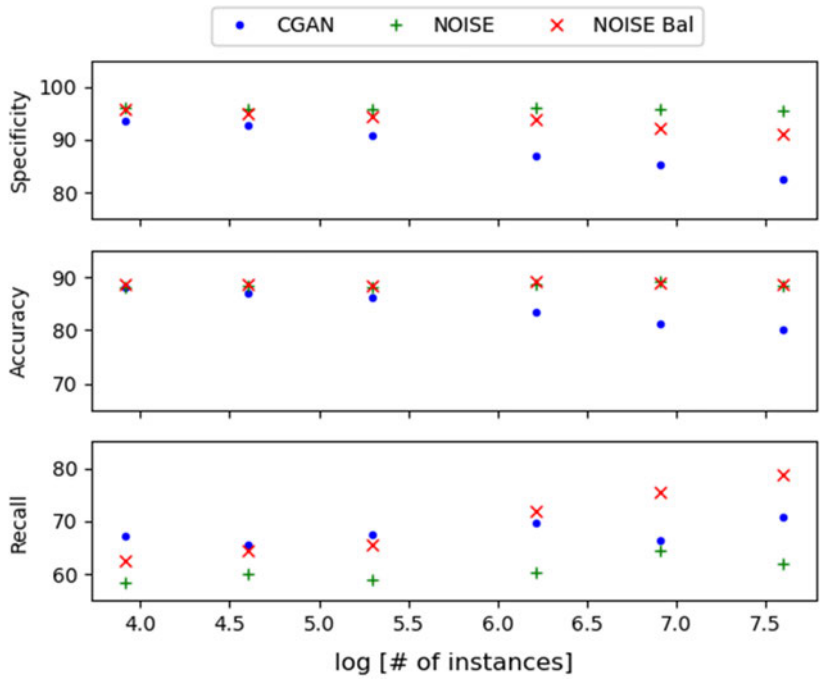

Figura 7.3: Comparación de la capacidad predictiva, especificidad y sensibilidad obtenidas usando regresión logística con diferentes modelos de DA, con respecto al logaritmo del número de instancias creadas. Los puntos representan los resultados con el modelo CGAN; las cruces con el método NOISE; y las aspas con el método NOISE Bal.

En la Figura 7.3 se presentan análisis adicionales de la influencia del tamaño de los conjuntos de datos en los resultados obtenidos con diferentes métodos de DA. Aquí, se presentan tres métricas de test (capacidad predictiva, especificidad y sensibilidad) obtenidas con tres métodos de DA (CGAN, Noise y Noise Bal), *versus* el número de instancias en una escala logarítmica en el eje de abscisas. Teniendo en cuenta los diferentes modelos, para los métodos Noise y Noise Bal, los resultados de capacidad predictiva obtenidos son aproximadamente estables con respecto al incremento en el número de instancias creadas. Además, los valores de especificidad son estables con el método Noise y disminuyen ligeramente para Noise Bal. Los valores de sensibilidad aumentan ligeramente con Noise, y claramente existe una correlación positiva significativa entre la sensibilidad y el número de instancias creadas mediante el método Noise Bal. Los resultados obtenidos con el modelo CGAN indican una correlación negativa entre el número de instancias creadas y la ganancia de especificidad y capacidad de predicción. Sin embargo, también se observó una correlación positiva para la ganancia de sensibilidad.

La Figura 7.4 presenta un análisis de la distribución de los resultados de test obtenidos por las diferentes técnicas de DA. Esta figura muestra los

UNIVERSIDAD DE MÁLAGA

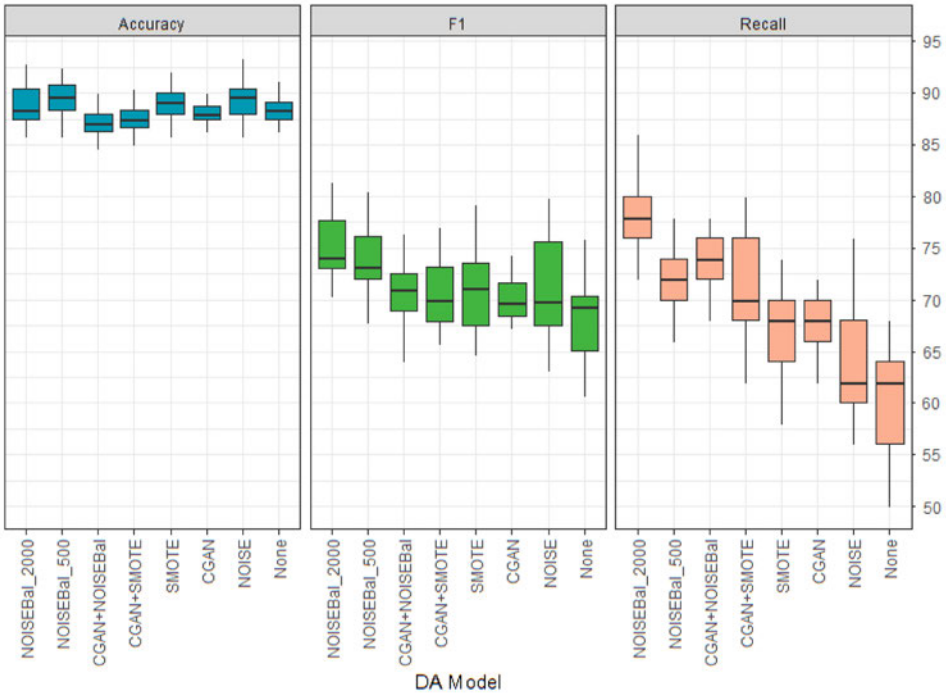

Figura 7.4: Comparación de las métricas de test obtenidas con diferentes modelos de DA y una regresión logística como sistema clasificador.

resultados en términos de capacidad predictiva, sensibilidad y F $_1$. Cabe señalar que los resultados se han obtenido utilizando los porcentajes seleccionados en la Tabla 7.2. La Figura 7.4 muestra una mayor dispersión de los resultados para los métodos de adición de ruido que para los métodos que incluyen la generación de muestras con el modelo CGAN. También muestra que la mejora potencial en los resultados es mucho mayor cuando se realiza un aumento del tamaño de los datos que con respecto a la obtenida con el uso únicamente del conjunto de datos original. A destacar las notables mejoras obtenidas en la métrica de sensibilidad gracias a la aplicación de DA.

Los resultados anteriores se obtuvieron utilizando una regresión logística como modelo clasificador. A modo de comparación, también se implementa un sistema de Random Forest (RF) (Breiman, 2001). Los resultados obtenidos con los diferentes métodos y modelos de DA aplicados se muestran en la Tabla 7.3. Esta tabla muestra el porcentaje de datos sintéticos creados y los valores de prueba obtenidos para la capacidad predictiva, especificidad, sensibilidad, precisión y F_1. En este caso, el mayor valor de capacidad de predicción se obtiene aumentando el conjunto con las muestras sintéticas creadas con SMOTE (86.67 %). Con respecto a las métricas de sensibilidad y F_1, los mayores valores se alcanzan de nuevo con el método Noise Bal. Los resultados muestran un valor de sensibilidad del 77.6 % utilizando un incre-

UNIVERSIDAD DE MÁLAGA

mento del tamaño de datos de 2000 % y un valor de F_1 de 68.62 % utilizando un incremento del 500 %. Esto demuestra una mejora sustancial con respecto a los resultados obtenidos cuando no se aplica ningún método de aumento de datos (39.6 % de sensibilidad, 52.38 % de F_1).

Tabla 7.3: Resultados de test obtenidos utilizando un sistema de Random Forest como clasificador. Comb 1 y Comb 2: descritas en la Tabla 7.2.

Método	Porcentaje	Accuracy	Specificity	Recall	Precision	F_1
None	None	85.00	**96.95**	39.60	77.34	52.38
CGAN	100	86.58	93.68	59.60	71.29	64.92
NOISE	1000	86.17	96.63	46.40	**78.38**	58.29
SMOTE	100	**86.67**	93.16	62.00	70.45	65.96
NOISE Bal	500	86.58	90.84	70.40	66.92	**68.62**
NOISE Bal	2000	78.92	79.26	**77.60**	49.62	60.53
Comb 1	100	83.17	85.68	73.60	57.50	64.56
Comb 2	50	85.33	88.63	72.80	62.76	67.41

La Figura 7.5 muestra la influencia que tiene el tamaño de los conjuntos de datos en la capacidad predictiva, especificidad y sensibilidad de los

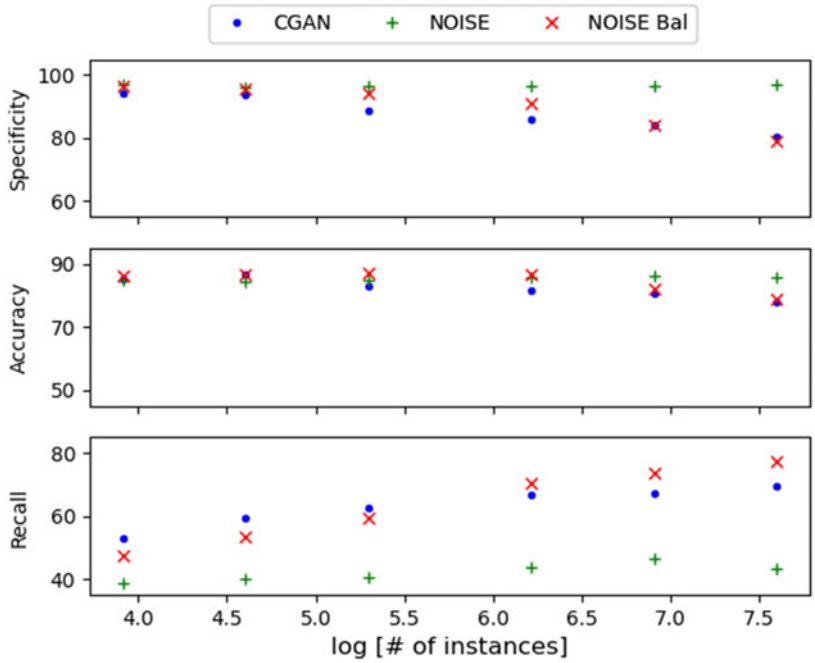

Figura 7.5: Comparación de la capacidad predictiva, especificidad y sensibilidad obtenidas usando un Random Forest con diferentes modelos de DA, con respecto al logaritmo del número de instancias creadas. Código de símbolos: igual que la Figura 7.3.

UNIVERSIDAD DE MÁLAGA

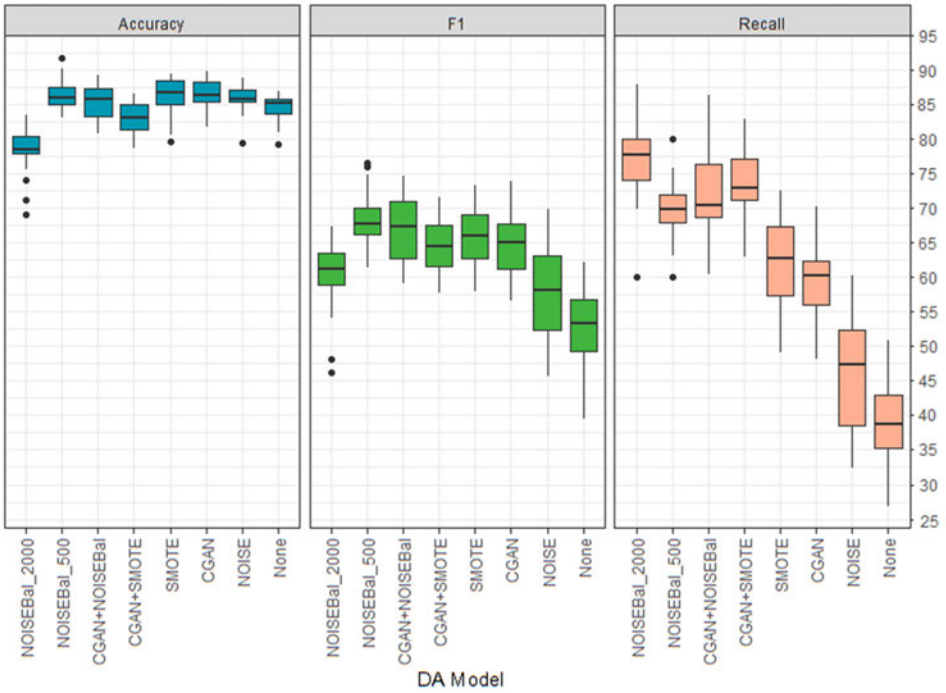

Figura 7.6: Comparación de las métricas de test obtenidas con diferentes modelos de DA y una estrategia Random Forest como sistema clasificador.

resultados obtenidos con los métodos CGAN, Noise y Noise Bal. En estos experimentos, se emplea de nuevo un sistema Random Forest, y los resultados del mismo se comparan con el número de instancias en una escala logarítmica en el eje de abscisas. Los resultados son similares a los obtenidos al utilizar una estrategia de regresión logística. Destacan los casos de CGAN y Noise Bal, con una correlación negativa entre el número de instancias creadas y la ganancia en especificidad, y también la capacidad de predicción. Por el contrario, existe una correlación positiva para la ganancia de sensibilidad.

La simetría de la distribución de los resultados de capacidad predictiva, sensibilidad y F_1 obtenidas mediante las diferentes técnicas de DA se muestra en la Figura 7.6. Los resultados indican que se puede lograr un mayor rendimiento del sistema de Random Forest mediante un aumento en el tamaño del conjunto de entrenamiento. La distribución de los resultados es similar a la distribución mostrada anteriormente con la regresión logística.

También se utiliza un modelo de máquina de vectores de soporte (SVM) como modelo clasificador. Los resultados obtenidos para los diferentes métodos y modelos DA aplicados se muestran en la Tabla 7.4. Esta tabla muestra el porcentaje de datos sintéticos creados y los valores de test obtenidos para la capacidad predictiva, especificidad, sensibilidad, precisión y F_1. Los re-

UNIVERSIDAD DE MÁLAGA

Tabla 7.4: Resultados de test obtenidos utilizando un sistema de SVM como clasificador. Comb 1 y Comb 2: descritas en la Tabla 7.2.

Método	Porcentaje	Accuracy	Specificity	Recall	Precision	F_1
None	None	83.17	**99.37**	21.60	**90.00**	38.84
CGAN	200	87.25	93.26	64.40	71.56	67.79
NOISE	100	88.25	96.84	55.60	82.25	66.35
SMOTE	100	89.25	95.68	64.80	79.80	71.52
NOISE Bal	500	**89.50**	94.53	70.40	77.19	**73.64**
NOISE Bal	2000	87.25	89.50	**79.60**	66.11	72.23
Comb 1	100	89.00	93.05	73.60	73.59	73.60
Comb 2	50	88.50	93.59	69.20	73.93	71.49

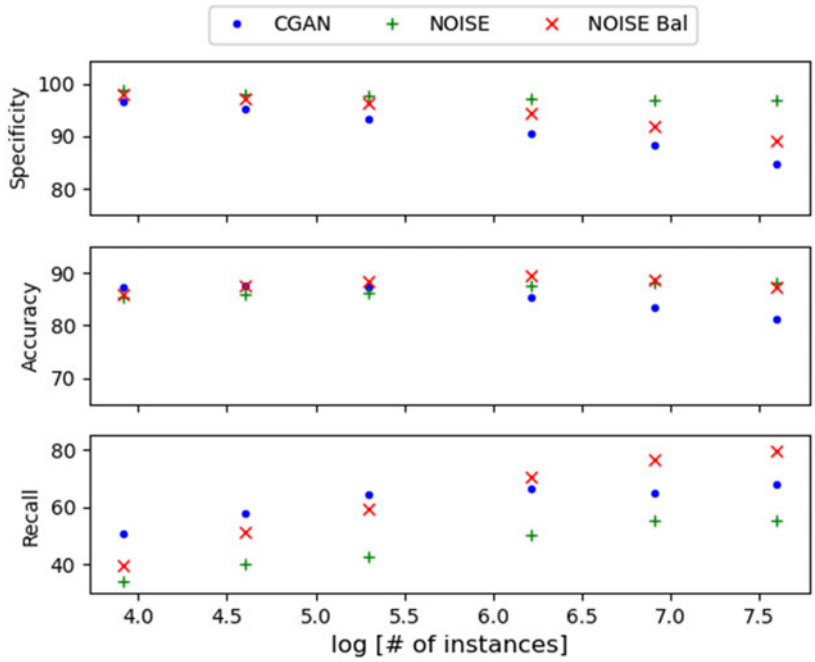

Figura 7.7: Comparación de la capacidad predictiva, especificidad y sensibilidad obtenidas usando un sistema de SVM con diferentes modelos de DA, con respecto al logaritmo del número de instancias creadas. Código de símbolos: igual que la Figura 7.3.

sultados de la Tabla 7.4 muestran una mejora en la capacidad predictiva, sensibilidad y F_1 en predicción con el método Noise Bal por encima de la obtenida con el conjunto de datos sin aumento. Utilizando el método Noise Bal con 500 %, se logra una capacidad predictiva de 89.50 % y un F_1 de 73.64 %. Noise Bal con un 2000 % también es muy eficaz, alcanzando el valor de sensibilidad más alto con un valor de 79.6 %. En este caso, todos los modelos de DA, incluidos el modelo CGAN y las combinaciones obtienen valores más

UNIVERSIDAD DE MÁLAGA

altos de capacidad predictiva, sensibilidad y F_1 en comparación con utilizar solo el conjunto de datos original, sin embargo, este arroja los valores más altos de especificidad (99.37 %) y precisión (90 %).

La influencia del tamaño del conjunto de datos en los resultados se puede ver en la Figura 7.7. De hecho, esta figura muestra la capacidad predictiva, especificidad y sensibilidad de los resultados obtenidos con los métodos CGAN, Noise y Noise Bal. Los resultados obtenidos se basan en la experimentación con un modelo SVM y se comparan con el número de instancias en una escala logarítmica. Esta figura muestra resultados similares a los anteriores en la relación entre la especificidad y el tamaño del conjunto de datos. También se pueden observar resultados similares para el aumento de datos realizado con la estrategia CGAN. Los resultados con ambos métodos de adición de ruido muestran una correlación positiva entre el número de instancias creadas y la ganancia de sensibilidad. La Figura 7.8 muestra la distribución de los resultados de capacidad predictiva, sensibilidad y F_1 obtenidas por las diferentes técnicas de DA empleadas. Los resultados mostrados son muy similares a los mostrados en la Figura 7.6 para el sistema de Random Forest.

Para realizar una comparación entre los métodos de DA con independencia del clasificador utilizado, se muestra en la Tabla 7.5 el promedio de los resultados mostrados anteriormente. Esta tabla muestra el promedio del

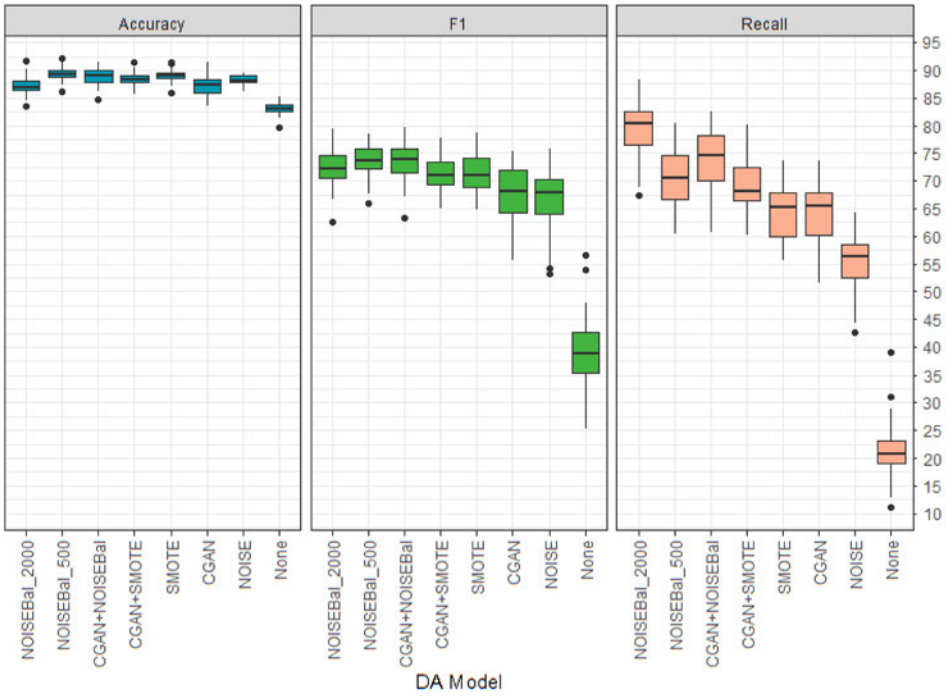

Figura 7.8: Comparación de las métricas de test obtenidas con diferentes modelos de DA y un sistema de SVM como clasificador.

UNIVERSIDAD DE MÁLAGA

Tabla 7.5: Resultados de test promedios obtenidos con la aplicación de los diferentes clasificadores. Comb 1 y Comb 2 representan las combinaciones de creación de muestra descritas en la Tabla 7.2.

Método	Porcentaje	Accuracy	Specificity	Recall	Precision	F_1
None	None	85.53	**97.40**	40.40	**82.24**	53.19
CGAN	117	87.30	93.51	63.73	72.07	67.62
NOISE	700	87.92	96.45	55.47	80.38	65.40
SMOTE	100	88.25	94.49	64.53	75.62	69.62
NOISE Bal	500	**88.44**	93.05	70.93	73.14	**71.96**
NOISE Bal	2000	84.92	86.64	**78.67**	61.95	68.99
Comb 1	83	86.58	90.21	72.80	66.96	69.55
Comb 2	67	87.00	90.95	72.00	68.07	69.84

porcentaje de datos sintéticos creados con cada modelo seleccionado y los valores promedio de test obtenidos para las distintas métricas evaluadas. Los resultados indican que mediante el entrenamiento sin aumento de datos se alcanzan los valores de especificidad y precisión más altos, 97.4 % y 82.24 % respectivamente, lo que iindica una mayor eficacia del modelo clasificador a la hora de predecir las muestras que no presentan la enfermedad. Sin embargo, se obtienen los valores más bajos de sensibilidad y F_1, indicando una escasa eficacia en detección de la enfermedad, lo que tiene una importancia vital. El método Noise Bal alcanza los valores más altos de capacidad predictiva, sensibilidad y F_1, destacando especialmente su aplicación con un porcentaje de aumento de 500 %, que alcanza los mejores valores de capacidad predictiva (88.44 %) y F_1 (71.96 %), y valores por encima de la media para el resto de métricas. Los demás modelos de DA obtienen valores de capacidad predictiva, sensibilidad y F_1 mejores que mediante la no aplicación de DA, pero no destaca ninguno. Mencionar que el uso de combinaciones de muestras no parece proporcionar una ventaja frente al resto. Con este análisis concluimos que el método de DA que sería seleccionado para su implementación real, es el método Noise Bal con un aumento de 500 % por su mejor rendimiento, especialmente en la detección de la enfermedad.

7.3.2. Análisis del conjunto generado

Para analizar cómo los métodos de DA utilizados en la experimentación replican la información presente en el conjunto de datos metabolómicos, se realizan un análisis de componentes principales (PCA) y un análisis discriminatorio de mínimos cuadrados parciales (PLS-DA). A través de estos métodos, la configuración de las muestras se puede visualizar en un espacio bidimensional (es decir, componente 2 *vs* componente 1). Esto proporciona un medio para verificar la distribución de las muestras sintéticas creadas y compararla con la distribución de las muestras originales.

UNIVERSIDAD DE MÁLAGA

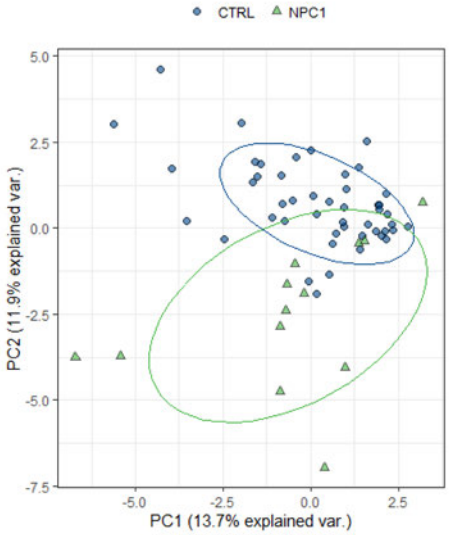

Figura 7.9: Gráfico de PCA con el conjunto original de la enfermedad NPC1. Código de colores: triángulos verdes, orina con enfermedad NPC1; círculos azul oscuro, orina de control de portadores heterocigotos.

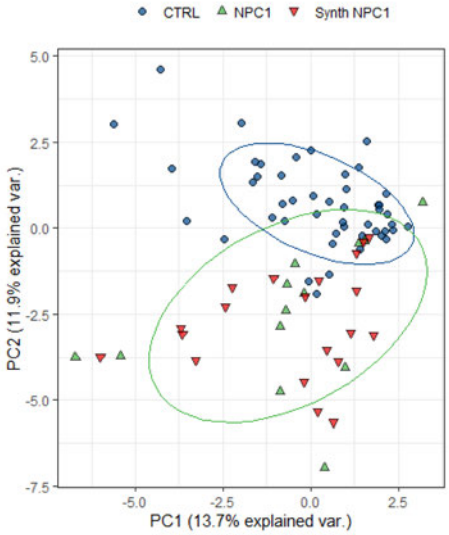

Figura 7.10: Gráfico de PCA con el conjunto original y las muestras generadas con SMOTE. Código de colores: igual que la Figura 7.9, con muestras de orina de la enfermedad NPC1 generadas por SMOTE representadas por triángulos rojos invertidos.

La Figura 7.9 muestra los resultados del modelo PCA obtenidos mediante el uso del conjunto de datos NPC1 original. El análisis revela que hay dos

UNIVERSIDAD DE MÁLAGA

clústeres significativos pertenecientes a las dos clases posibles que representan el "estado de enfermedad". El clúster perteneciente al grupo de portadores heterocigotos (CTRL, círculos azul oscuro), aparece como un grupo compacto de muestras. Por el contrario, el clúster formado por las muestras de orina donadas por los pacientes con la enfermedad NPC1 (NPC1, triángulos verdes), se encuentra más disperso, con varias muestras alejadas del núcleo principal de la distribución. También se puede notar que existe un área donde ambos grupos (control y NPC1) convergen y se solapan un reducido número de muestras.

Los resultados de las puntuaciones de PCA después de agregar las muestras generadas con SMOTE se muestran en la Figura 7.10. El ella se observa claramente cómo funciona la creación de muestras a través de SMOTE. Las muestras sintéticas (Synth NPC1, triángulos rojos inversos) se distribuyen a lo largo de la distribución de las muestras originales de orina de la enfermedad NPC1. Al generarse estas muestras sintéticas por interpolación de las muestras originales, se encuentran rellenando el espacio de muestras en una línea recta que interconecta pares de muestras originales entre sí.

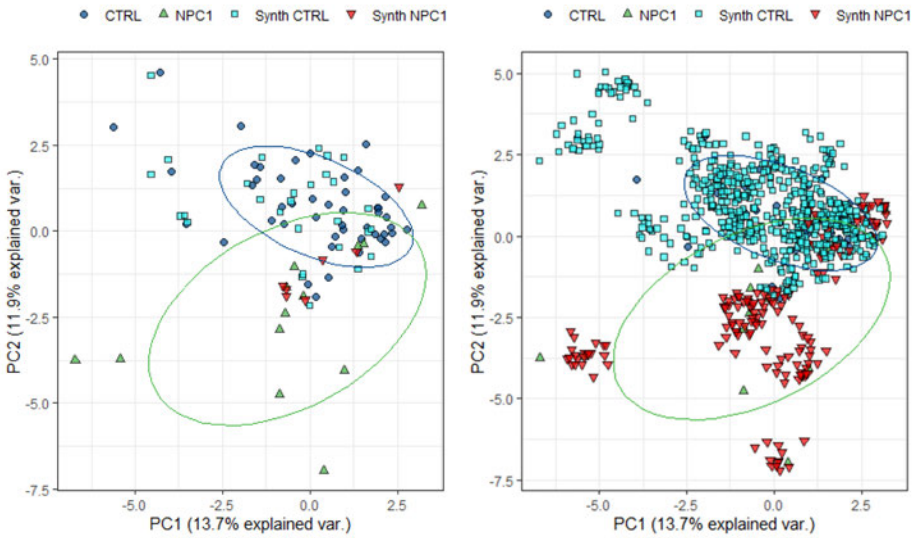

Figura 7.11: Gráfico de PCA con el conjunto original y las muestras generadas vía adición de ruido. El porcentaje de muestras generadas es 100 % (izquierda) y 2000 % (derecha). Código de colores: igual que la Figura 7.9, con cuadrados de color azul claro, muestras de orina generadas de control de portadores heterocigotos; triángulos rojos invertidos, muestras de orina generadas de la enfermedad NPC1.

La Figura 7.11 muestra la distribución de las muestras sintéticas creadas usando el método Noise con respecto a las muestras originales. Se utilizan porcentajes de aumento de 100 (izquierda) y 2000 (derecha), respectivamen-

UNIVERSIDAD DE MÁLAGA

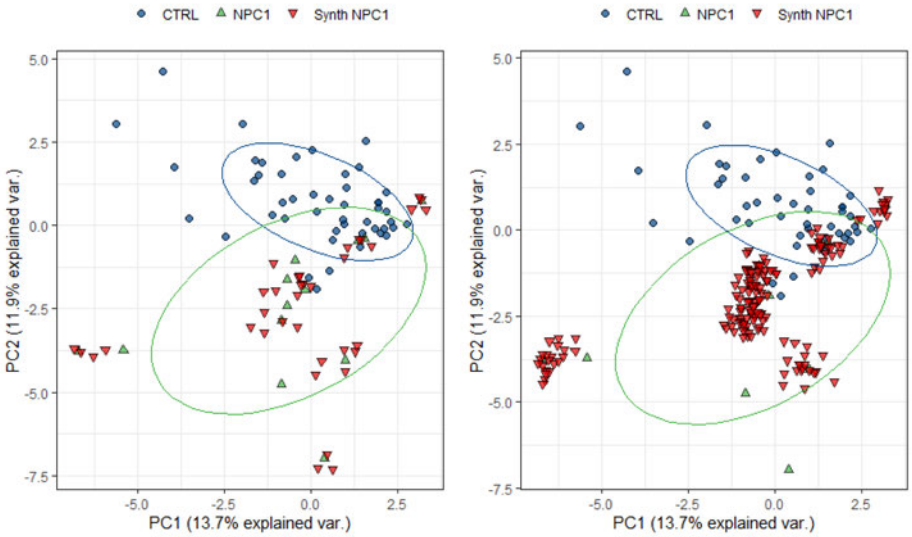

Figura 7.12: Gráfico de PCA con el conjunto original y las muestras generadas con la adición de ruido en un proceso de balanceo. El porcentaje de muestras generadas es 500 % (izquierda) y 2000 % (derecha). Código de colores: igual que la Figura 7.9, con triángulos rojos invertidos representando las muestras de orina NPC1 generadas.

te. La Figura 7.12 muestra la distribución de las muestras creadas por el método Noise Bal. Se utilizan porcentajes de aumento de 500 (izquierda) y 2000 (derecha), respectivamente. La diferencia entre ambos métodos de adición de ruido se puede apreciar completamente en este análisis, con la creación de muestras mediante Noise Bal exclusivamente para la clase minoritaria. Dado que la operación básica utilizada por ambos métodos es la misma en cuanto a modificaciones gaussianas de las características presentes en las muestras originales, se puede observar cómo las muestras sintéticas se agrupan alrededor de las muestras originales que modifican. Esta formación de pequeños clústeres alrededor de las muestras originales ayuda a los modelos clasificadores a aprender a clasificar estas muestras.

La generación de muestra de los métodos de DA anteriores difiere claramente de la seguida por el método CGAN. La Figura 7.13 muestra los resultados obtenidos con el uso de PCA y la distribución de las muestras sintéticas creadas por CGAN con respecto a las muestras originales. Se utilizan porcentajes de aumento 100 (izquierda) y 2000 (derecha), respectivamente. Con un mayor número de muestras, se puede observar mejor el comportamiento en generación de este modelo. La mayoría de las muestras se concentran en las áreas de dispersión de los datos originales. Esto produce una mezcla de muestras sintéticas asignadas a las clases de ambos grupos en la zona de convergencia originalmente presente, lo que puede conllevar un problema en

UNIVERSIDAD DE MÁLAGA

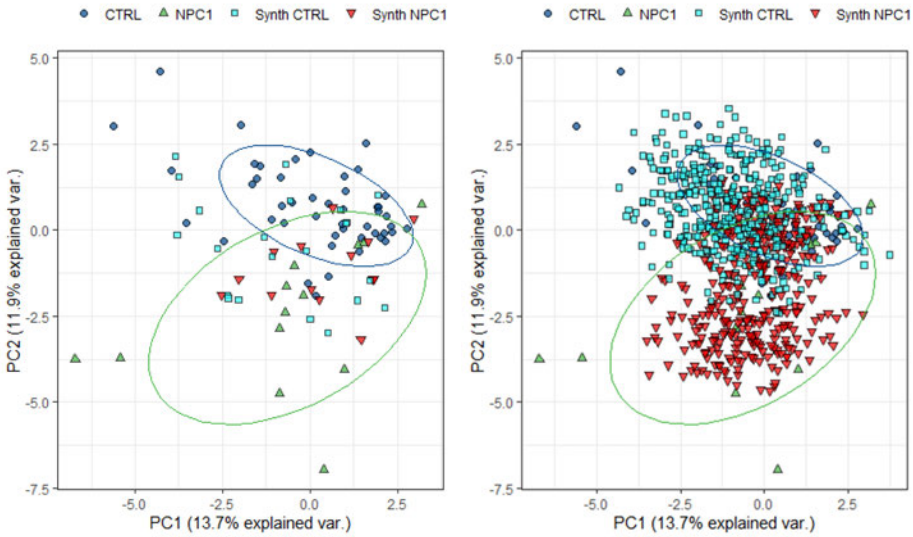

Figura 7.13: Gráfico de PCA con el conjunto original y las muestras generadas con el modelo CGAN. El porcentaje de muestras generadas es 100 % (izquierda) y 2000 % (derecha). Código de colores: igual que la Figura 7.11.

la clasificación. El modelo CGAN no crea ninguna muestra que se desvíe excesivamente de estos clústeres, y esto probablemente sea responsabilidad de la red discriminadora del modelo, que clasifica como falsas aquellas muestras demasiado alejadas de estos grupos y, por lo tanto, son descartadas.

Los resultados derivados de la representación gráfica de la componente 2 *versus* componente 1 de PLS-DA y la validación cruzada, brindan resultados que respaldan los análisis anteriores con PCA. Estos análisis se realizaron utilizando el software *MetaboAnalyst v4.0* (Universidad de Alberta y Consejo Nacional de Investigación, Instituto Nacional de Nanotecnología (NINT), Edmonton, AB, Canadá) (Chong et al., 2018). La aplicación de PLS-DA proporciona métricas que indican el poder predictivo del modelo (Szymańska et al., 2012), que son exactitud, bondad de ajuste (R^2) y bondad de predicción (Q^2). Durante el PLS-DA, se realiza una validación cruzada y luego se comparan los datos predictivos con los datos originales, y se suma el error de predicción en todas las muestras (suma de cuadrados residual prevista o PRESS). Para calcular el valor de Q^2, PRESS se divide por la suma inicial de cuadrados y se resta de 1 para igualarlo a la escala de R^2.

La Figura 7.14 (izquierda) también revela dos grupos significativos en los datos originales para las dos clases posibles que representan el "estado de enfermedad", con un área pequeña de convergencia entre los dos. La Figura 7.14 (derecha) muestra los resultados de exactitud, R^2 y Q^2 obtenidos con el conjunto de datos original con respecto al número de componentes utiliza-

UNIVERSIDAD DE MÁLAGA

dos en el proceso PLS-DA. El número óptimo de componentes seleccionados para el conjunto de datos sin aumento es de 2 componentes, obteniendo una exactitud de 0.873, un valor de R^2 de 0.781 y un valor de Q^2 de 0.457.

De manera similar a la representación por PCA de las muestras generadas con SMOTE, la Figura 7.15 (izquierda) muestra cómo las muestras creadas con esta técnica se distribuyen a lo largo del clúster 'real' a partir

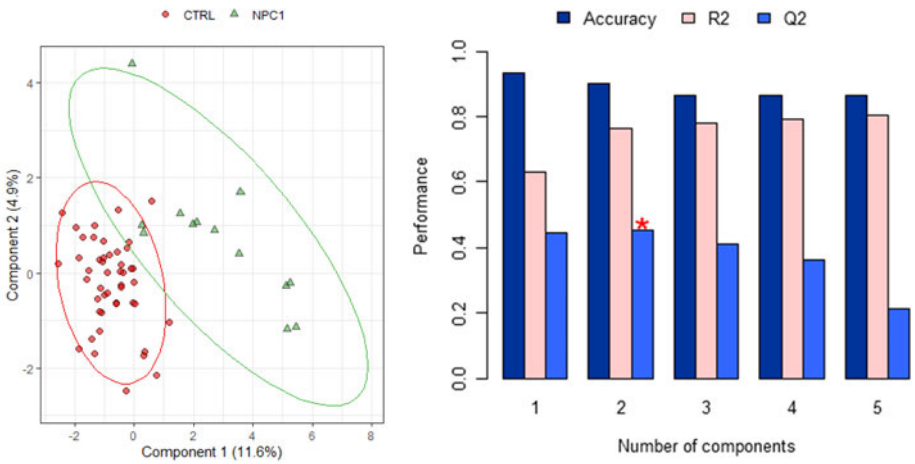

Figura 7.14: Gráfico de PLS-DA componente 2 *versus* componente 1 (izquierda) y resultados de validación cruzada para el conjunto de datos original (derecha). Código de colores (izq): triángulos verdes, orina con enfermedad NPC1; círculos rojos, orina de control de portadores heterocigotos.

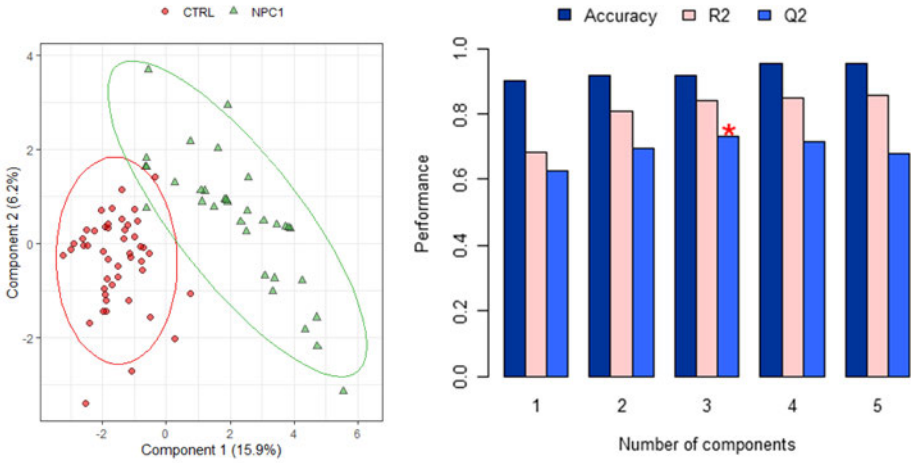

Figura 7.15: Gráfico de puntuaciones PLS-DA y resultados de validación cruzada adquiridos con los datos originales y las muestras generadas con la aplicación de SMOTE. Código de colores (izq): igual que la Figura 7.14.

UNIVERSIDAD DE MÁLAGA

del proceso de interpolación. La Figura 7.15 (derecha) muestra los resultados de la validación cruzada con respecto al número de componentes usados, en este caso con el conjunto de datos aumentado mediante la aplicación de SMOTE. Este modelo obtiene los resultados más efectivos con 3 componentes, con una exactitud de 0.930, un valor de R^2 de 0.840 y un índice de Q^2 de 0.743. Estos resultados confirman que este modelo da lugar a una mejora significativa con respecto al conjunto de datos original sin aumento de datos.

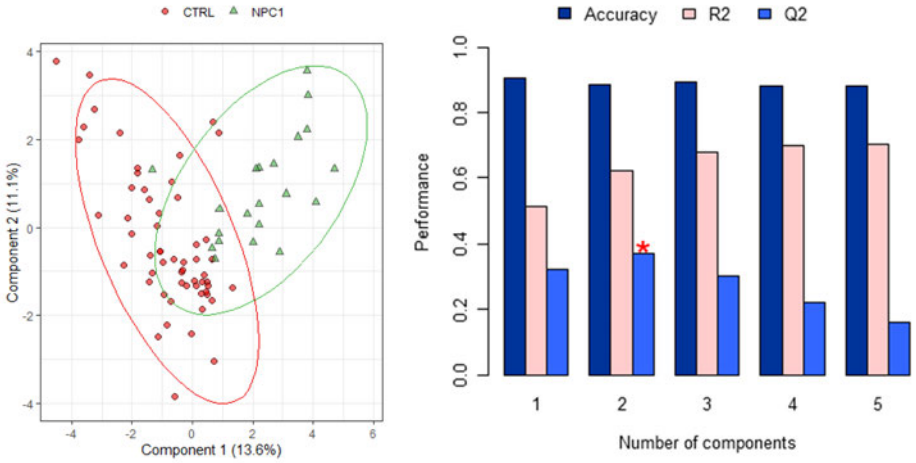

Figura 7.16: Gráfico de puntuaciones PLS-DA y resultados de validación cruzada obtenidos con los datos originales y las muestras generadas con el modelo CGAN. Código de colores (izq): igual que la Figura 7.14.

La Figura 7.16 (izquierda) muestra la gráfica de componente 2 *versus* componente 1 de PLS-DA cuando se usa el conjunto de datos aumentado con las muestras sintéticas CGAN. El porcentaje de aumento del conjunto de entrenamiento utilizado es del 50 %, con el que se obtienen los mejores resultados de capacidad predictiva (Tabla 7.2). El análisis revela un cambio sustancial en comparación con el resultado obtenido del conjunto de datos original. En este caso, el análisis de los componentes arroja puntuaciones diferentes, y esto cambia significativamente la distribución mostrada. Aunque la dispersión y el ángulo de las distribuciones son diferentes, aún es posible diferenciar ambos grupos, con una zona de convergencia más grande que la presente en el análisis inicial. La Figura 7.16 (derecha) muestra los resultados de la validación cruzada con respecto al número de componentes utilizados, empleando el conjunto de datos aumentado con las muestras sintéticas generadas con el modelo CGAN. El número óptimo de componentes seleccionados es dos, obteniendo una exactitud de 0.880, y valores de R^2 y Q^2 de 0.620 y 0.398 respectivamente, para un modelo con dos componentes. El cambio inducido en el análisis de los componentes y la mayor zona de convergencia,

UNIVERSIDAD DE MÁLAGA

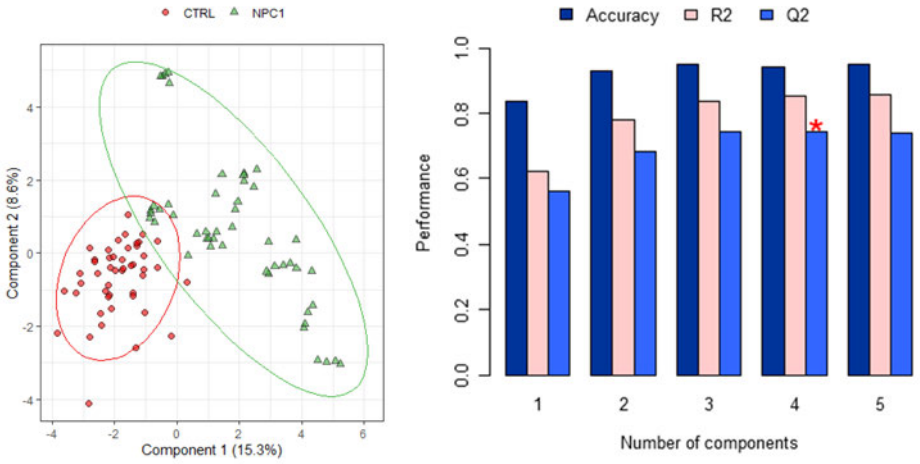

Figura 7.17: Gráfico de puntuaciones PLS-DA y resultados de validación cruzada obtenidos con los datos originales y las muestras generadas con la estrategia Noise Bal. Código de colores (izq): igual que la Figura 7.14.

puede ser la causa del bajo desempeño en la capacidad predictiva observada.

La Figura 7.17 (izquierda) muestra los resultados de PLS-DA cuando se utiliza el conjunto de datos aumentado con las muestras generadas por el método Noise Bal (con un porcentaje de aumento de 500 %). Los resultados son similares a los obtenidos con el conjunto de datos original y con el conjunto aumentado producido por la estrategia SMOTE. También son similares a los resultados obtenidos con PCA. Las muestras generadas para la clase de enfermedad NPC1 se acumulan alrededor de las originales. El modelo obtiene los resultados óptimos con 4 componentes, con una exactitud de 0.960, un valor de R^2 de 0.852 y un valor de Q^2 de 0.749. Estos resultados mejoran significativamente los resultados obtenidos en el conjunto de datos sin aumento, y también con el enfoque de aumento CGAN. También mejoran los resultados obtenidos con el método de aumento SMOTE, pero solo ligeramente.

7.3.3. Replicación de importancia de los metabolitos

Mediante la suma ponderada de los cuadrados de los pesos de PLS-DA, se obtienen los valores de la importancia de las variables en la proyección (VIP) con respecto a la componente 1. El valor VIP es una medida de la contribución de una variable individual en el modelo PLS-DA (Farrés et al., 2015). El análisis indica la importancia de cada metabolito en el proceso de diferenciación de los clústeres entre el grupo de portadores heterocigotos (control) y las muestras de orina de la enfermedad NPC1. Esto identifica los metabolitos que pueden emplearse como biomarcadores valiosos.

UNIVERSIDAD DE MÁLAGA

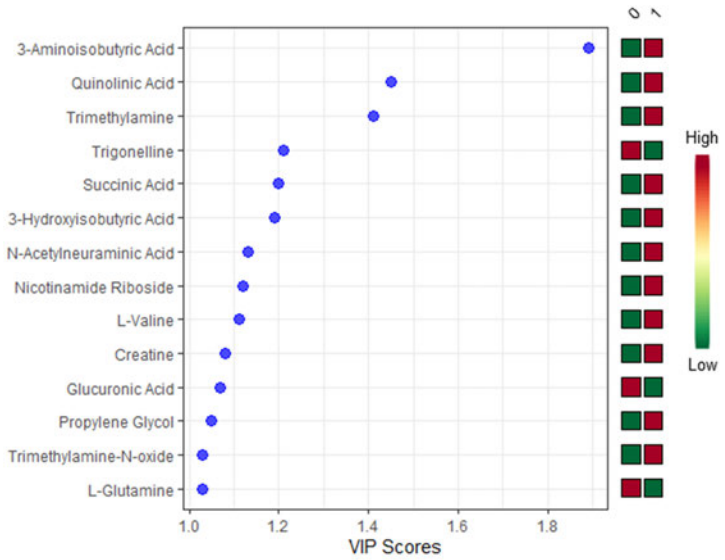

Figura 7.18: Valores la importancia de las variables en la proyección (VIP) derivados de la aplicación de PLS-DA al conjunto de datos original.

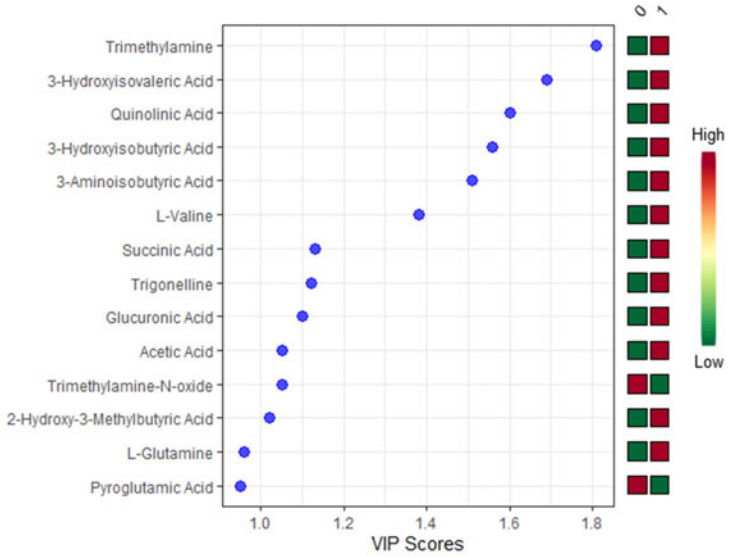

Figura 7.19: Valores de PLS-DA VIP obtenidos con el conjunto de datos original y las muestras generadas con la aplicación de SMOTE.

La Figura 7.18 muestra las puntuaciones PLS-DA VIP obtenidas mediante la aplicación del conjunto de datos original. Se muestran los valores de los 14 metabolitos principales. Los recuadros de colores de la derecha indican las concentraciones relativas del metabolito correspondiente en cada grupo estudiado (0: control; 1: enfermedad NPC1). El metabolito más destacado

UNIVERSIDAD DE MÁLAGA

es el ácido *3-aminoisobutyric* (*3-aminoisobutyrate* en los valores de pH normales de la orina humana), con un valor de VIP igual a 1.89 (los valores mayores de 1.00 se consideran significativos).

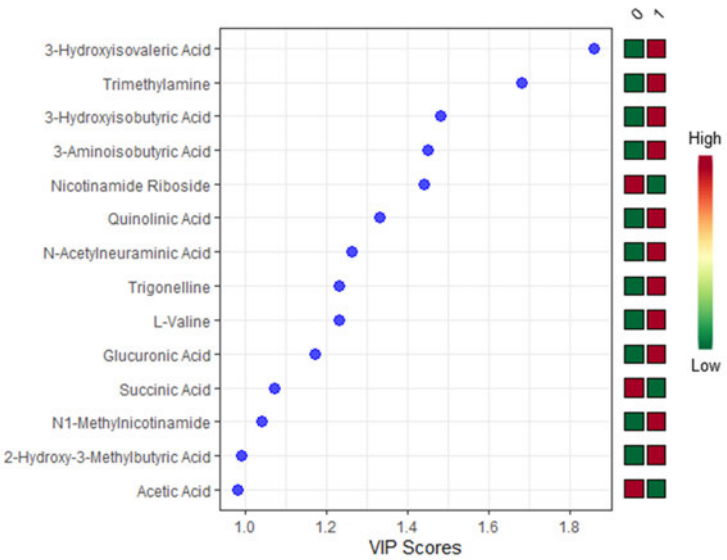

Figura 7.20: Valores de PLS-DA VIP obtenidos con el conjunto de datos original y las muestras generadas con el método Noise Bal (usando un 500 % de muestras sintéticas).

La Figura 7.19 muestra los resultados del análisis VIP con el conjunto de datos aumentado utilizando las muestras sintéticas creadas con el método SMOTE. Los metabolitos más destacados son la *trimethylamine* con un valor de VIP de 1.81 y el ácido *3-hydroxyisovaleric* con un valor de VIP de 1.69. El metabolito de ácido *3-aminoisobutyric* ocupa la posición 5 en este análisis.

El análisis VIP con el conjunto de datos aumentado utilizando las muestras sintéticas creadas con Noise Bal, refleja un resultado similar al obtenido con SMOTE. De hecho, la Figura 7.20 revela que los metabolitos que presentan una puntuación más alta con este conjunto aumentado son la *trimethylamine* con un valor de VIP de 1.68 y el ácido *3-hydroxyisovaleric* con un valor de VIP de 1.86. El metabolito de ácido *3-aminoisobutyric* ocupa la posición 4 en este análisis.

La Figura 7.21 muestra el análisis VIP con el conjunto de datos aumentado utilizando las muestras sintéticas creadas con el modelo CGAN. En este caso, los resultados presentan más diferencias que las observadas para los otros métodos aplicados. De hecho, los metabolitos más destacados son la *trigonelline* con un valor de VIP de 1.79 y el ácido *pyroglutamic* con un valor de VIP de 1.70. El metabolito de ácido *3-aminoisobutyric* ocupa la posición 8 en este análisis.

UNIVERSIDAD DE MÁLAGA

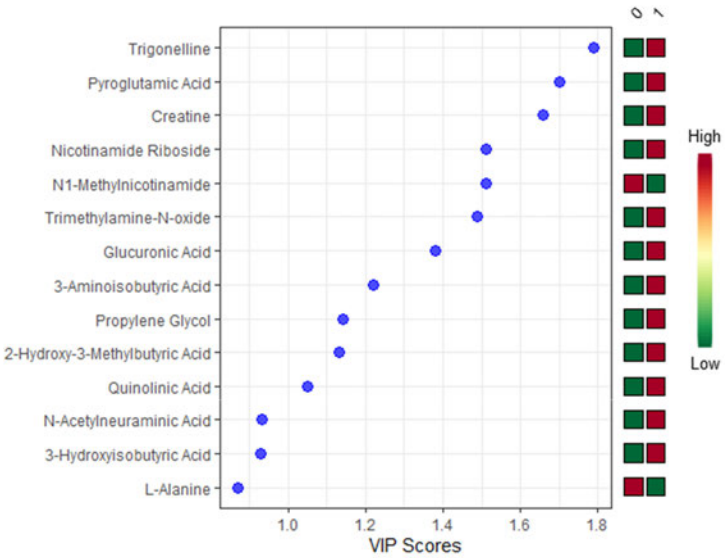

Figura 7.21: Valores de PLS-DA VIP obtenidos con el conjunto de datos original y las muestras generadas con el modelo CGAN.

Los resultados de las puntuaciones VIP que se muestran en todas las figuras anteriores se resumen en la Tabla 7.6. Esta tabla muestra los valores de VIP obtenidos para cada uno de los 14 principales marcadores metabolitos utilizando los diferentes conjuntos de datos aumentados. Los resultados muestran que solo hay 4 metabolitos que no están incluidos en el conjunto de datos original y que están presentes en el análisis con la aplicación de los métodos SMOTE y Noise Bal. Especialmente, los resultados obtenidos con estos dos enfoques son similares, presentando metabolitos que ocupan posiciones de clasificación equivalentes. Entre estos, cabe destacar los siguientes metabolitos: *trimethylamine* (posiciones 1 y 2); ácido *3-hydroxyisovaleric* (pos. 2 y 1); ácido *3-hydroxyisobutyric* (pos. 4 y 3); ácido *3-aminoisobutyric* (pos. 5 y 4); y *trigonelline* (pos. 8 y 8). La similitud entre las puntuaciones VIP obtenidas con estos métodos de DA puede deberse al hecho de que ambos son métodos de sobremuestreo. El mayor número de muestras sintetizadas para la clase minoritaria (enfermedad NPC1) influye significativamente en el análisis, lo que indica una mayor relevancia de estos metabolitos para separar este grupo del grupo de control de los portadores heterocigotos. La Tabla 7.6 también muestra el análisis con el conjunto aumentado utilizando el modelo CGAN. Este presenta 5 metabolitos diferentes de los deducidos con el conjunto de datos original. Cabe señalar la diferencia en importancia del metabolito de ácido *pyroglutamic*, en la posición 2 con CGAN y en la posición 19 con el conjunto de datos original. De manera similar, debe destacarse la diferencia entre el metabolito de *trimethylamine* en la posición 16 con el enfoque CGAN y la posición 3 con respecto al conjunto original.

UNIVERSIDAD DE MÁLAGA

Tabla 7.6: Valores de PLS-DA VIP obtenidos con los diferentes métodos de DA (los paréntesis representan la posición de ranking relativo de cada metabolito para cada método aplicado).

Metabolito	Original	SMOTE	Noise Bal	CGAN
3-Aminoisobutyric Acid	1.89 (1)	1.51 (5)	1.45 (4)	1.22 (8)
Quinolinic Acid	1.45 (2)	1.60 (3)	1.33 (6)	1.05 (11)
Trimethylamine	1.41 (3)	1.81 (1)	1.68 (2)	0.79 (16)
Trigonelline	1.21 (4)	1.12 (8)	1.23 (8)	1.79 (1)
Succinic Acid	1.20 (5)	1.13 (7)	1.07 (11)	0.70 (21)
3-Hydroxyisobutyric Acid	1.19 (6)	1.56 (4)	1.48 (3)	0.93 (13)
N-Acetylneuraminic Acid	1.13 (7)	0.93 (15)	1.26 (7)	0.93 (12)
Nicotinamide Riboside	1.12 (8)	0.68 (20)	1.44 (5)	1.51 (4)
L-Valine	1.11 (9)	1.38 (6)	1.23 (9)	0.66 (22)
Creatine	1.08 (10)	0.78 (18)	0.65 (22)	1.66 (3)
Glucuronic Acid	1.07 (11)	1.10 (9)	1.17 (10)	1.38 (7)
Propylene Glycol	1.05 (12)	0.43 (28)	0.40 (27)	1.14 (9)
L-Glutamine	1.03 (13)	0.96 (13)	0.77 (19)	0.70 (20)
Trimethylamine-N-oxide	1.03 (14)	1.05 (11)	0.93 (16)	1.49 (6)
2-H-3-Methylbutyric Acid	1.02 (15)	1.02 (12)	0.99 (13)	1.13 (10)
Acetic Acid	1.00 (16)	1.05 (10)	0.98 (14)	0.77 (17)
3-Hydroxyisovaleric Acid	0.99 (17)	1.69 (2)	1.86 (1)	0.58 (25)
N1-Methylnicotinamide	0.99 (18)	0.86 (17)	1.04 (12)	1.51 (5)
Pyroglutamic Acid	0.84 (19)	0.95 (14)	0.89 (17)	1.70 (2)
L-Alanine	0.43 (28)	0.16 (30)	0.24 (31)	0.87 (14)

Un estudio realizado previamente del conjunto de datos analizado, y que involucró una serie de técnicas contemporáneas de análisis de metabolómica multivariante y una estrategia de análisis de componentes correlacionados (CCA), y sin ningún tipo de estrategias de aumento del conjunto de datos, reveló una serie de biomarcadores que eran valiosos para distinguir entre los perfiles ^{1}H NMR urinarios de pacientes NPC1 y sus controles sanos heterocigotos (parentales). Estos biomarcadores incluían el aminoácido de cadena-ramificada *valine*, *3-aminoisobutyrate*, *quinolinate*, *succinate*, *trigoneline*, *2-hydroxy-3-methylbuyrate* y *L-alanine*. Estos metabolitos y sus clasificaciones de importancia relativa eran similares a los que se informan en esta investigación, en la Tabla 7.6. De manera similar, los enfoques de análisis de inteligencia computacional que involucran técnicas de algoritmos genéticos (GA) proporcionaron evidencia para respaldar estas observaciones. Se quieren destacar además dos metabolitos, la *trigoneline* y algunos intermediarios adicionales de la *nicotinamide*, biomarcadores que obtienen posiciones altas en el análisis realizado, pero que no pueden ser distintivos. La *trigoneline* es un metabolito de la cafeína y, por lo tanto, se esperaría que los padres (grupo de control heterocigoto) ingirieran más de este agente a través del consumo de café. De igual forma, la nicotina corresponde al consumo de tabaco por parte de los padres.

UNIVERSIDAD DE MÁLAGA

A partir de este estudio y de los resultados disponibles en Ruiz-Rodado et al. (2014), parece que los BCAA urinarios como la *valine*, los BCAA y el producto de degradación de la *timine 3-aminoisobutyrate*, y quizás también el metabolito *quinolinate*, pueden emplearse como valiosos biomarcadores para el diagnóstico, y quizás incluso monitorización, de la enfermedad NPC1. Sin embargo, existe un requisito principal para su validación, y será necesario lograrlo mediante una comparación de estos niveles de biomarcadores urinarios en pacientes tratados con fármaco selectivo y no tratados con NPC1. Dos terapias de éxito moderado empleadas para controlar esta enfermedad y atenuar su actividad son el inhibidor de la *glucosylceramide synthase* administrado por vía oral, y más recientemente también el agente encapsulante de colesterol *2-hydroxypropyl-β-cyclodextrin*.

7.4. Conclusiones

En esta investigación se ha analizado el efecto del uso de diferentes técnicas de aumento de datos en el rendimiento de predicción que se puede obtener cuando se considera un conjunto de datos urinarios metabolómicos de NPC1. Si bien en el momento de esta investigación, había varios estudios relativamente nuevos que muestran la utilidad del uso de DA en problemas biomédicos con conjuntos de datos no estructurados (no imágenes, señales o series de tiempo), ninguno ha demostrado su aplicación y efecto en el análisis de conjuntos de datos procedentes de la metabolómica.

Los resultados mostrados en las Tablas 7.2, 7.3 y 7.4, indican que el DA puede conducir a un aumento en la capacidad predictiva con independencia de la técnica utilizada como modelo clasificador en la experimentación. Concretamente, teniendo en cuenta el enfoque de regresión logística como modelo clasificador, con el cual se obtiene el valor de capacidad predictiva más alto mediante un análisis más convencional realizado en el conjunto de datos original (88.42), el conjunto aumentado con la adición de ruido alcanza aproximadamente una mejora del 1 %. Sin embargo, esta mejora es pequeña, aunque dadas las características del problema, la métrica del porcentaje de aciertos no es representativa. Dado que el conjunto de datos está muy desequilibrado, es fácil obtener un alto valor de capacidad predictiva simplemente prediciendo todas las muestras como control. Además, tratando un problema de predicción de enfermedad como en este caso, implica más valor la capacidad del modelo para lograr que el mayor número de pacientes con la enfermedad sean diagnosticados como tales y no como pacientes control. Por lo tanto, las métricas de predicción más representativas para este problema son la sensibilidad y la puntuación F_1. A este respecto, la Tabla 7.2 muestra que al realizar el aumento de datos, el método de adición de ruido balanceado (Noise Bal) con un aumento del 2000 %, se obtiene una mejora aproximada del 19 % en la sensibilidad y una mejora del 6 % en F_1.

UNIVERSIDAD DE MÁLAGA

Resultados más generalizados se muestran en la Tabla 7.5, donde se observan los valores medios obtenidos con los enfoques de DA en las experimentaciones con los distintos modelos clasificadores. Si bien se encuentran ligeramente sesgados por el pobre rendimiento en sensibilidad obtenidos con Random Forest y el sistema de máquinas de vectores de soporte, pueden considerarse los resultados más relevantes, pues exhiben los resultados más efectivos obtenidos con cierta independencia del modelo clasificador. Los resultados demuestran que el método con mejor rendimiento es Noise Bal con un aumento del 500 %. Este método conduce a los mejores valores medios de capacidad predictiva (88.44) y F_1 (71.96) comparado con la aplicación únicamente de los datos originales (85.53 y 53.19 respectivamente). Los resultados obtenidos de la validación cruzada de PLS-DA respaldan este hecho (Fig. 7.17 derecha). El conjunto aumentado con datos creados con el método Noise Bal y 500 % DA proporciona aproximadamente una mejora del 9 % en la exactitud y una mejora de 0.3 (de 1) en la bondad de predicción (Q^2).

Para determinar la capacidad de los métodos DA para replicar la información metabólica, se realizó un análisis utilizando técnicas convencionales de análisis de datos multivariados. Los resultados de PCA y PLS-DA para el componente 2 *versus* componente 1 muestran claramente las diferencias en la generación de datos por cada método aplicado. Los métodos de adición de ruido implementados generan muestras sintéticas que se agrupan alrededor de las muestras originales que modifican, por lo tanto, estos métodos tienen una clara capacidad para replicar la información si el error aplicado no es excesivo. Respecto a la aplicación de la técnica SMOTE, el hecho de que el conjunto de datos original presente dos grupos separados con una pequeña zona de convergencia, evita las desventajas que puede presentar y refleja una buena capacidad de replicación. El análisis de los datos aumentados con el modelo CGAN refleja la capacidad del modelo para replicar información que se ajusta a la distribución de las muestras "reales". Sin embargo, también se puede concluir una desventaja a partir de estos resultados. Dada la función interna de la red discriminadora, cuando el generador intenta crear una muestra sintética cercana a valores atípicos (lejos del núcleo de la distribución), el discriminador los descarta. Los resultados de PLS-DA se ven afectados por las muestras sintéticas creadas por este modelo; esto se muestra claramente en el análisis gráfico del componente 2 *versus* componente 1 (Fig. 7.16). Por último, los resultados de las puntuaciones VIP concluyen que los métodos de DA son capaces de replicar la información metabolómica importante para encontrar biomarcadores, y apoyan otros resultados obtenidos en este campo indicando que la *valine*, los BCAA, el producto de *3-aminoisobutyrate*, *3-hydroxyisovalerate* y *3-hydroxyisobutyrate*, y los metabolitos *quinolinate* y *trimethylamine*, pueden emplearse como valiosos biomarcadores para el diagnóstico de la enfermedad NPC1.

UNIVERSIDAD DE MÁLAGA

UNIVERSIDAD
DE MÁLAGA

Capítulo 8

Predicción del Subtipo Molecular de Cáncer

Resumen: El cáncer de mama representa un grave problemas de salud, con 2.3 millones de nuevos casos al año a nivel mundial. Dentro de las vías de investigación, aumentar la capacidad del sistema inmunitario frente a las células cancerígenas puede prevenir la formación de distintos tipos de cáncer y permite experimentar con técnicas más eficaces de inmunoterapia, menos agresiva que la radioterapia o la quimioterapia. Sin embargo, el sistema inmunitario se ve afectado por el microambiente local para generar una respuesta eficiente. Esto implica realizar estudios personalizados para una región, lo que conlleva un número reducido de muestras de pacientes dentro de los estudios. Para minimizar esta desventaja, se ha experimentado con los diversos métodos de aumento de datos (DA) estudiados y desarrollados. Los experimentos se realizan sobre un conjunto de expresión génica del sistema inmunitario de pacientes de la región de Málaga afectadas por cáncer de mama, para predecir el subtipo molecular del cáncer que presentan, Los datos generados muestran una calidad suficiente para lograr una mejora en el rendimiento de la predicción del subtipo, especialmente en su clasificación binaria en subtipos de grado alto o bajo.

8.1. Introducción al Problema

Uno de los mayores problemas médicos a los que se enfrenta la sociedad es el cáncer. Según el Instituto Nacional del Cáncer (NCI) de los Estados Unidos de América, cáncer es el nombre con el que se designa a un conjunto de enfermedades relacionadas por el hecho de que en estas se produce un descontrol celular, donde algunas de las células del cuerpo empiezan a dividirse sin detenerse y se diseminan a los tejidos de alrededor. Actualmente,

UNIVERSIDAD DE MÁLAGA

el cáncer es la primera causa de muerte en países en vías de desarrollo, y la segunda en países desarrollados, solamente por detrás del amplio abanico de enfermedades cardiovasculares. Según datos revelados en el estudio GLOBOCAN (Sung et al., 2021) de la Agencia Internacional para la Investigación del Cáncer, se estima que en el año 2020 hubo una incidencia de 19.3 millones de nuevos casos y 10.0 millones de muertes por cáncer.

Más concretamente, el cáncer de mama representa hoy en día uno de los principales problemas de salud a nivel mundial debido al elevado número de casos nuevos que se diagnostican cada año. Representa el tumor más frecuente en las mujeres, con una estimación cercana a 2.3 millones de casos a nivel mundial, un 11.7 % de todos los casos de cáncer diagnosticados. Por suerte, la mortalidad de este es más reducida que la de otros tipos de cáncer, con una mortalidad cercana al 6.9 % del total relacionadas con cáncer, gracias a las mejoras en técnicas de detección temprana de cáncer de mama y a terapias menos agresivas. Aun así, se registraron 685.000 defunciones en 2020, que representa un aumento del 9.8 % de nuevos casos y del 9.2 % de mortalidad respecto al año 2018. Respecto a la diferencia según la región geográfica, las tasas de incidencia son un 88 % más altas en los países desarrollados que en los países en vías de desarrollo. Sin embargo, los países en vías de desarrollo presentan unas tasas de mortalidad un 17 % más altas en comparación.

Debido a la magnitud de esta enfermedad, hay incontables frentes para abordarla. Uno de los temas con mayor interés es la capacidad del sistema inmunitario para actuar contra a las células cancerígenas. El sistema inmunitario, o inmunológico, está implicado en el mantenimiento de la integridad del organismo y en la regulación de su propia función. Afectado por el microambiente local, este sistema determina la generación de una respuesta eficiente frente a antígenos extraños. Se sabe que una de las funciones más importantes del sistema inmunitario es conferir protección frente a patógenos externos como virus, bacterias y parásitos, sin embargo, fue un tema muy debatido si la prevención del cáncer es una función primaria del sistema inmune. Ehrlich (1908) fue uno de los primeros en introducir el concepto de inmunovigilancia al plantear la idea de que el sistema inmune podría ejercer un efecto represor sobre la formación de tumores. Anteriormente, Coley (1893) realizó experimentos en los que, tras la inoculación de cultivos bacterianos, pacientes presentaban una reducción tumoral, prolongando la supervivencia después del tratamiento. Este hecho fue investigado posteriormente, y se llegó a la conclusión de que estas bacterias activaban la producción masiva de las células del sistema inmune, las cuales son capaces de detectar la presencia y reconocer estructuras específicas de la superficie celular tumoral, o bien responder ante moléculas secretadas por el propio tumor, y una vez reconocidas las células tumorales, el sistema era capaz de eliminarlas. En base a este mayor conocimiento de la inmunología tumoral, Burnet (1971) y Thomas (1982) sugirieron que los linfocitos eran los responsables de reconocer y eli-

UNIVERSIDAD DE MÁLAGA

minar las células transformadas, y que la principal función de la inmunidad celular era proteger al organismo de enfermedades neoplásicas, quedando así formalmente postulada la "teoría de la inmunovigilancia tumoral".

A mediados de los años 90, se insufló el interés por la inmunovigilancia tumoral. Se produjo el descubrimiento de que la producción de una proteína por parte de linfocitos del sistema inmunitario podía defender a un individuo contra el crecimiento de tumores trasplantados o inducidos (Kaplan et al., 1998), y que la pérdida de otra componente de estos linfocitos hacía que los individuos fueran más propensos a desarrollar tumores inducidos (van den Broek et al., 1996). Ambos estudios revelaron que las componentes moleculares del sistema inmunitario estaban implicadas en el desarrollo de tumores primarios. En la última década, la inmunoterapia se ha convertido en una poderosa estrategia clínica para el tratamiento del cáncer. El número de aprobaciones de fármacos de inmunoterapia ha ido en aumento, con numerosos tratamientos en desarrollo clínico, e innumerables estudios en este ámbito que, de forma clara, han supuesto un empuje para permitir el conocimiento de los efectos del sistema inmune en el desarrollo y establecimiento de los tumores (Pardoll, 2012; Sharma et al., 2017; Martin et al., 2020).

El conjunto de datos utilizado en el trabajo presenta la expresión genética de las pacientes más el subtipo molecular del cáncer de mama. El conjunto de datos original recabado por el estudio incluía otros factores de la enfermedad, como son el grado histológico (el grado de diferenciación de las células cancerígenas con las células normales), el tamaño del tumor y el tipo histológico del mismo. Estos datos no son incluidos en el conjunto utilizado, ya que el análisis necesario para todos ellos sería complejo, y queremos centrarnos en el subtipo molecular del cáncer que presentan las pacientes.

Este subtipo molecular es la variable que va a ser tomada como clase en el análisis con algoritmos de aprendizaje supervisado. De acuerdo a estudios de los patrones de expresión genética relacionados con el pronóstico o con el riesgo de metástasis (Perou et al., 2000; Sørlie et al., 2001; Goldhirsch et al., 2013), se puede dividir el cáncer de mama en dos grandes grupos basándonos en la positividad para el receptor de estrógeno. Un primer grupo son las neoplasias de grado bajo, o tumores receptores de estrógeno positivo, que se dividen a su vez en distintos tipos denominados luminales; y un segundo grupo, las neoplasias de grado alto, o tumores receptores de estrógeno negativo, que se dividen principalmente en basales y sobreexpresión de HER2.

- **Luminal A (LA)**: Estos tumores se caracterizan por tener una expresión de receptores de estrógenos (RE) y progesterona (RP) positivos, Her2/neu negativo, y un bajo índice de proliferación Ki67 ($<14\,\%$). Son el subtipo más común y menos agresivo, con un buen pronóstico ante tratamiento hormonal. Tienen un bajo grado histológico y están asociados al incremento de edad.

UNIVERSIDAD DE MÁLAGA

- **Luminal B negativo (LBH-)**: Tumores que se caracterizan por tener una expresión de RE positivo, Her2 negativo y un índice Ki67 alto (>14 %), o bien un RP positivo y un índice Ki67 medio (14-19 %).
- **Luminal B positivo (LBH+)**: Estos tumores se caracterizan por tener expresión de RE positivo y Her2 positivo. Estos pueden presentar cualquier expresión de RP y valor de índice de proliferación Ki67. Los luminales B, ya sean positivo o negativo, son similares a los luminales A, sin embargo, tienen un pronóstico más desfavorable que estos y requieren, en la mayoría de casos, la administración de quimioterapia en las pacientes.
- **Her2 positivo puro (Her2)**: Tumores que se caracterizan por tener una expresión de receptores hormonales (RH) negativos y Her2 positivo. Se asocian con caracteres histológicos altos y representan un subtipo realmente agresivo, asociados a mal pronóstico además de una respuesta mala a las terapias.
- **Triple negativo (TN)**: Estos tumores se caracterizan por tener expresión de RE y RP negativos, al igual que Her2 negativo y receptores hormonales negativos. Los tumores de este tipo pueden ser tratados mediante radioterapia o quimioterapia, según el tamaño de los mismos, aunque suele ser un subtipo agresivo. Las mujeres con edades menores a los 40 años que presentan este subtipo de cáncer tienen un mayor riesgo. Es el subtipo más frecuente en mujeres afroamericanas (Dietze et al., 2015; Prakash et al., 2020).

El poder predecir el subtipo del cáncer de mama a partir de la expresión génica tiene en nuestro trabajo doble intención. No solo predecirlo a partir de la información obtenida del sistema inmunitario del paciente, sino conocer la relación entre expresión génica y subtipo, de forma que se pueda identificar aquellos casos que se beneficien de las diferentes terapias disponibles.

8.2. Implementación de los modelos

Con el objetivo de comprobar el efecto del DA en el rendimiento predictivo sobre un conjunto de expresión génica del sistema inmunitario para el estudio del subtipo molecular del cáncer de mama, y realizar una comparación entre diversos métodos de DA, se utilizan la adición de ruido gaussiano, la técnica SMOTE, el modelo CGAN estándar y en un proceso de sobremuestreo, la técnica SMOTE, los modelos CGAN y ACGAN estándar, y el modelo CGAN en el marco adaptativo del modelo clasificador generativo expuesto en el Capítulo 5, incluyendo una modificación de balanceo de datos. En esta sección se resume la implementación de estos modelos, incluyendo el modelo profundo utilizado como clasificador en la experimentación.

UNIVERSIDAD DE MÁLAGA

En primer lugar, se utilizan los métodos de adición de ruido gaussiano explicados en el Capítulo 4 sección 4.2. Estos seleccionan aleatoriamente muestras del conjunto de entrenamiento y modifican aleatoriamente un 25 % de las características, calculando el nuevo valor mediante la adición de ruido que surge de una distribución normal aleatoria. El valor desviación utilizado para la distribución es de 0.2, pues, aunque preprocesamos los datos con PCA para trabajar sobre componentes principales, en este caso, los datos se normalizan para la aplicación de los modelos de aumento. De este modo, un valor de 0.2 ha sido suficiente para la creación de muestras que difieran significativamente de las originales. Tras la adición del ruido, se verifica que los nuevos valores no excedan los límites establecidos (máximo y mínimo) para su característica, se desnormaliza la muestra creada para que se encuentre escalada con las componentes principales, y se repite el proceso con la siguiente muestra seleccionada.

A parte de la versión 'estándar'de adición de ruido, también se aplica el método con el propósito de realizar sobremuestreo. Este método, abreviado como 'Noise Bal', únicamente se diferencia del método estándar en la selección aleatoria, pues solo selecciona y genera muestras de entrenamiento de la clase minoritaria. En esta experimentación, teniendo en cuenta que si queremos realizar la predicción multiclase del subtipo se tienen 3 clases minoritarias distribuidas con un 12.1 % (LBH+), 8.5 % (Her2) y 9.7 % (TN), respecto con dos clases mayoritarias distribuidas con un 36.9 % (LBH-) y 32.7 % (LA), se toman aleatoriamente muestras de las clases minoritarias para la adición de ruido y creación de muestras sintéticas. En cuanto a la técnica SMOTE aplicada, se utiliza la versión estándar (Chawla et al., 2002), detallada en el Capítulo 2 sección 2.1.3. Para la clasificación multiclase, se aplica este método con la diferenciación 'LBH+ *vs.* LBH-', 'Her2 *vs.* LBH-' y 'TN *vs.* LBH-'. De este modo se crean muestras sintéticas para estas tres clases. Para la implementación de esta técnica se utiliza el módulo *imbalanced-learn* (Lemaître et al., 2017) con los valores por defecto ($sampling_strategy =$ '$auto$'; $k_neighbors = 5$).

Los detalles de implementación y arquitectura del modelo CGAN utilizado para el proceso generativo de muestras sintéticas, son los mismos que los utilizados en el experimento de predicción de eventos en cáncer y de predicción de la enfermedad de Niemman-Pick Tipo C. Estos están presentes en la Tabla 6.1, mostrando una distinción entre la red generador y discriminador, y exponiendo para cada una el tipo de capa, tamaño de salida de cada una (número de neuronas presentes en cada capa), el uso de la normalización por lotes, la función de activación utilizada en la capa, los algoritmos de optimización utilizados, la parametrización de la función de activación y el software usado como backend. El único detalle que es diferente en la implementación del modelo CGAN utilizado para esta experimentación es la variación del tamaño del vector latente de ruido utilizado como entrada

UNIVERSIDAD DE MÁLAGA

de la red generadora. Este vector latente pasa a tener un tamaño igual al número de características presentes en el conjunto de datos.

Además del modelo CGAN 'estándar', se aplica el modelo ACGAN, abreviatura de *Auxiliary Classifier* GAN (Odena et al., 2017). Este modelo se encuentra explicado con más detalle en el Capítulo 2 sección 2.3.4. Al igual que el modelo CGAN, ACGAN proporciona tanto el espacio latente como la etiqueta de clase como entrada al generador, encontrándose su principal diferencia en el discriminador. En el modelo ACGAN solo se proporciona la muestra como entrada del discriminador, sin la etiqueta de clase, además esta red presenta una doble funcionalidad, debe predecir si la muestra dada es real o falsa pero también debe predecir la etiqueta de clase de la muestra. De este modo, la Tabla 8.1 muestra la configuración de la red discriminador implementada para el modelo ACGAN, que varía el tamaño de la capa de entrada del discriminador, siendo uno menos al no incluir la etiqueta, e incluye una capa de salida para diferenciar las muestras reales y falsas con una neurona y función de activación sigmoide, y dos capas adicionales para la predicción de la etiqueta de clase. La capa de 128 neuronas se encuentra conectada a la capa oculta anterior de 512 neuronas, y presenta normalización por lotes, conectando a su vez con una capa de salida que presenta tantas neuronas como sean necesarias para representar la etiqueta de clase y función de activación softmax.

Tabla 8.1: Características del discriminador del modelo ACGAN utilizado para generar muestras sintéticas.

Capa	Salida	Norm.	Activación
Discriminador			
Entrada = [x, y]	Tamaño Datos	-	-
Fully connected	64	Sí	Leaky ReLU
Fully connected	128	Sí	Leaky ReLU
Fully connected	256	Sí	Leaky ReLU
Fully connected	512	Sí	Leaky ReLU
Salida Real/Falsa			
Fully connected	1	No	Sigmoide
Salida Etiqueta			
Fully connected	128	Sí	Leaky ReLU
Fully connected	Tamaño Etiqueta	No	Softmax
Discriminador optimizador	Adam (lr = 0.0002, beta1 = 0.5, beta2 = 0.9)		
Leaky ReLU	$\alpha = 0.2$		
Backend	Tensorflow		

Por último, las modificaciones realizadas en el proceso de generación de datos y que dan lugar al modelo adaptativo con clasificador generativo, incluyendo su versión 'Balanced Multiclass' de control del balanceo, tienen la

UNIVERSIDAD DE MÁLAGA

misma implementación que el modelo CGAN estándar con la excepción de la implementación específica del clasificador generativo. El modelo adaptativo que se encuentra desarrollado en el Capítulo 5 sección 5.1, se denomina en los resultados de esta investigación como ModCGAN. La implementación del clasificador generativo utilizado es similar a la que se desarrolló para el estudio de la predicción de eventos en cáncer. Así mismo, esta implementación es también utilizada para el modelo clasificador profundo utilizado en los experimentos para la predicción del subtipo molecular. La Tabla 8.2 muestra las características de la red neuronal utilizada, el número de capas utilizadas y de neuronas presentes en cada capa, el uso de normalización por lotes y regularización mediante dropout, así como la función de activación, optimizador y backend. Se señala además la tasa de abandono utilizada en la regularización mediante dropout en la primera, segunda y tercera capa oculta respectivamente, haciendo una distinción entre los valores utilizados para el Clasificador Generativo y el clasificador utilizado para los experimentos. La adición de una cuarta capa es posible debido al reducido número de instancias en el conjunto, logrando un mayor rendimiento de la red.

Tabla 8.2: Características de la red neuronal utilizada como clasificador.

Capa	Salida	Norm.	Dropout	Activación
Entrada = [x]	Tamaño Datos	-	-	-
Fully connected	1024	Sí	Sí	Leaky ReLU
Fully connected	512	Sí	Sí	Leaky ReLU
Fully connected	256	Sí	Sí	Leaky ReLU
Fully connected	128	Sí	Sí	Leaky ReLU
Fully connected	2	No	No	Softmax
Optimizador	Adam (lr = 0.0002, beta1 = 0.5, beta2 = 0.9)			
Dropout ratio	Clasificador Generativo (0.1, 0.5, 0.3, 0.1)			
Dropout ratio	Clasificador para test (0.3, 0.6, 0.4, 0.1)			
Leaky ReLU	$\alpha = 0.2$			
Regularización	regularización L^2 en pesos (scale = 1e-6)			
Backend	Tensorflow			

Para esta investigación, los modelos generativos principales son probados con 6 niveles diferentes de porcentajes de aumento de datos (50 %, 100 %, 200 %, 500 %, 750 %, 1000 %). En este caso, presentando el conjunto original un total de 165 muestras, estos porcentajes son suficientes como para comprobar el efecto del aumento en la clasificación. En la Tabla 8.3 se muestran el número de muestras generadas y presentes en el conjunto de entrenamiento aumentado, para cada porcentaje de aumento de datos que se aplica con los modelos generativos, incluyendo SMOTE. En este caso, SMOTE presenta una división dependiendo de si el objetivo es la predicción binaria (grado bajo/Luminal, o grado alto/No Luminal) o la predicción multiclase (LA, LBH+, LBH-, Her2 o TN). Debido al funcionamiento de SMOTE, se generan tantas muestras como sean necesarias para equilibrar las clases en

UNIVERSIDAD DE MÁLAGA

Tabla 8.3: Número de muestras generadas y tamaño del conjunto de entrenamiento aumentado, según el porcentaje de DA aplicado. La aplicación de SMOTE siempre crea el mismo número de muestras.

	Porcentaje de DA						SMOTE	
Conjunto	50	100	200	500	750	1000	Binario	Multiclase
Generado	41	82	164	410	615	820	52	67
Aumentado	123	164	246	492	697	902	162	180

el conjunto aumentado. Para el problema binario, se tiene que equilibrar la clase minoritaria, pero en el problema multiclase, SMOTE equilibra las tres clases minoritarias a la clase mayoritaria LBH-. Dentro de estas distinciones, SMOTE siempre genera el mismo número de muestras sin ser controlada por un porcentaje de aumento. En la experimentación los conjuntos de entrenamiento tienen un tamaño medio de 82 muestras en su aplicación con un sistema de vectores de soporte como modelo clasificador y una división del 50 % de las muestras para entrenamiento. Mediante esta experimentación, se generan un máximo de 820 muestras sintéticas con un porcentaje de 1000 %, teniendo en cuenta que mediante el método Noise Bal todas las muestras generadas pertenecen a la clase minoritaria. No se han llevado a cabo todas las versiones modificadas de los modelos generativos profundos, CGAN, ACGAN y ModCGAN, debido a que estos conllevan un tiempo de cómputo más elevado.

8.3. Resultados Experimentales

El proceso de experimentación realizado para la aplicación de DA en el problema de la clasificación del subtipo molecular del cáncer de mama a partir de muestras de expresión génica del sistema inmunitario se puede observar en la Figura 8.1. El conjunto de datos utilizado se encuentra descrito en el Apéndice A.5, explicando también la división en dos subproblemas realizada. De forma resumida, se va a tratar un problema binario realizando una clasificación según el subtipo sea de grado bajo (Luminal) o de grado alto (No Luminal); y un problema multiclase realizando una clasificación en los 5 subtipos estudiados (LA, LBH-, LBH+, Her2 y TN). En la Tabla A.9 del Apéndice A.5 se muestra la distribución del subtipo molecular en el conjunto de datos disponible. Mencionar al respecto que, para reducir el número de variables que influyen en la predicción, se ha realizado una selección de características mediante la importancia Gini del método Random Forest, siendo realizada de forma independiente para cada problema tratado. Los detalles se encuentran descritos en el Apéndice A.5.

Como muestra la Figura 8.1, en ambos problemas tratados, binario y multiclase, a partir del conjunto de datos original se realiza en primer lugar

UNIVERSIDAD DE MÁLAGA

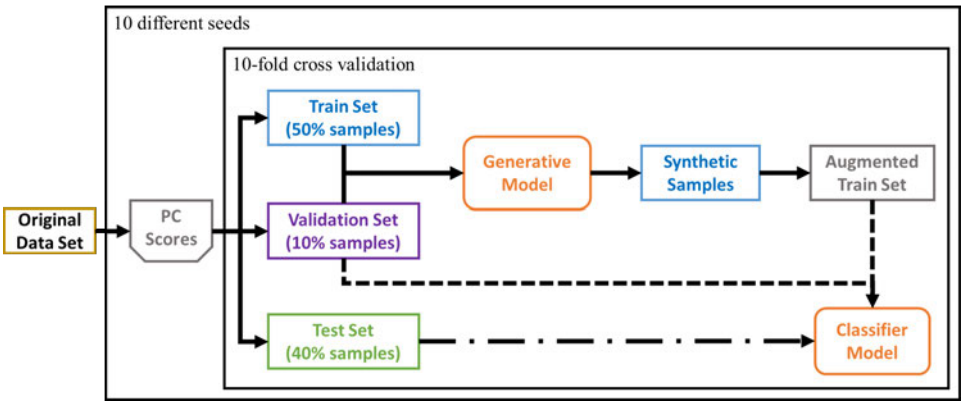

Figura 8.1: Diagrama de flujo del proceso de experimentación. Transformación de variables a componentes principales; división de los datos en conjuntos de entrenamiento, validación y test en un proceso de validación cruzada; aplicación del modelo generativo con el conjunto de entrenamiento y validación, y aumento con las muestras sintéticas; entrenamiento del clasificador y predicción con el conjunto de test. El proceso se repite con 10 semillas.

un análisis de componentes principales (PCA), obteniendo los vectores de puntaciones de las componentes y transformando el conjunto de datos. Esta estrategia de PCA se aplica para cada semilla antes de cualquier experimentación realizada. Se realiza esta transformación debido a que las variables del conjunto de expresión génica solo indican la presencia o no presencia de las variables moleculares codificadas en el complejo mayor de histocompatibilidad. Esto significa que las variables son todas binarias, implicando una mayor dificultad para realizar DA y la predicción del subtipo molecular. En este caso, al haber realizado con anterioridad una selección de características mediante Random Forest, no es necesario considerar los porcentajes de la varianza explicada por cada componente principal con el fin de seleccionar un número suficiente de componentes. Una vez realizada la transformación, se procede con una división estratificada en conjuntos de entrenamiento, validación y test al utilizar una red neuronal profunda como modelo clasificador.

Gracias a la división realizada, se mantiene la independencia total dentro de la experimentación entre el proceso de generación de datos sintéticos y el entrenamiento de los modelos de clasificación y la evaluación de las métricas de predicción. En este contexto, el conjunto de test no se incluye en el proceso de generación de datos y se mantiene por separado hasta que se lleva a cabo la prueba final para obtener una evaluación honesta externa del rendimiento alcanzado. Se establece una división del 50 % de los datos para entrenamiento, 10 % de datos para validación y del 40 % para test cuando se aplica una red neuronal profunda, en vista del reducido número de muestras presentes para algunos subtipos en el conjunto de datos de referencia. Si se utiliza un

UNIVERSIDAD DE MÁLAGA

sistema de máquinas de vectores de soporte como modelo clasificador, solo es necesaria una división del 60 % de las muestras para entrenamiento y 40 % para test. Estas proporciones permiten una mejor evaluación de los resultados de test. Además, todo el proceso de división de los datos y su aplicación se realiza mediante un procedimiento de validación cruzada para proporcionar una mejor estimación de los resultados adquiridos, reduciendo los efectos aleatorios que surgen del proceso. El proceso de generación de datos emplea el conjunto de entrenamiento para crear el número deseado de muestras sintéticas, a excepción de SMOTE que crea una cantidad suficiente de muestras para que el conjunto aumentado se equilibre. Cada modelo de DA aplicado sigue el procedimiento descrito en sus correspondientes capítulos de la presente tesis. Una vez creadas las muestras sintéticas, estas se agregan a los datos de entrenamiento, un proceso que da como resultado un conjunto de entrenamiento aumentado con el que se entrena el modelo clasificador. Una vez que se completa el entrenamiento, se realiza la predicción del conjunto de prueba, obteniendo las métricas necesarias, que luego se promedian para la validación cruzada. Todo el proceso descrito se repite 10 veces con diferentes semillas para aliviar los efectos aleatorios.

8.3.1. Rendimiento en predicción

La Tabla 8.4 muestra los resultados de test obtenidos para los diferentes métodos y modelos DA aplicados, utilizando una red neuronal profunda como modelo clasificador en la experimentación. En esta tabla, “Original” indica los resultados obtenidos con los datos cuando no se aplica ningún método de aumento del número de muestras, sirviendo como referencia y permitiendo observar si los resultados muestran una mejora del rendimiento en clasificación gracias al DA. La segunda columna (‘Porc.’) se refiere al tamaño de los datos sintéticos creados en comparación con el conjunto de entrenamiento original. Estos porcentajes junto con los valores del número de muestras creadas puede ser revisados en la Tabla 8.3 de la sección anterior. Las siguientes columnas muestran los valores obtenidos para cada una de las métricas de test, indicando la fuente en negrita los mejores valores alcanzados. Las métricas utilizadas en la investigación están descritas de forma extendida en el capítulo 3 sección 3.2. En este caso, para medir sensibilidad y especificidad se toma como clase negativa una equivalencia con subtipo de grado bajo (mayoritario) y como positiva una equivalencia a grado alto (minoritario). Como apunte señalar que el modelo adaptativo ModCGAN con la modificación de balanceo de los datos “Balanced Multiclass”, ha sido acortado como Mod_BM para mejorar la presentación de la tabla.

Los resultados presentes en la Tabla 8.4 muestran que mediante la aplicación del método Noise Bal se alcanza el mejor valor de capacidad predictiva en test, representando una mejora en comparación con el conjunto de da-

UNIVERSIDAD DE MÁLAGA

Tabla 8.4: Resultados de test obtenidos para la predicción binaria del subtipo molecular, con los distintos métodos de DA aplicados.

Método	Porc.	Acc	Sens	Espec	F_1	MCC	AUC
Original	None	0.8141	0.3058	0.9270	0.3478	0.2701	0.6042
CGAN	500	0.8288	0.3567	0.9337	0.4133	0.3436	0.5899
ACGAN	1000	0.8303	0.3667	0.9333	0.4202	0.3511	0.6145
ModCGAN	1000	0.8262	0.3342	**0.9356**	0.3241	0.3172	0.6030
Mod_BM	1000	0.8338	0.4008	0.9300	0.4229	0.3800	0.6613
NOISE	100	0.8214	0.3175	0.9333	0.3747	0.2993	0.5796
NOISE Bal	1000	**0.8424**	**0.5242**	0.9131	**0.5331**	**0.4535**	**0.7302**
SMOTE	None	0.8094	0.2717	0.9289	0.3190	0.2444	0.5155

tos sin aumento. Utilizando este método con un porcentaje de aumento de 1000 % se consigue una capacidad predictiva del 0.8424 con una especificidad del 0.9131, siendo este último un valor inferior al obtenido con el conjunto de datos de referencia. Sin embargo, este método destaca especialmente por su capacidad de aumentar la sensibilidad, obteniendo un valor de 0.5242, siendo el modelo que obtiene el mayor valor de sensibilidad. Esto deriva en la obtención de los mayores valores para F_1 (0.5331), MCC (0.4535) y AUC (0.7302). Estos valores muestran una mejora sustancial en comparación con el análisis del conjunto de datos sin aumento, con el que se obtiene un 0.3058 de sensibilidad. Los valores obtenidos con Noise Bal suponen una mejora en diferencia relativa (RD) de 3.5 % en capacidad predictiva y 20.8 % en AUC, y una diferencia porcentual de 18.5 % en F_1 y 18.3 % en MCC, respecto a los valores obtenidos con el conjunto original sin aumento. Sobre los modelos generativos profundos, es necesario mencionar que todos ellos mejoran los valores de capacidad predictiva, sensibilidad y MCC respecto a los datos originales. En esta ocasión, el modelo ModCGAN obtiene el mayor valor de especificidad, sin embargo presenta una sensibilidad menor que el resto de modelos. Por el contrario, el modelo ModCGAN con la modificación de balanceo de los datos, no solo obtiene un valor de capacidad predictiva similar, sino que obtiene los mejores valores de sensibilidad, F_1, MCC y AUC para estos modelos generativos profundos. Estos valores son además los que más se acercan a los valores obtenidos con la estrategia Noise Bal.

Teniendo en cuenta los resultados anteriores, los mejores resultados para la mayoría de los métodos de DA se han obtenido generando el máximo de muestras posibles. Un análisis de la influencia del tamaño del conjunto de datos generado en los resultados obtenidos con diferentes métodos de DA, se muestra en la Figura 8.2. En esta figura se presentan los valores para tres métricas de test (capacidad predictiva, especificidad y sensibilidad) obtenidas mediante la aplicación de cuatro métodos de DA (CGAN, ModCGAN_BM, Noise y Noise Bal), *versus* el número de instancias en una escala logarítmica en el eje de abscisas. El método de adición de ruido estándar es el que

UNIVERSIDAD DE MÁLAGA

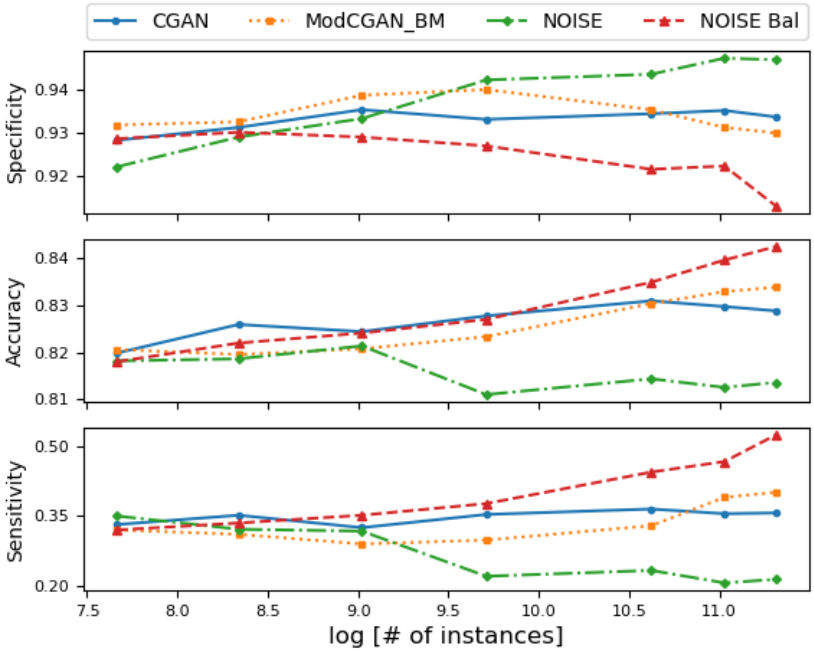

Figura 8.2: Comparación de la capacidad de predicción, especificidad y sensibilidad obtenidas usando diferentes modelos de DA, con respecto al logaritmo del número de instancias creadas. Código líneas: sólida azul, modelo CGAN; puntos naranja, ModCGAN con balanceo; guiones y puntos verde, Noise; guiones roja, Noise Bal.

presenta su mejor resultado de capacidad predictiva con un número menor de muestras generadas (100 %). Se puede observar una correlación positiva significativa entre la especificidad y el número de muestras creadas, sin embargo, la sensibilidad presenta una correlación negativa, lo que conduce a un descenso de la capacidad de predicción cuando el número de muestras creadas es elevado. De forma contraria, la estrategia Noise Bal presenta una significativa correlación positiva para la ganancia de sensibilidad y capacidad predictiva, y una ligera correlación negativa para la ganancia de especificidad. Esto provoca que los mejores resultados de rendimiento sean obtenidos con el máximo porcentaje de creación probado (1000 %).

Respecto a los modelos generativos basados en CGAN, el modelo ModCGAN con balanceo presenta unas correlaciones similares a Noise Bal, con valores más estables para la especificidad. Para este modelo, se puede observar que inicialmente la especificidad presenta una ganancia relacionada con el aumento de muestras, y la sensibilidad una pequeña pérdida. Este efecto se ve intercambiado a partir de un porcentaje de 200 % DA, con una ganancia en la sensibilidad y una pérdida de especificidad. Por último, los resultados para el modelo CGAN indican que los valores de las métricas permanecen estables con independencia del número de muestras generadas.

UNIVERSIDAD DE MÁLAGA

Tabla 8.5: Resultados de test obtenidos para la predicción multiclase del subtipo, con los distintos métodos de DA aplicados.

Método	Porc.	Acc	LA	LBH-	LBH+	Her2	TN
Original	None	0.3224	7.83	11.71	0.44	0.86	0.44
CGAN	200	0.3359	8.24	12.1	0.5	0.83	0.5
ACGAN	750	0.3352	**8.53**	11.96	0.31	0.67	0.65
ModCGAN	500	0.3420	8.44	**12.82**	0.55	0.42	0.34
Mod_BM	500	0.3359	7.96	12.2	0.44	0.85	**0.72**
NOISE	750	**0.3423**	8.3	12.74	0.31	0.86	0.38
NOISE Bal	500	0.3239	7.64	11.55	**0.61**	**1.14**	0.44
SMOTE	100	0.3089	7.6	10.83	0.39	1.09	0.48

Los resultados expuestos anteriormente hacen referencia al enfoque del problema binario. Tratando con el problema multiclase, los resultados obtenidos se encuentran expuestos en la Tabla 8.5. En este caso, para cada método de DA estudiado se muestra el porcentaje aplicado con el que se obtienen los mejores resultados, la capacidad predictiva alcanzada por el modelo y el número de casos correctamente clasificados de cada subtipo molecular (LA, LBH-, LBH+, Her2 y TN). Como se puede observar, el problema multiclase es claramente más complejo que el enfoque binario, obteniendo resultados de capacidad predictiva muy pobres en general. Aun así, los resultados muestran que mediante la aplicación de DA se logra mejorar la capacidad de predicción en test respecto al valor obtenido con el conjunto de datos de referencia. Los resultados muestran que la aplicación de DA mediante el método de adición de ruido, aumenta la capacidad predictiva de la red neuronal hasta un valor de 0.3423, lo que representa una mejora en RD de 6.16 % respecto a no aplicar DA. Sin embargo, se observa que el número de muestras correctamente clasificadas para las clases minoritarias es muy inferior a lo deseable. Este hecho se observa con todos los modelos aplicados, siendo los valores ligeramente superiores para algunas clases mediante los modelos ModCGAN_BM, el método Noise Bal o la aplicación de SMOTE.

Para analizar la distribución de las predicciones realizadas y explicar el bajo rendimiento en el problema multiclase, se muestran en la Figura 8.3 las matrices de confusión normalizadas. Estos resultados son obtenidos con los datos originales (arriba izquierda) y con la aplicación de DA mediante NOISE Bal (arriba derecha), ACGAN (abajo izquierda) y ModCGAN_BM (abajo derecha). Las matrices de confusión indican que la red neuronal se centra en la predicción de las clases mayoritarias, clasificando casi todas las muestras como pertenecientes al subtipo LBH- o LA. Esta tendencia se ve ligeramente alterada por la aplicación de DA, pero no es suficiente. También se puede observar cierta tendencia a clasificar de forma errónea los subtipos LBH+ y TN, especialmente mediante la aplicación de la estrategia Noise Bal. Este obtiene valores de 0.14 y 0.22 para la clasificación de muestras del subtipo LBH- como TN y viceversa.

UNIVERSIDAD DE MÁLAGA

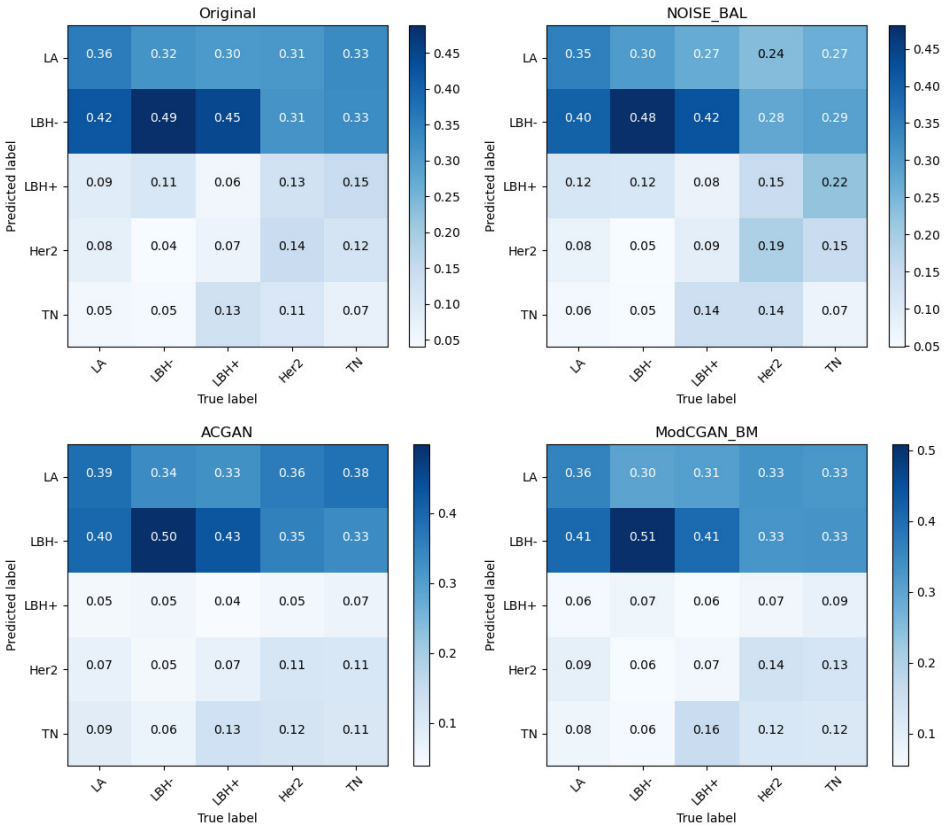

Figura 8.3: Matriz de confusión normalizada obtenida con los datos originales (arriba izquierda) y realizando DA con NOISE Bal (arriba derecha), ACGAN (abajo izquierda) y ModCGAN_BM (abajo derecha).

Los resultados anteriores se obtuvieron utilizando una red neuronal profunda como modelo clasificador. A modo de comparación, también se utilizan un sistema de Ranfom Forest (RF) y un sistema de máquina de vectores de soporte (SVM). Los resultados obtenidos en el enfoque de problema binario para los diferentes métodos y modelos DA aplicados se muestran en la Tabla 8.6. Esta tabla muestra el porcentaje de datos sintéticos creados y los valores de test obtenidos para la capacidad predictiva, sensibilidad, especificidad, F_1, MCC y AUC. Los resultados de la Tabla 8.6 muestran una mejora del rendimiento de la predicción en la mayoría de métricas de evaluación con el método Noise Bal para ambos modelos clasificadores, por encima del rendimiento obtenido con el conjunto de datos sin aumento. Concretamente, para el sistema de RF, el aumento mediante la estrategia Noise Bal con 1000 % de DA alcanza una capacidad predictiva de 0.8433, un valor de F_1 igual a 0.5988, un MCC igual a 0.5100 y un AUC de 0.8107. Estos valores suponen una mejora en diferencia relativa (RD) de 3.8 % en capacidad predictiva y 6.1 % en AUC, y una diferencia porcentual de 23.4 % en F_1 y 23.0 % en MCC,

UNIVERSIDAD DE MÁLAGA

Tabla 8.6: Resultados de test obtenidos para el problema binario, con los distintos métodos de DA utilizando un Random Forest (RF) y un sistema de máquinas de vectores de soporte (SVM) como clasificador.

Método	Porc.	Acc	Sens	Espec	F_1	MCC	AUC
RF							
Original	None	0.8121	0.3083	0.9241	0.3652	0.2800	0.7643
CGAN	750	0.8341	0.4133	0.9276	0.4624	0.3856	0.7930
ACGAN	500	0.8335	0.3917	0.9317	0.4465	0.3740	0.7851
ModCGAN	750	0.8364	0.4117	0.9307	0.4672	0.3945	0.7857
Mod_BM	750	0.8370	0.4500	0.9230	0.4903	0.4125	0.8015
NOISE	50	0.8302	0.3250	**0.9424**	0.3978	0.3362	0.7674
NOISE Bal	1000	**0.8433**	**0.6500**	0.8863	**0.5988**	**0.5100**	**0.8107**
SMOTE	None	0.8380	0.5825	0.8948	0.5597	0.4690	0.8040
SVM							
Original	None	0.8177	0.2358	**0.9470**	0.2723	0.2165	0.7920
CGAN	500	0.8423	0.5092	0.9163	0.5291	0.4485	0.8225
ACGAN	500	0.8452	0.4767	0.9270	0.5125	0.4398	0.8301
ModCGAN	200	0.8427	0.4917	0.9207	0.5204	0.4440	0.8296
Mod_BM	750	0.8488	0.5383	0.9180	0.5535	0.4785	0.8358
NOISE	1000	0.8188	0.2483	0.9456	0.3187	0.2595	0.7592
NOISE Bal	750	**0.8490**	**0.7262**	0.8763	**0.6360**	**0.5512**	**0.8405**
SMOTE	None	0.8321	0.6600	0.8704	0.5877	0.4954	0.7978

respecto a los valores obtenidos con el conjunto original sin aumento. Por otra parte, para el sistema SVM, utilizando el método Noise Bal con 750 %, se logran valores muy similares. Se alcanza un valor de 0.8490 en capacidad predictiva, de 0.6332 en F_1, 0.5512 en MCC y de 0.8405 en AUC. Esto supone una mejora en RD de 3.8 % en capacidad predictiva y 6.1 % en AUC, y una diferencia porcentual de 36.4 % en F_1 y 33.5 % en MCC.

Respecto a los modelos generativos profundos basados en CGAN, se obtienen con ambos clasificadores valores elevados de especificidad y valores medios de sensibilidad. En general, mediante la aplicación de estos modelos se obtienen valores muy similares que indican una mejora del rendimiento del clasificador. Entre estos destaca el modelo ModCGAN con la modificación de balanceo de muestras, el cual logra los mayores valores para las diferentes métricas. De forma respectiva para RF y SVM, la aplicación de este modelo obtiene unos valores de 0.8370 y 0.8489 para la capacidad de predicción, y unos valores de 0.4125 y 0.4785 para MCC. Estos valores se asemejan a los obtenidos con Noise Bal, si bien este último obtiene un mejor valor de sensibilidad y menor de especificidad, derivando en valores superiores para el resto de métricas, principalmente para F_1 y MCC. Destacar finalmente que los valores de capacidad predictiva son similares a los obtenidos mediante una red neuronal, si bien los valores de F_1 y MCC son mayores en RF y SVM, especialmente para este último modelo clasificador.

UNIVERSIDAD DE MÁLAGA

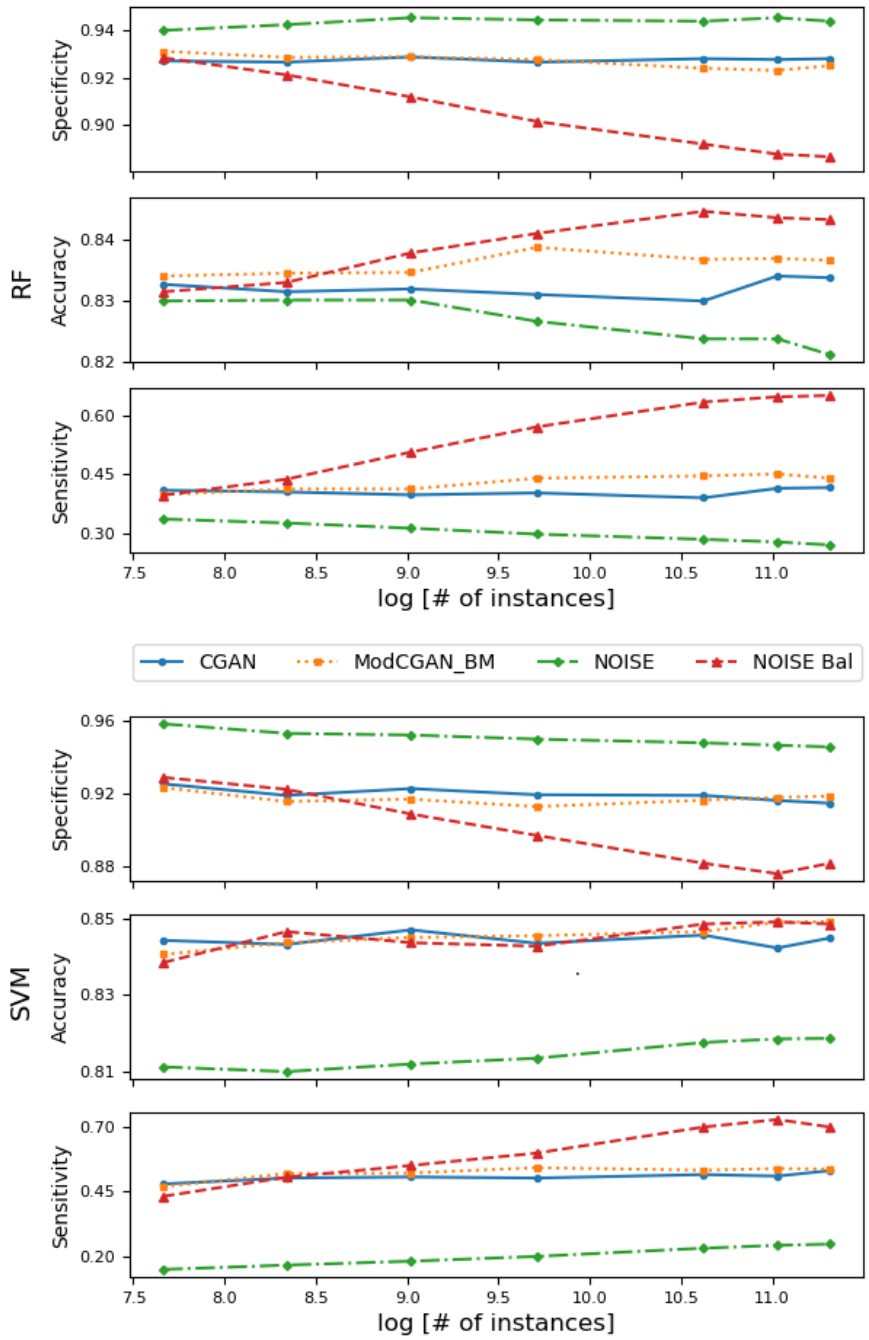

Figura 8.4: Comparación del capacidad de predicción, especificidad y sensibilidad obtenidas usando un sistema de RF y de SVM mediante diferentes métodos de DA, con respecto al logaritmo del número de instancias creadas. Código líneas: igual que la Figura 8.2.

UNIVERSIDAD DE MÁLAGA

La influencia del tamaño del conjunto de datos en los resultados se puede ver en la Figura 7.7. Esta muestra la capacidad predictiva, especificidad y sensibilidad obtenidas con los métodos CGAN, ModCGAN_BM, Noise y Noise Bal. Los análisis se basan en la experimentación con los sistemas de RF y SVM, y se comparan con el número de instancias en una escala logarítmica en el eje de abscisas. Los resultados mostrados son similares a los obtenidos con la red neuronal. Los resultados para CGAN y ModCGAN_BM indican una estabilidad en los valores de las métricas respecto al aumento del número de muestras generadas. Esta es una ligera diferencia, ya que el modelo ModCGAN no presenta una ganancia en sensibilidad tan significativa. También se pueden observar resultados similares para la estrategia Noise Bal, con una significativa correlación negativa para la ganancia de especificidad, y una correlación positiva para la ganancia de sensibilidad y capacidad predictiva. Por último, el método de ruido presenta una ligera ganancia de sensibilidad y capacidad de predicción respecto al número de muestras para el sistema de SVM, mientras que para RF presenta una ligera correlación negativa.

Tabla 8.7: Resultados de test obtenidos para la predicción multiclase, con los distintos métodos de DA utilizando un Random Forest (RF) y un sistema de máquinas de vectores de soporte (SVM) como clasificador.

Método	Porc.	Acc	LA	LBH-	LBH+	Her2	TN
RF							
Original	None	0.3624	8.55	14.47	0.	0.65	0.25
CGAN	500	0.3633	7.96	14.95	0.07	0.71	0.29
ACGAN	100	**0.3748**	**14.25**	9.79	0.03	0.52	0.15
ModCGAN	500	0.3723	13.32	10.73	0.	0.52	0.
Mod_BM	100	0.3583	7.57	**14.73**	0.15	0.85	0.35
NOISE	200	0.3676	8.26	15.07	0.	0.68	0.25
NOISE Bal	100	0.3638	13.91	9.05	0.01	0.68	0.36
SMOTE	None	0.3295	6.21	13.48	**0.16**	**1.2**	**0.7**
SVM							
Original	None	0.3455	6.77	15.84	0.02	0.01	0.16
CGAN	500	0.3829	9.61	14.68	0.2	0.28	0.5
ACGAN	100	0.3735	**14.26**	9.77	0.19	0.05	0.38
ModCGAN	500	**0.3911**	12.78	12.66	0.1	0.04	0.23
Mod_BM	100	0.3818	8.12	**15.37**	**0.42**	0.58	0.71
NOISE	50	0.3761	10.	13.56	0.2	0.45	0.61
NOISE Bal	200	0.3605	12.25	9.46	0.38	**0.97**	**0.73**
SMOTE	None	0.2889	3.82	14.25	0.03	0.95	0.02

También se muestran los resultados obtenidos con RF y el sistema SVM como clasificadores, para el enfoque del problema multiclase. En la Tabla 8.7 se muestran estos resultados, similares a los obtenidos con la red neuronal. En esta tabla se observa para cada método de DA el porcentaje aplicado, la capacidad predictiva ('*Acc*') alcanzada y el número de casos correctamen-

UNIVERSIDAD DE MÁLAGA

te clasificados de cada subtipo molecular. Los resultados muestran que la aplicación de DA impulsa al modelo a obtener mejores valores de capacidad predictiva en test respecto al valor obtenido con el conjunto de datos de referencia. Concretamente, ACGAN alcanza una capacidad predictiva del 0.3748, representando una mejora en RD del para RF, y el modelo ModCGAN alcanza un valor de 0.3911, representando una mejora en RD del 13.2 % para SVM. Si bien, en ambos casos se observa que solo clasifican correctamente muestras para las clases mayoritarias (LA y LBH-). A este respecto el modelo ModCGAN con la modificación BM (Mod_BM), el método Noise Bal o la aplicación de SMOTE (este último en el caso exclusivo de RF), son modelos más recomendables pues logran, aunque de forma muy escasa, clasificar una mayor cantidad de muestras correctamente para las clases minoritarias. Sin embargo, los resultados siguen siendo muy pobres.

Las matrices de confusión normalizadas obtenidas con los datos originales (arriba izquierda), NOISE Bal (arriba derecha), ACGAN (abajo izquierda) y ModCGAN_BM (abajo derecha), y utilizando un sistema de RF y de

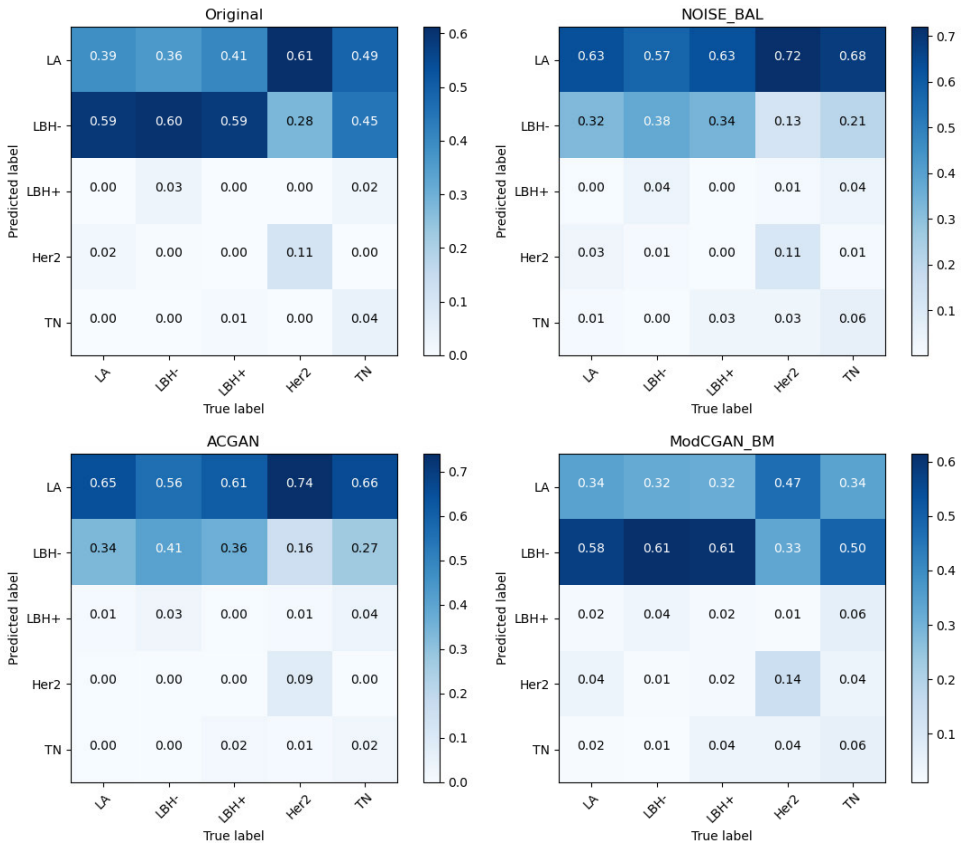

Figura 8.5: Matriz de confusión normalizada obtenida con diversos enfoques de DA. Se utiliza un sistema de RF como clasificador.

UNIVERSIDAD DE MÁLAGA

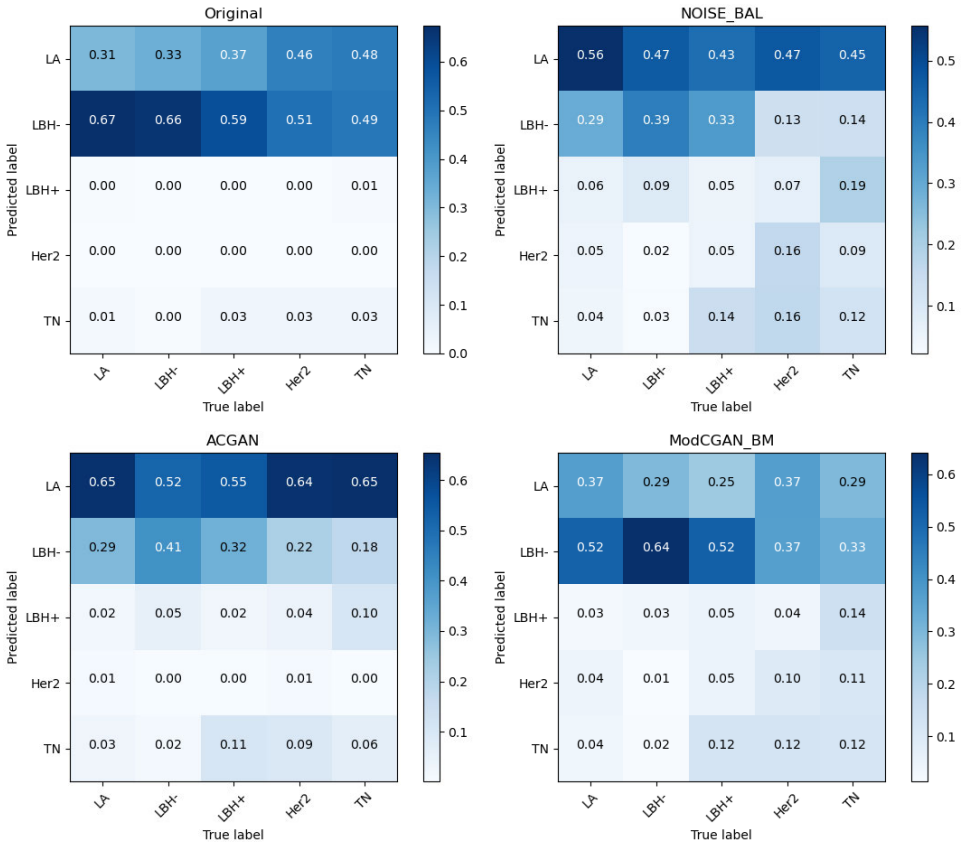

Figura 8.6: Matriz de confusión normalizada obtenida con diversos enfoques de DA. Se utiliza un sistema de SVM como clasificador.

SVM se muestran en las Figuras 8.5 y 8.6, respectivamente. Los resultados indican que el uso de los modelos RF o SVM no afecta mayormente a la clasificación. Al igual que la clasificación mediante la red neuronal profunda, los modelos se centran en devolver predicciones para las clases mayoritarias. El aumento del número de muestras de entrenamiento mediante los métodos Noise Bal y ModCGAN_BM reduce de forma muy leve este hecho, con un efecto ligeramente superior mediante la clasificación con SVM. Remarcar de estos resultados que el aumento con muestras generadas con Noise Bal provoca que el modelo clasificador esté sesgado hacia el subtipo LA, mientras que el aumento con ModCGAN_BM sesga al modelo hacia el subtipo LBH-.

8.3.2. Análisis del conjunto generado

Una vez analizado el rendimiento alcanzado mediante la aplicación de DA en la experimentación, es necesario analizar la calidad de la información sintética generada. Debido a que los resultados en el enfoque multiclase

UNIVERSIDAD DE MÁLAGA

del problema han sido considerablemente deficientes, los siguientes análisis solo han sido realizados para el problema binario (predicción del subtipo en "Luminal" y "No luminal"). En primer lugar, se llevó a cabo un análisis de componentes principales (PCA) para visualizar la configuración de las muestras en un espacio bidimensional (componente principal 1 *vs* componente principal 2). Mediante este proceso, se puede comparar la distribución de las muestras sintéticas con la distribución de las muestras originales, analizar el funcionamiento de los métodos de DA, y explicar el rendimiento alcanzado en clasificación.

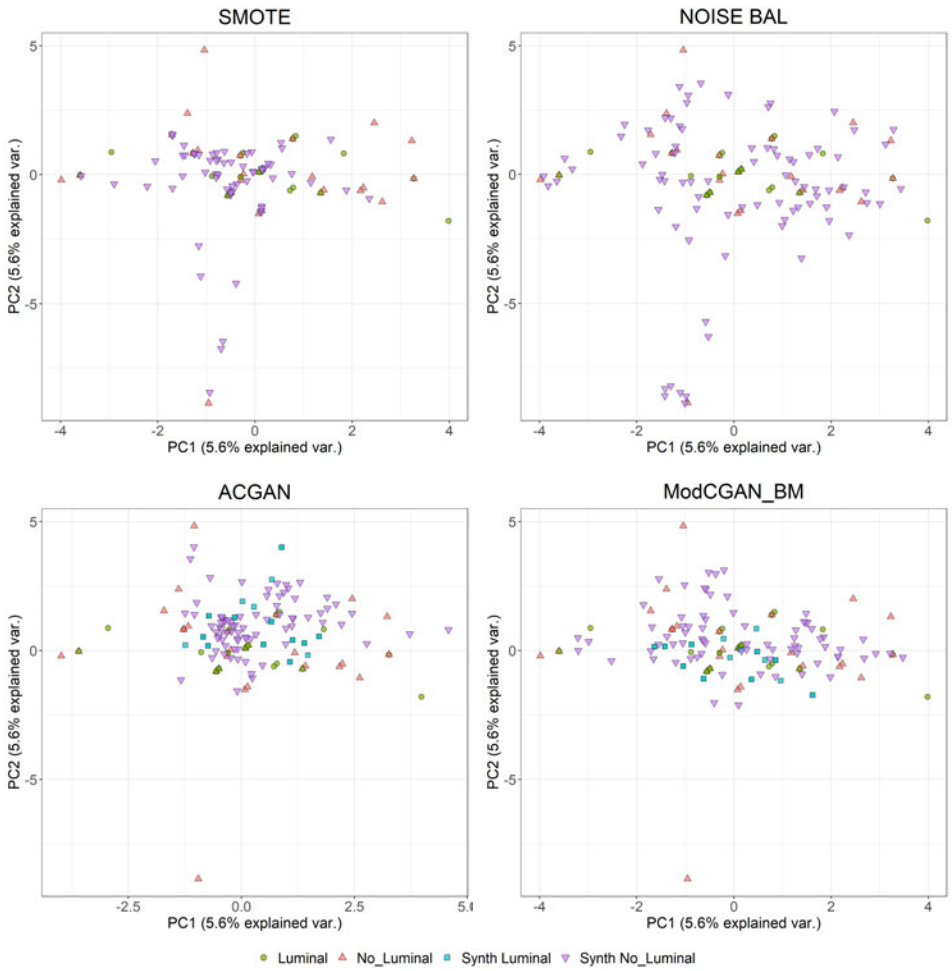

Figura 8.7: Gráfico de PCA con las muestras originales y las muestras generadas con los métodos SMOTE, Noise Bal, ACGAN y ModCGAN_BM para la predicción binaria del subtipo. Códigos de color: círculo verde, luminal; triángulo rojo, no luminal; cuadrado azul, luminal sintético; triángulo invertido morado, no luminal sintético.

UNIVERSIDAD DE MÁLAGA

La Figura 8.7 muestra la distribución de las muestras sintéticas creadas usando la técnica SMOTE, el método Noise Bal, y los modelos ACGAN y ModCGAN con la modificación BM. Para no sobrecargar los gráficos, se presenta una selección de muestras generadas en igual número que las muestras originales. Al igual que en anteriores estudios de aplicación en otros problemas bioinformáticos, estos análisis reflejan muy bien el funcionamiento interno de cada método y modelo de DA en la generación de muestras. Los métodos SMOTE y Noise Bal generan muestras únicamente para la clase minoritaria (No Luminal) de forma que, las muestras generadas por SMOTE se derivan de una interpolación, con lo que estas se encuentran dentro de la distribución de muestras reales; y las muestras generadas por Noise Bal se encuentran alrededor de las muestras originales modificadas con ruido, si bien al ser generados partiendo directamente de unos datos transformados en componentes principales, el ruido genera en este caso más variabilidad, es decir, algunas muestras se encuentran más alejadas de las muestras seleccionadas para modificar.

En cuanto a los modelos generativos profundos, se observa como ACGAN y ModCGAN_BM han generado en ambos casos más muestras de la clase minoritaria, con ciertas diferencias en la dispersión de las muestras. La mayoría de muestras generadas mediante ACGAN se encuentran situadas de forma concéntrica respecto a un núcleo, mientras que las generadas por el método adaptativo se ajustan mejor a la distribución real de las muestras. Es importante notar que, si hay muestras presentes en el conjunto de entrenamiento que pueden ser consideradas como *outliers*, tal y como ocurre en este caso con la muestra de la clase "No luminal" situada en la parte inferior del gráfico, los métodos SMOTE y Noise Bal generan muestras alrededor de estas, teniéndolas en cuenta en su aplicación. Sin embargo, los modelos generativos profundos, debido al funcionamiento discriminatorio interno, pueden considerar estos *outliers* como datos creados por el generador, lo que impide al modelo crear muestras cercanas, o que simplemente se encuentren influenciadas por estas muestras alejadas de la distribución principal.

Adicional a la inspección visual de los resultados de PCA, para analizar las muestras generadas se emplean las métricas *Fréchet Inception Distance* (FID) (Heusel et al., 2017), *Standard Distance Deviation* (SDD) y el desplazamiento del centro medio de la distribución (*Shift*). En primer lugar, la métrica FID se utiliza de forma habitual para medir la calidad de las imágenes generadas por los modelos generativos profundos, pero puede ser aplicada para evaluar el rendimiento de estos modelos en cualquier aplicación de generación de datos. Esta métrica realiza una comparación entre la distribución de los datos generados y la distribución de los datos reales a partir de las magnitudes del vector real (μ_r) y generado (μ_g), la traza de la matriz (Tr) y la matriz de covarianza de los vectores (Σ_r, Σ_g). La ecuación 8.1 muestra el cálculo de FID. Con el propósito de efectuar un análisis más preciso, se

UNIVERSIDAD DE MÁLAGA

obtiene el valor FID resultante de la comparación de la distribución sintética de la clase 0 con la distribución real de la clase 0, y de realizar la comparación de la distribución sintética de la clase 1 con la distribución real de la clase 1. Está métrica indica una mayor similitud entre las distribuciones con valores más bajos, y una menor similitud con valores altos.

$$\text{FID} = \|\mu_r - \mu_g\|^2 + \text{Tr}(\Sigma_r + \Sigma_g - 2(\Sigma_r \Sigma_g)^{1/2}) \tag{8.1}$$

Otra medida estadística empleada es la *Standard Distance Deviation* (SDD), que indica la distancia promedio de todas las muestras desde el centro medio y se usa para analizar la dispersión de una distribución. Esta métrica es similar a la desviación estándar, la cual mide la distribución de los valores de los datos alrededor del valor medio de la estadística. Dado que los conjuntos son transformados siguiendo un procedimiento PCA antes de generar los datos, se utilizan los valores presentes en las muestras para la componente principal 1 y la componente principal 2 para realizar un cálculo del SDD que abarque todas las variables presentes y conlleve un tiempo de ejecución poco costoso. Por último, el valor de desplazamiento del centro medio de la distribución (*Shift*) se calcula mediante la distancia euclidiana entre el centro de la distribución sintética y el centro de la distribución real teniendo en cuenta la componente principal 1 *versus* componente principal 2, de igual forma que en el cálculo de la métrica SDD.

Tabla 8.8: Fréchet Inception Distance ($\widehat{FID}$), Standard Distance Deviation (SDD) y desplazamiento del centro medio de la distribución (*Shift*) obtenidos con las muestras generadas por cada modelo de DA con respecto a las distribuciones reales de las clases binarias.

	Clase 0			Clase 1		
Método	FID	SDD	Shift	FID	SDD	Shift
None	-	0.88	-	-	2.85	-
CGAN	25.66	1.02	1.50	85.24	1.28	2.81
ACGAN	57.87	1.65	1.56	54.34	1.30	0.88
ModCGAN	71.00	2.74	1.34	86.20	2.27	0.98
ModCGAN_BM	38.20	1.23	0.43	46.57	2.04	0.50
Noise	13.88	1.03	0.03	39.08	2.64	0.50
Noise Bal	0.00	0.00	0.00	13.78	3.47	0.58
SMOTE	0.00	0.00	0.00	23.31	2.18	0.68

La Tabla 8.8 muestra los valores obtenidos de cada métrica analizada (FID, SDD y *Shift*) para los diferentes modelos de DA estudiados, y haciendo una distinción entre las métricas para las distribuciones de la clase 0 ("Luminal") y de la clase 1 ("No luminal"). También se muestra el SDD obtenido para la distribución original de muestras ("None") pudiendo compararla mejor con los conjuntos sintéticos. Centrándonos en primer lugar en la clase 0, que es la clase mayoritaria del conjunto de datos, se observa que Noise

UNIVERSIDAD DE MÁLAGA

Bal y SMOTE presentan todos los valores de las métricas como 0, pues solo crean muestras de la clase minoritaria. Para la adición de ruido, los resultados indican una mayor similitud con respecto a la distribución original, menor valor de FID (13.08), valor de SDD similar al original (1.03) y el menor valor de desplazamiento (0.03). El modelo CGAN obtiene unos resultados similares, pero el desplazamiento del centro es mayor (1.50), como ocurre con los métodos ACGAN y ModCGAN, si bien estos reflejan una similitud menor, especialmente ModCGAN obtiene el mayor valor de FID (71.00) y de SDD (2.74). Finalmente, los resultados del conjunto generado por el modelo ModCGAN con la modificación de balanceo "Balanced Multiclass", indican una similitud media en todas las métricas.

Si nos centramos en la clase 1, la cual tiene un mayor interés, se observa en primer lugar que el modelo que ha generado datos con una mayor similitud es Noise Bal, que obtiene el valor FID más bajo (13.78), si bien la dispersión de las muestras generadas es muy superior al original (3.47 frente a 2.85). Los métodos SMOTE y Noise estándar obtienen también valores bajos de FID, indicando una buena similitud, y valores de SDD similares al original, con un bajo desplazamiento del centro. Los datos creados mediante los modelos generativos profundos presentan una menor similitud, obteniendo CGAN y ModCGAN los mayores valores FID (> 0.85), y CGAN especialmente el menor valor de SDD y un desplazamiento muy alto. Por último, los modelos ACGAN y ModCGAN_BM presentan una similitud media, siendo más similar el conjunto generado por ModCGAN_BM, con un valor de 46.57 de FID, 2.04 de SDD y 0.50 de desplazamiento.

Si este análisis se toma en conjunto con los resultados de PCA, y el rendimiento observado en predicción, se puede observar que los modelos generativos profundos crean conjuntos de muestras que presentan una menor similitud general respecto al conjunto original, salvo el modelo ModCGAN_BM, siendo el conjunto aumentado mediante este último el que mejores resultados obtiene en predicción para estos modelos. Por otro lado, el método Noise genera muestras con una similitud mayor, pero la poca variabilidad que añade al conjunto original puede reducir la mejora en clasificación. Un hecho similar parece ocurrir con SMOTE, si bien el efecto del sobremuestreo hace su efecto aumentando la sensibilidad. Por último, el método Noise Bal genera el conjunto más similar, pero con una dispersión mayor, obteniendo los mejores valores de predicción. El hecho de que los modelos generativos no hayan otorgado en este caso una mayor variabilidad a las muestras generadas de la clase minoritaria, obteniendo los menores valores de SDD, puede deberse al hecho observado en el análisis visual de las gráficas PCA, donde se observaban ciertas muestras que el discriminador puede tomar como *outliers*, lo que implica la no generación de muestras para esta clase más allá de la distribución principal.

UNIVERSIDAD DE MÁLAGA

8.4. Conclusiones

En esta investigación se ha propuesto la aplicación de diferentes técnicas de aumento de datos con el objetivo de mejorar el rendimiento de predicción del subtipo molecular del cáncer de mama cuando se analiza un conjunto de datos con perfiles de expresión génica del sistema inmunitario de las pacientes. Se han tratado dos problemas a la hora de la predicción del subtipo, un problema binario realizando una clasificación según el subtipo sea de grado bajo (Luminal) o de grado alto (No Luminal); y un problema multiclase realizando una clasificación en los 5 subtipos presentes en los datos (LA, LBH-, LBH+, Her2 y TN). En cuanto a los métodos de DA estudiados, se ha comparado la aplicación de SMOTE, de métodos de inyección de ruido estándar y con sobremuestreo, y modelos generativos profundos como CGAN, ACGAN y el modelo adaptativo diseñado en el Capítulo 5, incluyendo la modificación "Balanced Multiclass" en la generación de datos sintéticos.

Los resultados reflejan un hecho ya previsto, la mayor dificultad en la predicción multiclase del subtipo respecto al enfoque binario (Ha et al., 2019; Ma et al., 2019). Independientemente del modelo clasificador o la técnica de DA aplicada, las diferencias son amplias, obteniendo para el enfoque binario un porcentaje de aciertos máximo de 84.2 %, frente a un escaso 34.2 % en el enfoque multiclase. El análisis indica que la aplicación del método de adición de ruido con el propósito de realizar sobremuestreo (Noise Bal) puede llevar a un aumento en el rendimiento de la predicción de aproximadamente un 3.4 % de RD en capacidad predictiva para el enfoque binario y del 6.2 % en RD para el problema multiclase. Esta mejora en la capacidad de predicción es suficiente para ambos enfoques, binario y multiclase. Sin embargo, dado que la predicción de la clase minoritaria en el enfoque binario posee implícitamente un mayor valor, pues se identifica con los subtipos de grado alto (No Luminal), más agresivos y cuyo diagnóstico tiene una mayor importancia, las métricas de predicción más representativas son la puntuación F_1 y el MCC. A este respecto, mediante la aplicación del método Noise Bal en el problema binario, se obtiene una mejora en diferencia porcentual del 18.5 % en F_1 y del 18.3 % en MCC, respecto a los valores de referencia.

Estos resultados, obtenidos con una red neuronal profunda como modelo clasificador, se ven replicados mediante el uso de un sistema de Random Forest o SVM. Con estos modelos clasificadores se obtienen valores mayores mediante la aplicación de DA para todas las métricas, especialmente la sensibilidad. Con un sistema SVM y la aplicación del método Noise Bal en el problema binario, se obtiene un porcentaje de aciertos de 84.9 %, un valor de F_1 de 0.64 y un 0.55 para MCC, representando unas mejoras del 3.8 %, 36.4 % y 33.5 % respectivamente. En el problema multiclase se alcanza un valor máximo de porcentaje de aciertos de 37.5 %, siendo un resultado pobre. Los análisis realizados muestran que la aplicación de DA mejora la

UNIVERSIDAD DE MÁLAGA

clasificación de los subtipos minoritarios, si bien en un grado mucho menor a lo deseable. En los experimentos realizados, la mayoría de las muestras se predicen como pertenecientes a las clases mayoritarias LBH- y LA.

Adicionalmente al análisis del rendimiento en predicción, se ha empleado el análisis de componentes principales (PCA) y diferentes métricas estadísticas para analizar las características y calidad de las muestras generadas. Los resultados adquiridos de los análisis apoyan las conclusiones obtenidas con otros problemas bioinformáticos. Teniendo en cuenta que en el conjunto original hay presentes muestras que podrían ser consideradas *outliers*, los análisis gráficos indican que mediante la aplicación de modelos basados en CGAN, las muestras generadas se ajustan a la distribución de los datos reales, siendo un ajuste mejor el realizado mediante ModCGAN con modificación del balanceo. Sin embargo, con estos modelos no se generan muestras que se acerquen a los *outliers*, pues el discriminador toma estas como falsas. Mientras, la estrategia Noise Bal sí genera muestras cercanas, siempre y cuando estas pertenezcan al conjunto de entrenamiento, lo que explica su mejor rendimiento en predicción. Estas conclusiones se apoyan en las métricas de similitud y dispersión, obteniendo los métodos de adición de ruido y SMOTE los mejores valores de FID, y valores de dispersión cercanos al original. En este caso, el valor de dispersión se encuentra influenciado por los *outliers*, por lo que los modelos generativos profundos obtienen valores bajos, más alejados del original. El único modelo basado CGAN que obtiene buenos valores para estas métricas es ModCGAN_BM, mostrando una buena similitud, una dispersión aceptable y un desplazamiento bajo.

En conclusión, la predicción del subtipo molecular del cáncer de mama mediante el análisis de perfiles de expresión génica, es un problema extremadamente complejo. Si se simplifica a la predicción binaria del subtipo de grado bajo o alto, los resultados obtenidos son mejores. Igualmente, para ambos enfoques la aplicación de técnicas de DA implica un aumento del rendimiento de la predicción. Para el problema binario, el método Noise Bal ha demostrado el mejor rendimiento, conduciendo a un aumento de la capacidad de predicción del 3.6 %, con especial atención al aumento de la sensibilidad, lo que deriva en una mejora del 18-36 % en F_1, del 18-33 % en MCC y del 6-20 % en AUC. Debido al aumento en sensibilidad observado gracias al DA para este problema, y en los problemas anteriores, se presenta interesante el desarrollo de un método capaz de aprovechar esta ventaja del DA, junto con la mayor especificidad alcanzada mediante el conjunto sin aumentar. Este estudio será presentado en el siguiente capítulo.

UNIVERSIDAD DE MÁLAGA

UNIVERSIDAD
DE MÁLAGA

Parte IV

Método Meta Clasificador

UNIVERSIDAD
DE MÁLAGA

UNIVERSIDAD
DE MÁLAGA

Capítulo 9

Meta-Clasificador para Conjuntos Desbalanceados

Resumen: Los datos categóricos obtenidos de los dominios del mundo real suelen estar desbalanceados, ya que presentan más muestras de una de las clases que del resto de las clases. Con respecto a las predicciones hechas por los clasificadores, esto tiende a ser un problema ya que la mayoría de ellos se enfocan en muestras de la clase mayoritaria. Entre las técnicas utilizadas para lidiar con conjuntos de datos desbalanceados, el DA constituye una alternativa, ya que puede mejorar la predicción de la clase minoritaria, pero también a costa de una pérdida con respecto a las predicciones de la clase mayoritaria. Para beneficiarnos de ambos comportamientos, se presenta en este capítulo un esquema de meta-clasificador que funciona como una mezcla de dos clasificadores, uno entrenado con los datos originales y el otro con DA mediante un procedimiento de filtrado automático de muestras.

El desequilibrio de clases en los conjuntos de datos es un problema común en la aplicación de algoritmos de aprendizaje automático a muchos problemas del mundo real, ya que en la mayoría de los casos las categorías de datos no están representadas por igual, algunas clases tienen más instancias que otras y, en particular, las categorías relevantes están sub-representadas. Varias aplicaciones presentan una distribución desbalanceada, como la detección de derrames de petróleo en imágenes de radar satelital (Kubat et al., 1998), detección de fraude con tarjetas de crédito (He y Garcia, 2009), aprendizaje de pronunciaciones de palabras (Van Den Bosch et al., 1997), clasificación de texto (Zheng et al., 2004), diagnóstico médico de enfermedades raras (Han et al., 2019), entre otras. Cuando se aplican algoritmos de aprendizaje automático en una tarea supervisada a estos conjuntos de datos, los algoritmos de clasificación se ven afectados por el desequilibrio de clases (Japkowicz y Stephen, 2002; Kubat et al., 1997). La mayoría de los algo-

UNIVERSIDAD DE MÁLAGA

ritmos predictivos se enfocan en mejorar la capacidad de clasificación, sin embargo, alcanzar una alta capacidad predictiva en un problema con un alto porcentaje de muestras mayoritarias es trivial; el clasificador solo debe predecir que todas las muestras pertenecen a la clase mayoritaria. Esto implica que las muestras de la clase minoritaria no sean predichas correctamente, lo que da como resultado un rendimiento deficiente del clasificador y un riesgo potencial según el dominio de la aplicación. En general, las muestras de la clase minoritaria de un conjunto de datos desbalanceado tienden a contener más información que es útil y su clasificación errónea tiene un costo asociado más alto, por ejemplo, el diagnóstico médico de enfermedades raras.

9.1. Enfoques para el desequilibrio de clases

Las técnicas propuestas por los investigadores para lidiar con conjuntos de datos desbalanceados se pueden dividir en tres categorías: métodos a nivel de algoritmo, métodos sensibles al costo y métodos a nivel de datos:

- Los métodos de nivel de algoritmo modifican los algoritmos de clasificación (Galar et al., 2011, 2016; Liu et al., 2008). Las posibles modificaciones incluyen la alteración del umbral de predicción o el entrenamiento de varios clasificadores cuyas salidas se combinan, razón por la cual la categoría también se llama aprendizaje por conjuntos de clasificadores (*ensemble learning*). Los modelos ensemble se han convertido en un método popular para solucionar problemas de desequilibrio de clases, pues se ha comprobado que el conjunto de clasificadores mejora el rendimiento de un solo clasificador y de todos los clasificadores de forma independiente (López et al., 2013). Estos métodos son especialmente útiles en predicción de clases múltiples, ya que la clasificación suele tener un rendimiento significativamente más bajo que en casos binarios. Para este caso los enfoques de uno contra uno (OVO) y de uno contra todos (OVA) son los más utilizados.

- Los métodos sensibles al costo utilizan diferentes costos de clasificación errónea para cada clase en los clasificadores (Fan et al., 1999; Galar et al., 2011). Estos pueden ser introducidos tanto a nivel de datos (remuestreo y selección de características) como a nivel de algoritmos. La mayor desventaja de este enfoque es la necesidad de un conocimiento profundo del problema para asignar los costos en la aplicación de los métodos. Como alternativa más simple para utilizar este enfoque, se puede establecer el costo de clasificación errónea de la clase mayoritaria en 1, y establecer el valor de penalización de la clase minoritaria igual a la ratio de desbalanceo (IR) (Castro y Braga, 2013).

- Los métodos de nivel de datos manipulan los datos de entrenamiento

UNIVERSIDAD DE MÁLAGA

con el fin de balancear las clases. Estas técnicas son las más populares y se pueden clasificar en técnicas de submuestreo, que eliminan muestras de la clase mayoritaria (Drummond et al., 2003; Hart, 1968; Tomek, 1976), y técnicas de sobremuestreo, que generan muestras para la clase minoritaria (Chawla et al., 2002; Han et al., 2005; Douzas et al., 2018). Los métodos que pertenecen a esta última categoría también se incluyen dentro de los métodos de DA, ya que se produce la creación de muestras sintéticas adicionales. Las técnicas de sobremuestreo se analizaron en mayor profundidad en el Capítulo 2 sección 2.1.3.

9.2. Trabajos relacionados

Hasta donde sabemos, esta investigación es la primera que propone un esquema meta-clasificador para aprovechar el comportamiento de los modelos cuando se entrenan con datos no balanceados y cuando se aplica el DA. La forma más popular de resolver problemas desbalanceados mediante el aprendizaje por conjunto de clasificadores, es a través del enfoque boosting (Kearns y Valiant, 1994; Schapire, 1990), donde se entrenan varios clasificadores independientes con el mismo conjunto de datos o con subconjuntos remuestreados, y las salidas de todos los clasificadores en la misma prueba se combinan utilizando algún método de votación simple o ponderado. Fan et al. (1999) diseña una variante de AdaBoost utilizando el costo de las clasificaciones erróneas para actualizar la distribución de entrenamiento en iteraciones de boosting sucesivas y reducir el costo acumulado de clasificación errónea. Domingos (1999), en un proceso similar, modifica un modelo de bagging para realizar el entrenamiento con clasificadores sensibles al costo.

También hay investigaciones desarrolladas con modelos ensemble más innovadores. Lin et al. (2009) divide el grupo mayoritario de un problema binario basándose en las características transformadas del "producto interno", para retener la relación geométrica entre los dos grupos, y con cada división entrena diferentes clasificadores de los que obtiene las probabilidades de pertenencia a las clases y aplica una regresión logística sobre estas probabilidades. Zhang et al. (2018) propone un enfoque basado en stacking y submuestreo aleatorio inverso. El submuestreo aleatorio inverso se usa para generar una gran cantidad de subconjuntos de entrenamiento diferentes con las muestras de la clase mayoritaria y usa un modelo de stacking para separar la clase minoritaria de la mayoritaria, donde el resultado producido por cada clasificador se toma como una característica para entrenar un meta-clasificador. Por último, destacamos el trabajo de Hou et al. (2020), que aplica una estrategia de selección de conjuntos dinámica (DES) con un mecanismo de ponderación al evaluar múltiples clasificadores, y un marco de meta aprendizaje (META-DES) para múltiples criterios, en un problema de evaluación de riesgo crediticio.

UNIVERSIDAD DE MÁLAGA

9.3. Modelo Meta-Clasificador

Como ha sido analizado a lo largo del presente trabajo de tesis, cuando se usa DA para entrenar un modelo clasificador en una tarea supervisada, la capacidad predictiva obtenida en las muestras de test generalmente presenta una ligera mejora en comparación con el uso del conjunto original sin aumento. Sin embargo, la sensibilidad o *recall* obtenida se puede ver incrementada de forma considerable. Este efecto aumenta en la aplicación de DA con conjuntos de datos desbalanceados. Una manera fácil de obtener una alta capacidad predictiva con estos conjuntos de datos es predecir que todas las muestras pertenecen a la clase mayoritaria. Obviamente, esto implica que las muestras de la clase minoritaria no se clasificarán correctamente. Este hecho no solo supone un bajo rendimiento para el clasificador, sino que es un riesgo potencial dependiendo del problema de la aplicación. Sin embargo, cuando un clasificador se ocupa de tales conjuntos de desequilibrios, no hay suficientes muestras de la clase minoritaria para que el clasificador las tenga en cuenta. La mayoría de las técnicas actuales de DA aplicadas a este tipo de problemas, se centran en el sobremuestreo, como SMOTE o ADASYN. El conjunto de datos de entrenamiento se balacea mediante la creación de muestras que solo pertenecen a la clase minoritaria. Con estas técnicas, los modelos clasificadores pueden mejorar en gran medida la capacidad de predecir correctamente las muestras de clases minoritarias. Sin embargo, esto generalmente también conduce a una gran pérdida de capacidad de predicción de la clase mayoritaria y a una reducción de la capacidad predictiva general. El clasificador entrenado en DA puede mejorar su capacidad para identificar las muestras que pertenecen a la clase minoritaria a expensas de las muestras de la clase mayoritaria, lo que reduce la capacidad de predicción.

Se ha diseñado un esquema que utiliza un meta-clasificador con el propósito de mejorar la capacidad de predicción para muestras minoritarias, pero disminuir la pérdida para muestras mayoritarias (Moreno-Barea et al., 2022d). El esquema propuesto se muestra en la Figura 9.1, además del proceso de experimentación seguido. Este esquema hace uso de dos clasificadores, uno entrenado con el conjunto original y otro entrenado con una aplicación DA, además de un proceso de filtrado para seleccionar las muestras de test. El proceso seguido en el esquema, comienza con la división estratificada del conjunto desbalanceado en conjuntos de entrenamiento y de test en un proceso de validación cruzada de 10-folds. Un primer modelo clasificador es entrenado únicamente con los datos originales, y una vez entrenado se obtienen las probabilidades de salida para la predicción con el conjunto de test. De esta manera, se espera que el modelo clasificador sufra por el desequilibrio de los datos del modo anteriormente mencionado, realizando una predicción cercana a clasificar la mayoría de las muestras de test como pertenecientes a la clase mayoritaria. Obviamente, debido a que la mayoría de las muestras de

UNIVERSIDAD DE MÁLAGA

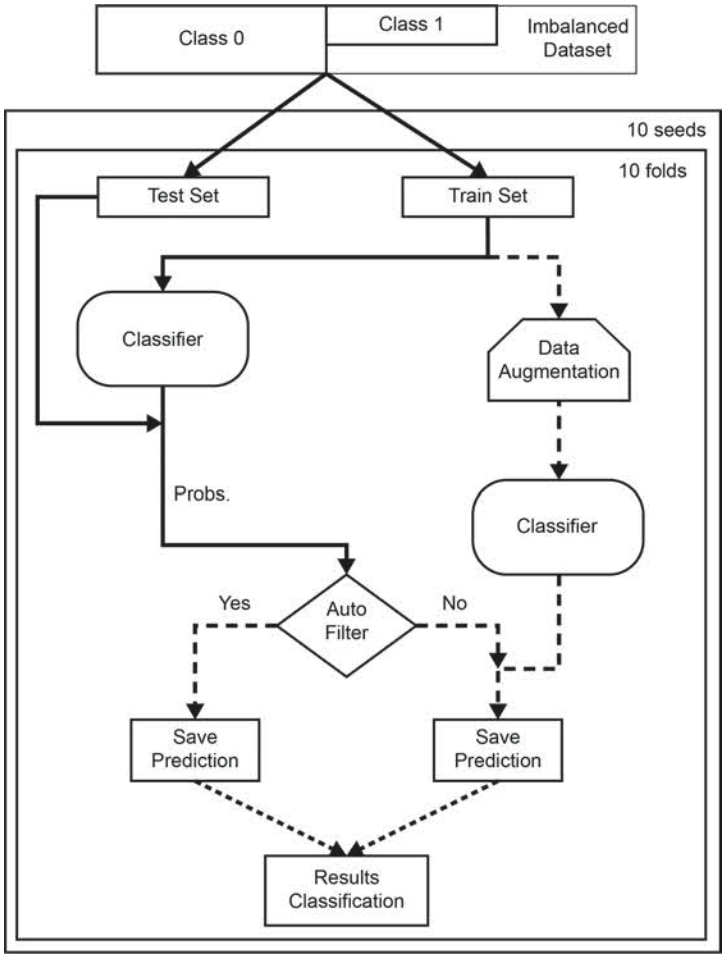

Figura 9.1: Diagrama de flujo de todo el proceso de experimentación con el esquema meta-clasificador para el aumento de datos.

test pertenecen a esta clase, una gran cantidad de muestras serán clasificadas correctamente. Para minimizar este efecto y poder aprovechar posteriormente la aplicación de DA, se realiza un filtrado de las probabilidades de salida del clasificador antes de la asignación de clases.

El procedimiento de filtrado automático desarrollado intenta separar las muestras de test que el clasificador indica claramente a qué clase pertenece. Para ello, el procedimiento utiliza valores umbral para las clases mayoritarias y minoritarias, que se comparan con las probabilidades de salida del clasificador. Inicialmente se ha asignado un valor de 0.7 al umbral de probabilidad de la clase minoritaria, por lo que solo se confía en la predicción del clasificador para esta clase si es bastante segura. Para el umbral de la clase mayoritaria se utiliza un valor inicial de 0.55, por lo que bastaría que el clasificador tuviera "algo claro" que la muestra pertenece a la clase mayoritaria, para que esta

UNIVERSIDAD DE MÁLAGA

Algorithm 5 Automatic Filtering Procedure

1: **procedure** Input(*probs*: classifier output probabilities of test samples to belong to each class.)
2: Initialize the threshold values $T_{maj} = 0.55$ for the majority class (0) and $T_{min} = 0.7$ for the minority class (1).
3:
4: **while** *samplesM*.length< 2 **do**
5: **for** $j = 1 \rightarrow N$ **do** ▷ loop over test outputs
6: **if** $probs_j[0] \geqslant T_{maj}$ **or** $probs_j[1] \geqslant T_{min}$ **then**
7: Index j is added to *samplesK*
8: **else**
9: Index j is added to *samplesM*
10: **end if**
11: **end do**
12: Update T_{maj} increasing the value for the next iteration and depending on the current value
13: **end do**
14: **return** *samplesK*: set of samples keeping their classifier output; *samplesM*: set of samples to be tested with the augmented classifier.

sea clasificada como tal por el algoritmo. Este valor se puede asumir bajo para realizar una buena predicción, manteniendo la predicción realizada con el primer clasificador para la mayoría de las muestras de test. Sin embargo, el proceso solo avanza si la probabilidad de salida de un número mínimo de muestras permanece fuera de estos valores umbrales. En la práctica, de forma generalizada no sucede que un número mínimo de muestras queden fuera, momento en el cual se produce, de forma automática, un aumento en el valor umbral en cada iteración realizada, más lentamente a medida que su valor es mayor. De esta manera, se puede hacer una distinción más precisa para las salidas de predicción cuya probabilidad promedio es mayor a 0.95, permitiendo que el umbral alcance un valor que permita que un número suficiente de muestras continúe en el proceso del meta-clasificador. En los experimentos realizados, el número de muestras conservadas mediante este procedimiento, ha superado el 10 % en la mayoría de las pruebas, siendo menor en un reducido número de ellas. Este porcentaje es suficiente para lograr una mejora en el rendimiento general, aunque puede ser visto como un porcentaje bajo. Sin embargo, los experimentos realizados en los que se obligó al procedimiento a que el número de muestras de test mantenidas fuera superior al 10-25 %, vieron disminuida la calidad de los resultados de predicción para algunos conjuntos de datos, mientras que otros no se vieron afectados. Por este motivo, se eligió 2 como número mínimo de muestras. El pseudocódigo del procedimiento de filtrado automático se puede describir

UNIVERSIDAD DE MÁLAGA

como se muestra en el Algoritmo 5.

Una vez que el procedimiento de filtrado se ha realizado con éxito, el procedimiento devuelve los índices pertenecientes a las muestras de test para las que se guarda la predicción y las muestras que aún se mantienen en el proceso meta-clasificador. Tras realizar este procedimiento, el método DA se aplica al conjunto de entrenamiento original, proceso que depende del método estudiado, y se generan muestras sintéticas en el porcentaje de aumento deseado. Estas muestras sintéticas se unen al conjunto de entrenamiento, y con el conjunto aumentado se entrena un segundo clasificador. La aplicación de DA debe ayudar a este segundo modelo a predecir mejor las muestras pertenecientes a la clase minoritaria, según los estudios e investigaciones previamente realizadas. Mediante el segundo clasificador se obtiene pues una predicción para las muestras de test que continuaron en el proceso. Teniendo en cuenta todas las predicciones obtenidas, del primer y segundo modelo clasificador se obtienen las métricas de test para medir la calidad de la predicción realizada. Es necesario entender que este modelo no incluye una votación de las salidas, sino una combinación de las mismas, pues cada clasificador obtiene la predicción para un subconjunto de muestras de test, dependiendo de si fueron seleccionadas por el filtrado automático.

9.4. Implementación de los modelos

Con el objetivo de poder verificar el efecto del esquema meta-clasificador diseñado independientemente de la técnica DA, se seleccionaron varias técnicas ya estudiadas en el presente trabajo, incluyendo métodos de sobremuestreo, submuestreo, adición de ruido y redes generativas.

Como ha sido mencionado anteriormente, en la aplicación de DA con conjuntos desbalanceados las dos variantes más utilizadas son el sobremuestreo y el submuestreo (Drummond et al., 2003; Liu et al., 2008). Para realizar sobremuestreo en los datos se aplica la versión estándar de la técnica SMOTE (Chawla et al., 2002), detallada en el Capítulo 2 sección 2.1.3. Esta técnica selecciona aleatoriamente diferentes vecinos para cada muestra y realiza una interpolación aleatoria entre los vecinos y la muestra. Específicamente, calcula la diferencia entre la muestra y el vecino, multiplica la diferencia por una normalización aleatoria entre [0,1] y luego agrega esto a la muestra original. Mediante este proceso, SMOTE genera nuevas muestras sintéticas ubicadas en el espacio entre las muestras de la clase minoritaria. Para la implementación se utiliza el módulo de Python *imbalanced-learn* (Lemaître et al., 2017) con los valores por defecto (balanceo del conjunto, $sampling_strategy =$ '$auto$'; vecinos a interpolar, $k_neighbors = 5$).

Si bien, técnicamente no es un método de DA, también se aplica submuestreo (*undersampling*). El propósito de los métodos de submuestreo es

UNIVERSIDAD DE MÁLAGA

el mismo que el de sobremuestreo, pero su aplicación es la opuesta. El submuestreo también se centra en balacear el conjunto de datos, pero mediante la reducción del número de muestras de la clase mayoritaria para equipararlo con la clase minoritaria. La técnica utilizada en esta investigación es el submuestreo aleatorio. Esta técnica clásica y sencilla se basa en el descarte aleatorio de muestras de la clase mayoritaria. Una desventaja del submuestreo aleatorio es el posible descarte de muestras características, que son más útiles que otras. Hay otras técnicas de submuestreo, como la Condensed '*Nearest Neighbour Rule*' (Hart, 1968) y los Tomek Links (Tomek, 1976) que intentan reducir estos problemas.

También se aplican los métodos de adición de ruido gaussiano explicados en el Capítulo 4 sección 4.2. Los métodos de adición de ruido diseñados seleccionan aleatoriamente muestras del conjunto de entrenamiento y agregan ruido a los valores del 25 % de las características, ruido que surge de una distribución normal aleatoria con una desviación de 0.2, pues los datos se encuentran normalizados. También se controla que el valor "ruidoso" no exceda los valores límite originales (máximo y mínimo) de la característica. A parte de la versión 'estándar', también se aplica el método para realizar sobremuestreo. Este método, abreviado como 'Noise Bal', únicamente se diferencia del método estándar en la selección aleatoria, pues solo selecciona y genera muestras de entrenamiento de la clase minoritaria.

Los modelos generativos profundos aplicados a esta experimentación se basan en el modelo CGAN (Mirza y Osindero, 2014). Como el modelo GAN estándar, CGAN tiene una estructura dividida en dos redes neuronales entrenadas en un entorno competitivo, pero CGAN tiene en cuenta un parámetro condicional y en ambas redes, parámetro que representa la etiqueta de muestra u otra información en los datos. De esta manera, la red denominada generador tiene como entrada una distribución aleatoria ruidosa y una etiqueta de clase, y produce como salida una distribución asignada al espacio de las muestras reales. El generador crea nuevas muestras "sintéticas" que son similares a las muestras reales y con las que intenta engañar a la segunda red, denominada discriminador. El discriminador recibe como entrada las muestras reales y sintéticas junto con sus etiquetas asignadas, y estima la probabilidad de que la muestra pertenezca o no a la distribución real. Una vez entrenado el modelo CGAN, se inicia el proceso de generación con la entrada a la red del generador, una distribución aleatoria ruidosa y la etiqueta de la clase deseada para la muestra a generar. El generador crea una muestra sintética que, junto con su etiqueta, se pasa al discriminador que estima la probabilidad de que la muestra sea realmente real. Si la muestra sintética se considera real, se guarda con la clase de etiqueta. Si la muestra se considera falsa, se descarta y el método procede a crear una nueva.

Se aplica también el modelo adaptativo denominado ModCGAN, que presenta modificaciones con respecto al CGAN estándar, siendo la principal

UNIVERSIDAD DE MÁLAGA

Tabla 9.1: Características del generador y discriminador que componen los modelos CGAN y ModCGAN utilizados para generar muestras sintéticas.

Capa	Salida	Norm.	Activación
Generador			
Entrada = [z, y]	128 + 1	-	-
Fully connected	1024	Sí	ReLU
Fully connected	512	Sí	ReLU
Fully connected	256	Sí	ReLU
Fully connected	128	Sí	ReLU
Fully connected	Tamaño Datos	No	Sigmoide
Discriminador			
Entrada = [x, y]	Tamaño Datos + 1	-	-
Fully connected	64	Sí	Leaky ReLU
Fully connected	128	Sí	Leaky ReLU
Fully connected	256	Sí	Leaky ReLU
Fully connected	512	Sí	Leaky ReLU
Fully connected	1	No	Sigmoide
Generador optimizador	Adam (lr = 0.0004, beta1 = 0.5, beta2 = 0.9)		
Discriminador optimizador	Adam (lr = 0.0002, beta1 = 0.5, beta2 = 0.9)		
Leaky ReLU	$\alpha = 0.2$		
Backend	Tensorflow		

el uso de un clasificador externo para la asignación de etiquetas de clase. Para reducir el tiempo de ejecución de este modelo, se realiza una modificación adicional con respecto al modelo explicado en el Capítulo 5 sección 5.1. Se utiliza una regresión logística como clasificador externo, en lugar de una red neuronal profunda. Además, el clasificador externo se entrena con DA mediante la aplicación de SMOTE para mejorar su predicción en muestras minoritarias, y se utiliza el discriminador del modelo para diferenciar entre muestras reales o falsas. El proceso de generación de ModCGAN sigue siendo similar al de CGAN, el generador crea una nueva muestra sintética y el discriminador estima la probabilidad de que pueda considerarse real. Si se considera como tal, el clasificador externo predice la etiqueta de clase que debe asignarse. La Tabla 9.1 muestra los detalles de implementación y arquitectura de las redes generador y discriminador de los modelos CGAN y ModCGAN. Las columnas muestran el tipo de capa (*"Fully connected"* representa una capa de neuronas totalmente conectada), el número de neuronas presentes en cada capa, el uso de la normalización por lotes y la función de activación utilizada. En la parte inferior también se muestra el algoritmo de optimización, la pendiente de Leaky ReLU y el *Backend* usado.

Por último, se aplica también el modelo ACGAN (Odena et al., 2017). Este modelo, levemente diferente del modelo CGAN estándar, puede verse similar al modelo ModCGAN desarrollado en su propósito de predecir la

UNIVERSIDAD DE MÁLAGA

etiqueta que debe asignarse a la muestra generada. Para ello, el discriminador de ACGAN presenta dos factores diferenciales respecto al discriminador de CGAN. En primer lugar, no se le proporciona la etiqueta de clase como entrada, únicamente la muestra real o generada. En segundo lugar, para presentar la doble funcionalidad de predecir si la muestra pertenece a la distribución original, y a su vez predecir la etiqueta de clase de la muestra, el discriminador del modelo ACGAN incluye una capa de salida adicional. De este modo, en la configuración de la red discriminador implementada en nuestro modelo ACGAN, la última capa oculta de 512 neuronas presente en el discriminador se encuentra conectada con la capa de salida habitual con 1 neurona y función de activación sigmoide para la discriminación de las muestras, pero, esta capa también se encuentra conectada a una capa oculta de 128 neuronas con normalización por lotes, conectando esta capa oculta a su vez con una capa de salida que presenta tantas neuronas como sean necesarias para representar la etiqueta de clase y con función de activación softmax. Esta configuración es la misma utilizada para la investigación de la predicción del subtipo molecular del cáncer, y se muestra en la Tabla 8.1. Los detalles de todos los métodos y técnicas de regularización, optimización y funciones de activación utilizadas en esta investigación pueden ser vistos con mayor detalle en las correspondientes secciones del Capítulo 2.2.

9.5. Resultados Experimentales

Para investigar el rendimiento de la aplicación DA en el esquema meta-clasificador desarrollado, se seleccionan 7 conjuntos de datos desbalanceados bien conocidos y 5 conjuntos de datos desbalanceados que corresponden a diferentes tipos de cáncer. Sobre estos conjuntos biomédicos se aplica un proceso de selección de características utilizando el modelo LASSO y la importancia Gini del método Random Forest, para reducir el número de genes. La Tabla A.10 del Apéndice A.6 muestra las características de los datos de referencia utilizados para los experimentos. Para aliviar la influencia de la aleatoriedad, el experimento se repitió diez veces con diferentes semillas y en cada una se realizó una división de 10 folds para entrenamiento y prueba. Los resultados de test informados son el promedio de los experimentos. El clasificador utilizado es un sistema de máquinas de vectores de soporte (SVM), proporcionado por el paquete *sklearn* de Python, con los valores de los parámetros predeterminados (Pedregosa et al., 2011).

Para evaluar el rendimiento de clasificación del meta-clasificador utilizando los diferentes métodos de DA, como medidas de evaluación se emplean la capacidad predictiva, sensibilidad, especificidad, medida F_1, *Mathews Correlation Coefficient* (MCC) y área bajo la curva ROC (AUC). Estas métricas están explicadas en más detalle en la sección 3.2 del Capítulo 3. Para todos los conjuntos de datos de referencia se toma como valor positivo la clase mi-

UNIVERSIDAD DE MÁLAGA

noritaria y como valor negativo la clase mayoritaria. Los valores más cercanos a 1 para las métricas indican un mejor rendimiento de predicción.

Si bien la medida F_1 es la métrica por excelencia en la evaluación de resultados sobre la aplicación de conjuntos desbalanceados, esta se ve alterada en exceso por la sensibilidad. Sin embargo, para lograr un buen valor de MCC, el clasificador debe realizar predicciones correctas tanto en la mayoría de los casos negativos como en la mayoría de los casos positivos, por lo que ha demostrado recientemente ser mejor métrica general para la clasificación (Chicco y Jurman, 2020). Para comparar los valores de F_1, MCC, sensibilidad y especificidad obtenidos por diferentes modelos, se calcula la diferencia entre los resultados y se multiplica por 100. Para comparar los valores obtenidos para la capacidad predictiva y AUC entre los diferentes enfoques se usa la diferencia relativa (RD). Esta medida indica el aumento porcentual basado en los resultados de referencia cuando se usa un enfoque aumentado.

Tabla 9.2: Resultados de test (capacidad predictiva y área bajo la curva ROC) obtenidos con los datos originales, los datos aumentados y aplicando esquema de meta-clasificador para todos los problemas de referencia.

	Accuracy			AUC		
Datos	Original	Aumento	MetaClass	Original	Aumento	MetaClass
pima	0.7614	0.7604	**0.7644**	0.8229	0.8260	**0.8263**
newthyroid	0.9797	0.9858	**0.9860**	0.9989	**0.9991**	**0.9991**
ecoli	0.9579	0.9574	**0.9605**	0.9499	0.9468	**0.9500**
segment1	0.8971	0.8981	**0.9045**	0.8839	0.8830	**0.8893**
segment2	0.9146	0.9057	**0.9204**	0.9404	0.9337	**0.9420**
satimage	0.9262	0.9260	**0.9268**	0.9316	0.9307	**0.9346**
oil	0.9683	0.9666	**0.9709**	0.9158	0.9192	**0.9240**
Comp.		-0.08	**0.44**		-0.08	**0.34**
gbm	0.8101	0.8158	**0.8261**	0.8123	0.8246	**0.8435**
coad	0.8403	0.8475	**0.8506**	0.7922	0.8101	**0.8152**
ucec	0.8595	0.8722	**0.8765**	0.8935	0.9103	**0.9193**
thca	0.9651	0.9672	**0.9697**	0.9244	0.9365	**0.9431**
prad	0.9796	0.9815	**0.9823**	0.8925	0.9085	**0.9175**
Comp.		0.66	**1.14**		1.74	**2.87**

La Tabla 9.2 muestra los mejores resultados de capacidad predictiva y AUC obtenidos para los 7 conjuntos de datos bien conocidos y los 5 conjuntos biomédicos de predicción del estatus vital en cáncer. La fuente en negrita indica los mejores valores entre las métricas obtenidas con los datos originales, con los datos aumentados en un proceso de DA simple, y con la aplicación del esquema meta-clasificador (MetaClass). Los resultados son obtenidos para el método de DA estudiado que obtiene el valor de MCC más alto entre todos, con un porcentaje de aumento del 100 % (duplicación del número de

UNIVERSIDAD DE MÁLAGA

muestras de entrenamiento). Se muestra una comparación (Comp.) entre las métricas obtenidas con los enfoques que utilizan DA (aumento simple o meta-clasificador) y los resultados de referencia obtenidos a partir de los datos originales. Como se describió anteriormente, la comparación de estas dos métricas se realiza mediante el cálculo del RD. Los resultados de capacidad predictiva y AUC muestran claramente que para todos los conjuntos de datos (12 de 12) es beneficioso utilizar el esquema meta-clasificador, especialmente para conjuntos de datos biomédicos con un aumento del 1.14 % en la capacidad predictiva y del 2.87 % en AUC en comparación a los valores de prueba obtenidos de los datos originales. Con el aumento simple se logra una mejora para ambas métricas con los conjuntos biomédicos, sin embargo, esta aplicación no logra obtener mejores resultados para los 7 conjuntos bien conocidos.

Tabla 9.3: Resultados de test (sensibilidad y especificidad) obtenidos con los datos originales, los datos aumentados y aplicando esquema de meta-clasificador para todos los problemas de referencia.

	Sensibilidad			Especificidad		
Datos	Original	Aumento	MetaClass	Original	Aumento	MetaClass
pima	0.5346	**0.6243**	0.5883	**0.8821**	0.8328	0.8582
newthyroid	0.9093	0.9743	**0.9857**	**0.9933**	0.9881	0.9861
ecoli	0.8381	**0.9014**	0.8619	**0.9800**	0.9677	0.9787
segment1	**0.4758**	0.4098	0.4091	0.9673	0.9794	**0.9871**
segment2	0.5569	**0.6162**	0.6149	**0.9742**	0.9540	0.9714
satimage	0.5312	**0.5573**	0.5473	**0.9687**	0.9657	0.9677
oil	0.3625	**0.5013**	0.4375	**0.9953**	0.9874	0.9947
Comp.		**5.37**	3.38		-1.23	**-0.24**
gbm	0.3571	0.4364	**0.4571**	**0.9255**	0.9124	0.9200
coad	0.3667	0.4960	**0.5600**	**0.9548**	0.9326	0.9210
ucec	0.4443	**0.5143**	0.4786	0.9463	0.9470	**0.9597**
thca	0.1406	0.2238	**0.3500**	**0.9944**	0.9943	0.9923
prad	0.0350	**0.0975**	0.0750	0.9971	0.9978	**0.9991**
Comp.		8.49	**11.54**		-0.68	**-0.52**

La Tabla 9.3 muestra los mejores resultados de sensibilidad y especificidad obtenidos para los conjuntos estudiados. Los resultados son obtenidos para el método de DA que obtiene el valor de MCC más alto, siendo los mismos que en la Tabla 9.2. Como se describió anteriormente, la comparación (Comp.) de estas dos métricas se realiza mediante la diferencia de resultados multiplicado por 100. Los resultados de estas métricas confirman que la aplicación de DA (simple o meta-clasificador) mejora la sensibilidad (clasificación minoritaria) a costa de obtener una peor especificidad (clasificación mayoritaria). Este hecho es más visible utilizando el enfoque de aumento simple con un aumento del 5.37 % en la sensibilidad y una pérdida del -1.23 % en

UNIVERSIDAD DE MÁLAGA

la especificidad para los 7 conjuntos de datos bien conocidos, y un aumento del 8.49 % en la sensibilidad y una pérdida del -0.68 % en especificidad para conjuntos de datos biomédicos. Los resultados parecen indicar que mediante el esquema meta-clasificador se logran los objetivos establecidos, beneficiándose de ambos comportamientos, minimizando la pérdida en especificidad ligeramente, pero obteniendo una mayor sensibilidad, incluso obteniendo los mejores resultados para los conjuntos de datos biomédicos (11.54 %).

Tabla 9.4: Resultados de test (F_1 y Mathews Correlation Coefficient) obtenidos con los datos originales, los datos aumentados y aplicando esquema de meta-clasificador para todos los problemas de referencia.

	F_1			MCC		
Datos	Original	Aumento	MetaClass	Original	Aumento	MetaClass
pima	0.6090	**0.6436**	0.6341	0.4516	0.4654	**0.4663**
newthyroid	0.9328	0.9572	**0.9586**	0.9242	0.9501	**0.9511**
ecoli	0.8597	0.8684	**0.8707**	0.8377	0.8456	**0.8496**
segment1	**0.5691**	0.5326	0.5476	0.5277	0.5146	**0.5441**
segment2	0.6484	0.6480	**0.6869**	0.6150	0.6018	**0.6499**
satimage	0.5826	**0.5939**	0.5923	0.5465	**0.5556**	0.5552
oil	0.4867	**0.5575**	0.5548	0.5149	0.5519	**0.5729**
Comp.		1.61	**2.24**		0.96	**2.45**
gbm	0.4329	0.4902	**0.5161**	0.3359	0.3842	**0.4174**
coad	0.4649	0.5521	**0.5774**	0.4175	0.4745	**0.5000**
ucec	0.5150	**0.5772**	0.5670	0.4587	0.5175	**0.5226**
thca	0.1953	0.3046	**0.4434**	0.2193	0.3412	**0.4706**
prad	0.0421	**0.1338**	0.1133	0.0391	**0.1478**	0.1329
Comp.		8.15	**11.34**		7.89	**11.46**

La Tabla 9.4 muestra los mejores resultados de F_1 y MCC obtenidos para los conjuntos estudiados. En este caso, la comparación (Comp.) de las métricas se realiza mediante la diferencia de resultados multiplicado por 100. Al igual que el resto de métricas, confirman que mediante la aplicación de DA se mejora el rendimiento del modelo clasificador. Los mejores valores de F_1 se reparten entre ambos enfoques de aumento. Sin embargo, los resultados de MCC muestran que para la mayoría de los conjuntos de datos (10 de 12) es más beneficioso utilizar el esquema de meta-clasificador que el aumento simple, mostrando una gran mejora en la capacidad predictiva general con un aumento del 2.45 % para los conjuntos de datos conocidos y del 11.46 % para los conjuntos de datos biomédicos en comparación con el MCC obtenido de los datos originales.

La Figura 9.2 muestra los valores de AUC, MCC, sensibilidad y especificidad ordenados según el balanceo de clases presente en los conjuntos de datos originales, que representan el eje x. También se muestra la regresión lineal ajustada a los resultados obtenidos con el esquema meta-clasificador.

UNIVERSIDAD DE MÁLAGA

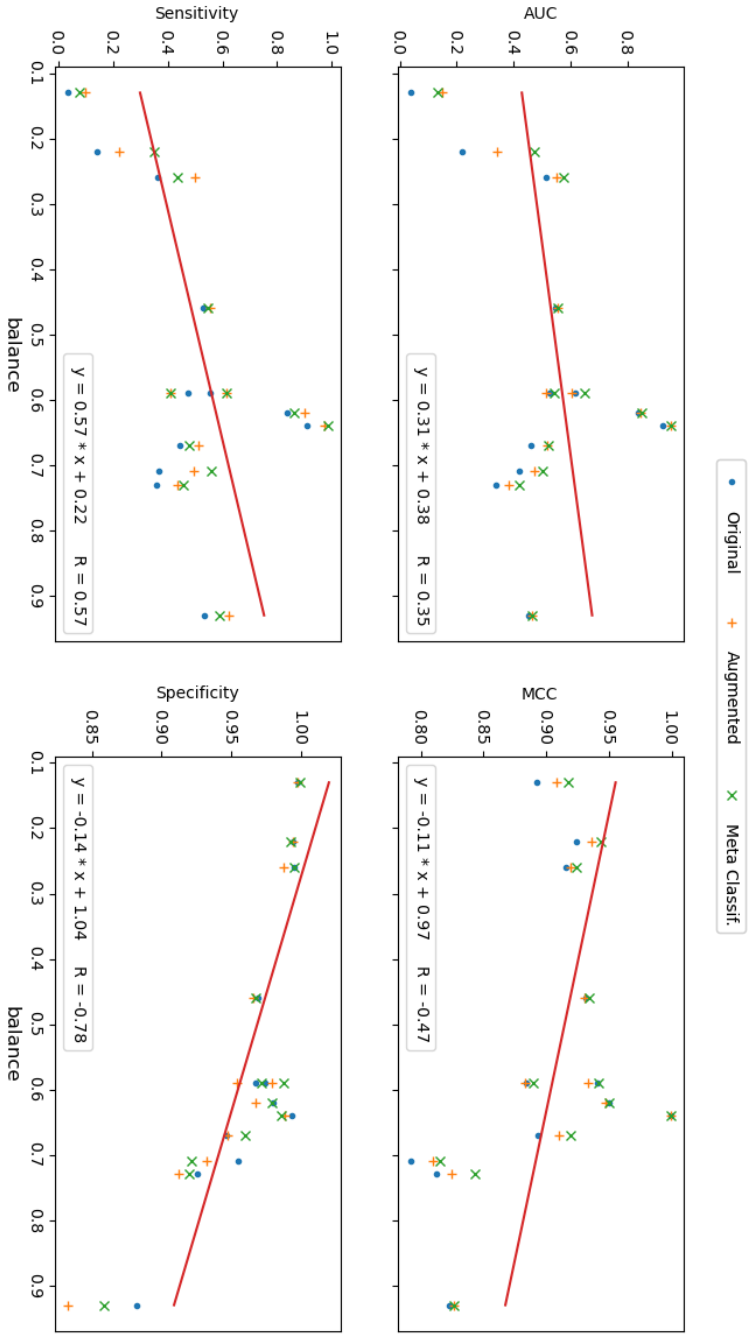

Figura 9.2: Comparación de las métricas de test obtenidas con los diferentes enfoques respecto al balanceo de los datos originales. Los puntos representan los resultados obtenidos con los datos originales, la cruz con los datos aumentados y las aspas con el esquema meta-clasificador. La línea continua es una regresión lineal ajustada a los resultados del esquema meta-clasificador.

UNIVERSIDAD DE MÁLAGA

Los coeficientes de correlación para el AUC y la sensibilidad indican una correlación positiva moderada entre el balanceo de los datos y la ganancia de estas métricas, 0.35 y 0.57 respectivamente. Por otro lado, los coeficientes de correlación para MCC y la especificidad indican una correlación negativa, -0.47 y -0.78 respectivamente. Este análisis indica que, en la aplicación sobre los conjuntos de datos más desbalanceados, la especificidad alcanzada con todos los enfoques es alta y el valor de MCC aumenta, mientras que los valores de sensibilidad y AUC obtenidos son bajos debido a la dificultad de predecir la clase minoritaria. Cuando los datos están más balaceados, los resultados se comportan forma inversa, reduciendo la especificidad lograda y aumentando su sensibilidad.

La Tabla 9.5 muestra una comparación entre los diferentes métodos y modelos de DA aplicados en cada enfoque de aumento, simple o meta-clasificador, utilizado. Los valores mostrados representan los valores medios obtenidos con el enfoque de aumento simple y con el esquema de meta-clasificador en comparación con los resultados de referencia obtenidos con los datos originales. Los valores de capacidad predictiva y AUC se calculan a partir de RD, mientras que los valores de sensibilidad, especificidad y MCC indican la diferencia porcentual. Los resultados muestran que la aplicación de los métodos de DA con el esquema meta-clasificador mejoran los resultados de capacidad predictiva, AUC, F_1 y MCC en comparación con su aplicación con el aumento simple. La única excepción es el valor de AUC al aplicar SMOTE, que obtiene -3.75 % con aumento simple respecto al -3.92 % con el esquema meta-clasificador.

Tabla 9.5: Resultados obtenidos utilizando el esquema de datos aumentados y meta-clasificador con cada método de DA aplicado en comparación con los resultados de los datos originales. Los resultados de capacidad predictiva y AUC se calcularon utilizando RD.

Método		Acc	Sens	Espec	F_1	MCC	AUC
CGAN	Aumento	-2.29	-2.90	-2.02	-5.35	-6.38	-4.46
	MetaClass	-0.90	-1.83	-0.65	-1.08	-0.59	-0.63
ACGAN	Aumento	-0.68	0.17	-1.36	-1.62	-2.78	-2.16
	MetaClass	0.01	-0.15	-0.42	0.10	0.04	0.26
ModCGAN	Aumento	-0.68	0.43	-1.35	-1.49	-2.40	-1.12
	MetaClass	0.03	0.26	-0.47	0.49	0.42	0.38
NOISE	Aumento	-0.15	-0.83	0.08	-0.27	-0.10	0.04
	MetaClass	0.14	-0.77	-0.12	0.23	0.83	0.67
NOISE Bal	Aumento	-0.39	9.00	-1.57	4.32	3.45	0.55
	MetaClass	0.61	6.27	0.26	5.44	5.69	1.35
SMOTE	Aumento	-2.68	14.03	-5.33	-0.10	-1.87	-3.75
	MetaClass	-0.15	3.46	-0.98	1.42	0.50	-3.92
UNDER	Aumento	-13.47	28.48	-13.57	-2.80	-3.85	-6.09
	MetaClass	-0.50	9.54	-0.06	5.84	5.08	0.52

UNIVERSIDAD DE MÁLAGA

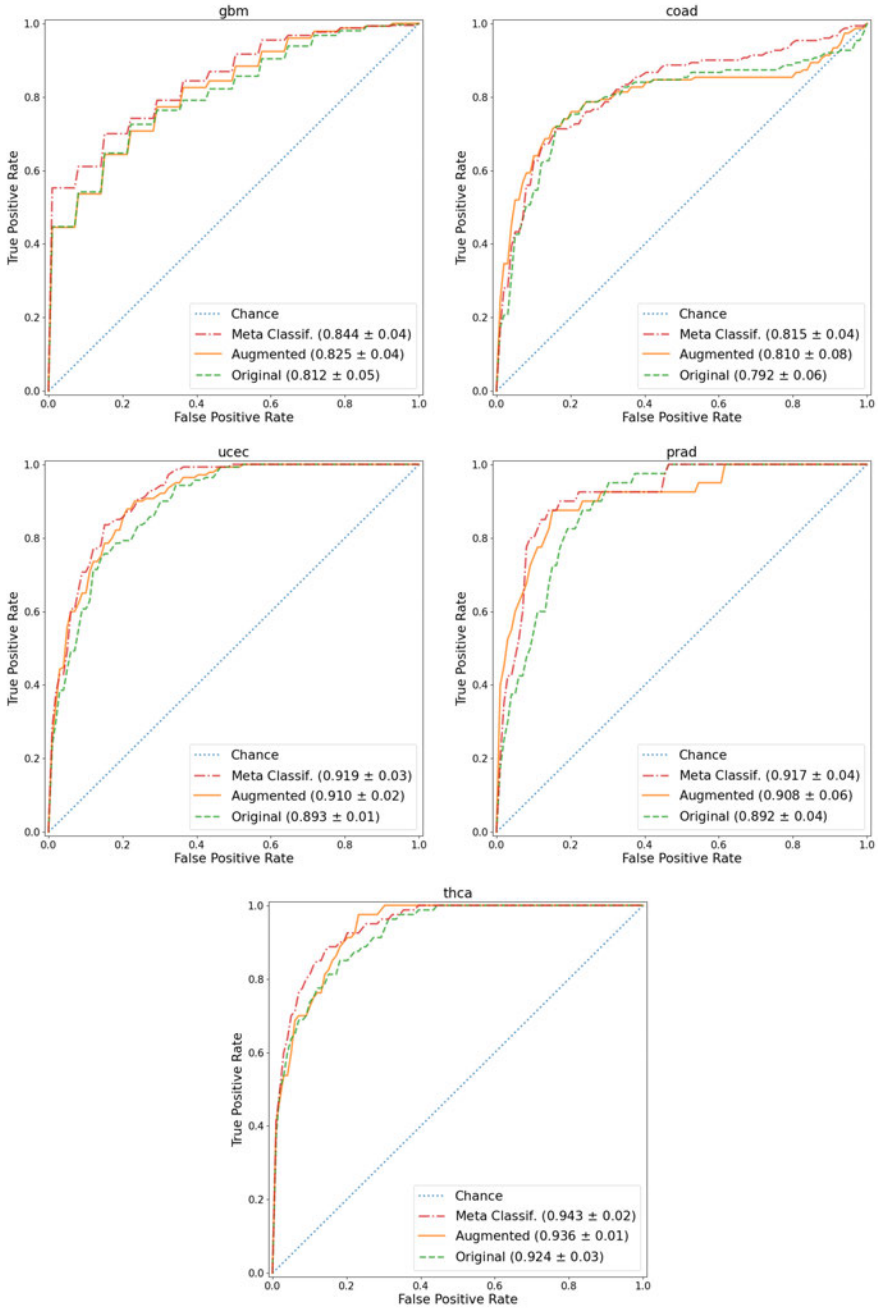

Figura 9.3: Curvas ROC obtenidas utilizando los diferentes enfoques de aumento de datos para conjuntos de datos biomédicos. La línea punteada representa el azar; la línea discontinua representa el enfoque con los datos originales; la línea continua representa el aumento de datos; y la línea de puntos y guiones representa el uso del esquema meta-clasificador.

UNIVERSIDAD DE MÁLAGA

Con respecto a una comparación entre los métodos de DA, la Tabla 9.5 indica que se alcanza el mejor rendimiento de predicción general con NOISE Bal, con un aumento del 5.69 % en MCC en comparación con los datos originales. Estos resultados también muestran que el enfoque de aumento simple mejora en gran medida la sensibilidad, aunque tiene un impacto en el rendimiento predictivo. Especialmente altos son los valores de sensibilidad obtenidos con las técnicas SMOTE y de submuestreo aleatorio aplicadas en un enfoque de aumento simple con un 14.03 % y 28.48 % de aumento respectivamente, si bien no logran obtener una mejora en la capacidad de predicción. Con respecto a los modelos generativos profundos, el modelo CGAN obtiene resultados pobres con ambos enfoques, siendo mejorados con el modelo ModCGAN desarrollado para el estudio. Destacar que el submuestreo puede ser una técnica eficaz para mejorar el rendimiento general, pero únicamente en su aplicación con el meta-clasificador. En una aplicación simple sus resultados empeoran los obtenidos con casi todos los métodos de DA.

Para realizar un análisis más detallado sobre los resultados obtenidos con los 5 conjuntos de datos biomédicos, se muestran en la Figura 9.3 las curvas ROC obtenidas para estos conjuntos. Las curvas ROC se basan en los resultados que se muestran en la Tabla 9.2. Un análisis de los resultados indica que el rendimiento logrado depende en gran medida del conjunto de datos analizado. Con los conjuntos *gbm* y *coad*, los resultados utilizando el aumento simple y los datos originales son bastante similares, al igual que para *thca* cuando se tiene una ratio de falsos positivos menor que 0.1, mientras que con *prad* y *ucec* las diferencias son mayores. La aplicación del meta-clasificador solo logra resultados similares a los datos originales con el conjunto *coad*, para el resto los resultados son mejorados de forma clara. De forma general, se puede afirmar que mediante la aplicación del esquema meta-clasificador se obtienen resultados ligeramente mejores que los obtenidos mediante la aplicación de un aumento simple, y mayores que utilizando solo los datos originales. Para la mayoría de ellos, aplicando el esquema de meta-clasificador, se pueden obtener buenos resultados independientemente del punto de corte que se desee seleccionar.

Adicionalmente se realizado la aplicación de una red neuronal profunda con los 5 conjuntos biomédicos, para mostrar el rendimiento del esquema meta-clasificador. La arquitectura es la utilizada para la predicción del subtipo molecular del cáncer (Capítulo 8), mostrada en la Tabla 8.1. Siendo estos conjuntos reducidos en número de instancias (<600), se ha podido utilizar una cuarta capa oculta sin grandes limitaciones. En la Tabla 9.6 se muestran las métricas de test obtenidas para el método de DA que obtiene el mayor valor de MCC, aplicando un aumento del 100 %. Se muestra además una comparación (Comp.) entre las métricas obtenidas con los enfoques que utilizan DA (aumento simple o meta-clasificador) y los resultados de referencia obtenidos a partir de los datos originales. Los resultados indican que bene-

UNIVERSIDAD DE MÁLAGA

Tabla 9.6: Resultados de test utilizando una red neuronal como modelo clasificador, entrenada con los datos originales, los datos aumentados mediante un procedimiento simple, y aumentados aplicando el esquema de meta-clasificador, para los problemas biomédicos de referencia.

Datos	Método	Acc	Sens	Espec	F_1	MCC	AUC
gbm	Original	0.7746	0.2407	0.9105	0.3024	0.1871	0.6703
	Aumento	0.7838	0.2314	0.9244	0.3028	0.2025	0.6806
	MetaClass	0.7888	0.2407	0.9284	0.3163	0.2209	0.6841
coad	Original	0.8223	0.3633	0.9334	0.4434	0.3561	0.7172
	Aumento	0.8164	0.3913	0.9192	0.4536	0.3531	0.7363
	MetaClass	0.8314	0.3900	0.9382	0.4741	0.3921	0.7363
ucec	Original	0.8201	0.3879	0.9104	0.4271	0.3240	0.7526
	Aumento	0.8290	0.3743	0.9240	0.4307	0.3382	0.7747
	MetaClass	0.8296	0.3857	0.9224	0.4390	0.3454	0.7658
thca	Original	0.9580	0.1075	0.9890	0.1523	0.1488	0.6521
	Aumento	0.9594	0.1113	0.9903	0.1614	0.1631	0.6257
	MetaClass	0.9629	0.1075	0.9940	0.1690	0.1919	0.6651
prad	Original	0.9785	0.0575	0.9956	0.0886	0.0966	0.5864
	Aumento	0.9778	0.0975	0.9941	0.1378	0.1415	0.5894
	MetaClass	0.9802	0.0900	0.9967	0.1420	0.1665	0.5703
Comp.	Aumento	0.29	0.98	0.26	1.45	1.71	0.83
	MetaClass	0.90	1.14	0.82	2.53	4.09	1.27

ficioso utilizar el esquema de meta-clasificador, obteniendo mejores valores promedios en comparación con los resultados obtenidos con los datos originales y el aumento simple. Destaca un aumento del 4.09 % de MCC respecto a los datos originales, por el 1.71 % obtenido con el aumento simple. Sin embargo, la mejora mediante DA es menor que la obtenida con SVM. Esto vuelve a evidenciar la dificultad presente en las redes profundas para mejorar su rendimiento únicamente mediante DA en conjuntos no estructurados.

También se realiza una comparación entre los diferentes métodos de DA aplicados en cada enfoque, simple o meta-clasificador, con la red neuronal para los conjuntos biomédicos. La Tabla 9.7 muestra los valores medios obtenidos con el enfoque de aumento simple y con el esquema meta-clasificador en comparación con los resultados de referencia obtenidos con los datos originales. Al igual que los resultados de SVM, estos muestran que la aplicación de los métodos de DA con el esquema meta-clasificador mejoran en la mayoría de los casos las métricas de test, en comparación con su aplicación con el aumento simple. En este caso, se logran valores positivos con el uso del meta-clasificador en capacidad predictiva obtenida, si bien siguen siendo valores bajos de mejora, al igual que los valores de MCC obtenidos. Cabe destacar que los valores obtenidos no pueden ser comparables con el resto de estudios y análisis realizados durante el presente trabajo de tesis, en cuanto a que los experimentos no han sido realizados con la misma especificación.

UNIVERSIDAD DE MÁLAGA

Tabla 9.7: Resultados obtenidos utilizando aumento simple y el meta-clasificador con cada método de DA aplicado en comparación con el conjunto de datos original. Los resultados de capacidad predictiva y AUC se calcularon utilizando RD. Se emplea una red neuronal como clasificador.

Método		Acc	Sens.	Espec.	F_1	MCC	AUC
CGAN	Aumento	-0.26	-0.66	0.02	-0.79	-1.22	-2.17
	MetaClass	0.75	0.05	0.83	1.32	3.00	-0.72
ACGAN	Aumento	-0.20	-0.46	0.05	-0.52	-0.91	-1.12
	MetaClass	0.66	0.19	0.72	1.35	2.79	-0.31
ModCGAN	Aumento	0.12	-0.45	0.44	0.17	0.42	-2.03
	MetaClass	0.55	0.43	0.69	1.66	2.76	0.40
NOISE	Aumento	0.09	-2.40	0.63	-1.67	-0.76	-2.48
	MetaClass	0.55	-0.62	0.73	0.50	2.03	-1.18
NOISE Bal	Aumento	-0.25	3.25	-0.78	2.73	2.00	0.46
	MetaClass	0.76	1.34	0.67	2.63	3.86	0.87
SMOTE	Aumento	-0.60	-1.18	-0.17	-1.37	-1.79	-4.05
	MetaClass	0.42	0.16	0.68	1.31	2.04	-2.03
UNDER	Aumento	-2.62	3.44	-2.34	-1.10	-2.68	-0.95
	MetaClass	0.52	0.10	0.70	1.29	2.20	-1.53

Por último, para analizar el tiempo de cálculo necesario para realizar la aplicación del esquema meta-clasificador, se muestra en la Figura 9.4 los tiempos de cómputo (en segundos y en escala logarítmica) empleados para cada enfoque cuando se usa una red neuronal, graficados en función del número de instancias de los conjuntos de datos de referencia en escala logarítmica. Se incluye además el tiempo de cómputo necesario para cada enfoque de DA si se aplica un modelo basado en GAN (CGAN, ACGAN o ModCGAN), de forma que los otros tiempos de ejecución se obtienen usando el resto métodos estudiados (adición de ruido, SMOTE y submuestreo), entre los cuales no existen diferencias significativas. Gracias a la aplicación de ModCGAN con una regresión logística como clasificador generativo, los tiempos de cálculo empleados son similares al resto de modelos generativos profundos.

Se observa de forma clara el aumento en el tiempo de cálculo que implica ambos enfoques de DA en comparación con el uso de los conjuntos de datos sin aumento, pues el número de instancias de entrenamiento se ve alterado. Sin embargo, el tiempo necesario para realizar la aplicación del esquema meta-clasificador no conlleva tanto aumento como cabe esperar, siendo mayor el tiempo de cálculo empleado por el enfoque simple con la aplicación de un modelo generativo profundo. Todos los experimentos se ejecutaron en un PC con Windows 10 64-bits OS, equipado con una CPU Intel Core i7-6700K 4.00 GHz, 32 GB RAM, y una NVIDIA® Titan Xp GPU con 12 GB RAM. El software utilizado para los experimentos fue CUDA® Toolkit 10.1, cuDNN v8.0, Python v3.7.0, y TensorFlow v2.1 con soporte GPU.

UNIVERSIDAD DE MÁLAGA

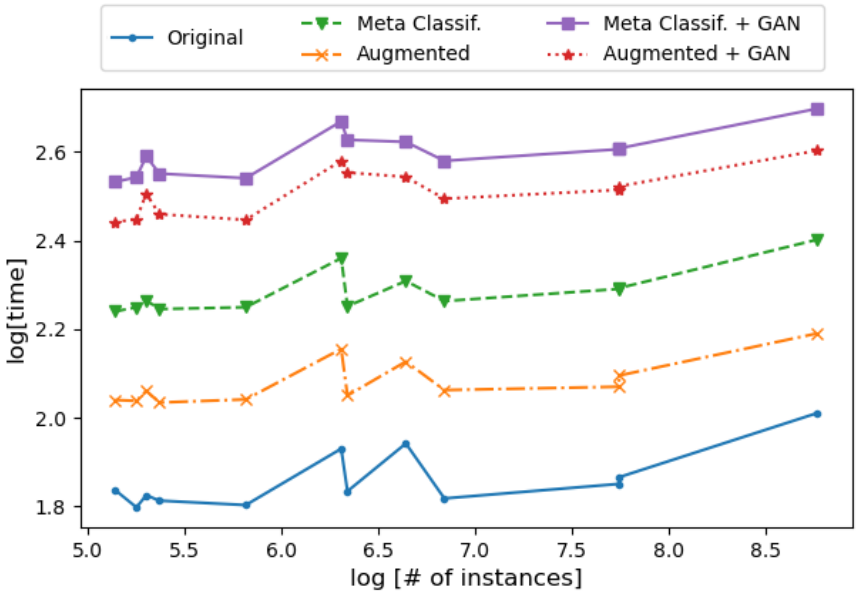

Figura 9.4: Tiempo de cálculo utilizado en los diferentes experimentos con una red neuronal. Código de colores: línea continua (punto), datos originales; línea de guiones y puntos (aspa) datos aumentados; línea discontinua (triángulo) esquema del meta-clasificador; línea de puntos (asterisco) datos aumentados y modelo DA basado en GAN; línea continua (cuadrado) esquema de meta-clasificador y un modelo DA basado en GAN.

9.6. Conclusiones

En esta investigación se introduce un esquema meta-clasificador para la tarea de aumento de datos con el objetivo de mejorar el rendimiento de la predicción en problemas de clasificación desbalanceada. La propuesta consiste esencialmente en entrenar un clasificador con el conjunto de datos original, filtrar las probabilidades de salida de este modelo sobre el conjunto de test para mantener la predicción de algunas de las muestras de presentes en este conjunto, y tomar para el resto la predicción realizada con un clasificador entrenado con aumento de datos. Una parte importante del procedimiento es el proceso de filtrado automático de las probabilidades de salida de test. Este proceso separa las predicciones de las muestras de test para obtener un beneficio de la especificidad del clasificador cuando se entrena solo con datos originales y de la sensibilidad cuando se entrena con datos aumentados. Se realizó una gran cantidad de simulaciones aplicando diversos enfoques a siete conjuntos de datos desbalanceados bien conocidos de los repositorios UCI y KEEL, y cinco conjuntos de datos biomédicos reales obtenidos del repositorio TCGA para predicción de eventos en intervalo de tiempo fijo, concretamente el estatus vital en cáncer.

UNIVERSIDAD DE MÁLAGA

El resultado general de los análisis realizados indica que el uso del esquema meta-clasificador propuesto puede conducir a un aumento en el rendimiento predictivo de aproximadamente un 2.5 % en promedio de MCC en comparación con el uso de los datos originales en problemas de datos conocidos. Este rendimiento aumenta hasta un 11.5 % cuando se trata de datos médicos reales sobre un problema de predicción del estatus vital de un paciente con cáncer. Además, a través del esquema propuesto, la capacidad predictiva y el AUC se mejoran con respecto a los resultados obtenidos utilizando un enfoque simple de aumento de datos. Con el análisis realizado, se sustenta la hipótesis de que los modelos clasificadores tradicionales entrenados con datos aumentados aumentan en gran medida la sensibilidad alcanzada, pero empeoran la especificidad en la misma medida. Si se utiliza una red neuronal profunda como clasificador, este efecto puede verse invertido dependiendo de la configuración de los datos aumentados. Con nuestra propuesta, la diferencia entre sensibilidad y especificidad parece disminuir, obteniendo una mejora independientemente del método DA aplicado.

UNIVERSIDAD DE MÁLAGA

UNIVERSIDAD
DE MÁLAGA

Parte V

Conclusiones

UNIVERSIDAD
DE MÁLAGA

UNIVERSIDAD
DE MÁLAGA

Capítulo 10

Conclusiones y Trabajos Futuros

Resumen: Este capítulo describe una discusión de los capítulos anteriores y las conclusiones generales a las que se ha llegado después de la finalización de este doctorado. tesis, y también indica una serie de posibles líneas de investigación futuras.

10.1. Discusiones

La investigación realizada en este trabajo de tesis comprende el diseño de técnicas de aumento de datos (DA), su aplicación en conjuntos habituales del aprendizaje automático, la aplicación de DA en problemas concretos del ámbito de la bioinformática y el diseño de un nuevo método basándose en la experiencia adquirida. Dicho esto, esta tesis se ha dividido en tres partes principales, que coinciden con las diferentes secciones propuestas.

La Parte II se centra en el diseño e implementación de métodos y modelos de DA. En el Capítulo 4 se han presentado varios métodos de DA basados en adición de ruido, un método denominado FNA (*Forward Noise Adjustment*) y dos métodos basados en adición aleatoria de ruido gaussiano. El método FNA desarrollado implementa un enfoque voraz en el cual se clasifican y ordenan las variables del conjunto estudiado, se agregan diferentes niveles de ruido usando una distribución uniforme, y se registra el nivel de ruido cuya aplicación en DA conduce a una mejora en predicción. Al final se obtiene una relación de niveles de ruido óptimos a aplicar para cada variable.

La aplicación del método FNA en conjuntos con un número reducido de muestras, consigue una ligera mejora en predicción utilizando una red profunda. La mejora es lograda realizando un DA igual a multiplicar por seis el número de muestras disponibles, acelerando la convergencia del modelo.

UNIVERSIDAD DE MÁLAGA

Sin embargo, este método presenta la desventaja de requerir un tiempo de computación muy elevado respecto otros métodos de DA tradicionales.

Debido al rendimiento del método FNA, en el Capítulo 5 se plantea el uso de modelos generativos profundos, VAE y GAN. Como se puede observar en la revisión del estado del arte realizada, estos modelos han sufrido un impresionante auge en los últimos años. Su rendimiento en generación de imágenes es excepcional, lo que ha provocado una expansión a otras aplicaciones, incluyendo la bioinformática. El modelo adaptativo desarrollado implementa un clasificador generativo que permite el uso de múltiples modelos generativos (GAN/VAE) en un proceso de creación con etiquetado para tareas supervisadas. También se introdujeron dos modificaciones a la distribución de clases, intentando incrementar el rendimiento en clasificación mediante un aumento que altere o respete el equilibrio del conjunto.

Es necesario incidir en el uso de la validación cruzada y de diferentes semillas, con objeto de mejorar la generalización y validez de todos los experimentos realizados en la tesis. También en el ajuste de hiperparámetros de los modelos y métodos utilizados, especialmente en los modelos GAN, que pueden presentar un comportamiento inestable. De la experimentación se concluye que las técnicas basadas en VAE y GAN constituyen enfoques útiles para aumentar la capacidad de predicción cuando se analizan pequeños conjuntos de datos, ya que generan datos de suficiente calidad que conducen a un incremento en la capacidad de predicción. Este aumento de rendimiento se produce tanto con modelos de aprendizaje profundo, como mediante regresión logística, si bien los resultados base utilizando DL mejoran los obtenidos mediante LR. Por otro lado, los modelos condicionales (CVAE y CGAN) parecen ser capaces de generar muestras de una calidad mayor, especialmente en su aplicación con problemas relacionados con la biomedicina, y se ven menos influenciados por las modificaciones de balanceo de los datos.

Tras el desarrollo de métodos y modelos de DA, en la Parte III se presenta su aplicación en problemas del ámbito de la bioinformática. Concretamente, trabajando con conjuntos de datos no estructurados, es decir, sin información espacial o temporal. De este modo, en el Capítulo 6 se propone la aplicación de DA con el objetivo de mejorar el rendimiento en predicción del estatus vital de pacientes con cáncer en intervalo de tiempo fijo, a partir del análisis de perfiles de expresión génica RNA-Seq. El estudio comprende 18 tipos diferentes de cáncer de la base de datos *The Cancer Genome Atlas* (TCGA).

La aplicación de DA lleva a un ligero aumento en la capacidad de predicción de una red neuronal profunda, respecto a los resultados de referencia obtenidos tras realizar un proceso de selección de características, siendo mayor la mejora respecto a los datos originales. La selección de características resulta fundamental para incrementar el potencial de predicción en conjuntos biomédicos, además de ayudar a los modelos generativos profundos a crear muestras de mayor calidad. Los modelos GAN aplicados mejoran los resulta-

UNIVERSIDAD DE MÁLAGA

dos de generalización obtenidos con técnicas tradicionales como SMOTE y la adición de ruido, obteniendo mediante el modelo adaptativo diseñado los mejores valores medios de AUC. Sin embargo, los análisis muestran una posible limitación en la aplicación en entornos reales, la disminución del rendimiento alcanzado en la selección de un solo método para diferentes conjuntos de datos. Esta selección debería hacerse dependiendo de las características de balanceo o número de instancias.

Mediante métricas estadísticas y métodos multivariantes se analizan las características de las muestras generadas mediante DA. Los modelos basados en GAN agregan una mayor variabilidad a los conjuntos, generando muestras que se ajustan a la distribución real pero también algunas posicionadas fuera de estas distribuciones. Las muestras generadas con el modelo adaptativo ModCGAN presentan una mayor similitud con las muestras reales que las generadas con el modelo CGAN estándar, lo que indica una mejora en la calidad desde un punto de vista estadístico, verificando su capacidad para replicar los datos de expresión génica.

En el Capítulo 7 se estudia la aplicación de DA en la predicción de la enfermedad Niemman-Pick tipo C1 (NPC1) a partir de muestras metabolómicas. La aplicación especialmente de un método de adición de ruido gaussiano con un propósito de balanceo (Noise Bal), aumentando únicamente el número de muestras de la clase minoritaria, conduce a un gran aumento en la sensibilidad y puntuación F_1, con independencia de la técnica utilizada como modelo clasificador. Los resultados obtenidos de métodos multivariantes tradicionalmente usados en análisis metabolómico respaldan esta conclusión, mostrando una mejora en la exactitud y la bondad de predicción.

Los análisis de la calidad de las muestras generadas indican una buena capacidad de los métodos DA para replicar la información metabólica, mostrando una posible limitación de los modelos basados en GAN. Si bien estos modelos son capaces de replicar información que se ajusta a la distribución de las muestras "reales", dada la función interna de la red discriminadora, los modelos no crean muestras sintéticas cercanas a valores atípicos, a muestras que podrían considerarse '*outliers*'. La replicación de la información metabolómica ha demostrados ser útil además para encontrar biomarcadores, apoyando otros resultados obtenidos en este campo, y que pueden emplearse para el diagnóstico y monitorización de la enfermedad NPC1.

Terminando con esta parte, se aplican estos modelos en la predicción del subtipo molecular del cáncer de mama en el Capítulo 8. Se analizan perfiles de expresión génica del sistema inmunitario de pacientes afectadas en la región de Málaga. Además, se han tratado dos problemas a la hora de la predicción del subtipo, un problema binario realizando una clasificación según el subtipo sea de grado bajo o alto; y un problema multiclase realizando una clasificación en los cinco subtipos presentes. La predicción multiclase ha resultado ser un problema extremadamente complejo, implicando un menor

UNIVERSIDAD DE MÁLAGA

rendimiento de los modelos clasificadores utilizados (redes profundas, Random Forest y máquinas de vectores de soporte), incluso tras la aplicación de DA. En los experimentos realizados, la mayoría de las muestras se predicen como pertenecientes a las clases mayoritarias.

Si se simplifica el enfoque a la predicción binaria del subtipo de grado bajo o alto, los resultados obtenidos son mejores. El aumento mediante la adición de ruido con propósito de sobremuestreo y mediante el modelo adaptativo ModCGAN con modificación del balanceo, aumentan la capacidad predictiva y el AUC alcanzado, pero especialmente aumentan la sensibilidad. El análisis de las características y la calidad de las muestras generadas en este problema apoyan las conclusiones obtenidas con otros problemas bioinformáticos.

Finalmente, debido al aumento en sensibilidad observado gracias al DA en los problemas bioinformáticos, era interesante el desarrollo de un método capaz de aprovechar esta ventaja del DA, junto con la mayor especificidad alcanzada mediante el conjunto sin aumentar. De este modo, en la Parte IV, compuesta por el Capítulo 9, se presenta un esquema meta-clasificador para aumentar el rendimiento en problemas de clasificación desbalanceados. La propuesta se fundamenta en el entrenamiento con el conjunto de datos original, el filtrado automático de las predicciones sobre las muestras de test para mantener la predicción de algunas de las muestras, y el uso para el resto de la predicción realizada con un clasificador entrenado con DA.

Las simulaciones aplicando este esquema muestran una mejora en la capacidad predictiva, AUC y MCC con respecto a la aplicación de un enfoque simple de DA. El análisis realizado sustenta la hipótesis de que los clasificadores entrenados utilizando datos aumentados alcanzan en gran medida un aumento de sensibilidad, pero empeoran la especificidad. La diferencia entre sensibilidad y especificidad disminuye con el esquema meta-clasificador, pudiendo obtener una mejora independientemente del método DA aplicado.

Analizado el trabajo desarrollado, se dilucidan como líneas futuras la aplicación de los modelos desarrollados a otros problemas bioinformáticos (ómicos o procesamiento de lenguaje natural), así como su comparación con un mayor número de métodos de DA del estado del arte.

10.2. Conclusiones

Este trabajo de tesis ha cubierto la aplicación de aumento de datos (DA) en aplicaciones del ámbito de la bioinformática, para el análisis de conjuntos de carácter genómico y haciendo uso de diferentes modelos del aprendizaje profundo (DL). Hemos mejorado el estado del arte del DA mediante el desarrollo de nuevos métodos y su aplicación específica en diferentes problemas biomédicos de relevancia. Para ello, hemos aplicado métodos de adición de ruido, sobremuestreo y modelos generativos profundos.

UNIVERSIDAD DE MÁLAGA

Partiendo de que los modelos de DL obtienen resultados impresionantes tanto en clasificación como en generación de imágenes sintéticas, hemos comprobado que este efecto se ve reducido en su aplicación con conjuntos genómicos. Probablemente, esto se debe al hecho de no poder extraer información espacial de los datos. De este modo, en algunos casos los resultados de predicción no son tan elevados como los obtenidos mediante técnicas tradicionales de aprendizaje automático, que además tienen la ventaja de requerir un tiempo de computación exponencialmente menor.

Esto se ve reflejado igualmente en la creación de muestras mediante los modelos generativos profundos. Ha sido necesario el desarrollo de un modelo adaptativo y el control del balanceo de las clases, para obtener resultados más efectivos en su aplicación con datos no estructurados. Además, teniendo en cuenta los análisis realizados, su efecto viene dado por ciertas características de los conjuntos de datos. Así, su uso sería recomendable siempre que el número de variables no sea extremadamente elevado, siendo paliado por el uso de métodos de selección de características; el número de muestras disponibles no sea exageradamente reducido, provocando ciertos problemas en la aplicación del modelo adaptativo; y las variables no sean binarias, si bien se evita mediante su transformación en componentes principales (PCA).

Respecto a la generación de datos, los modelos basados en GAN introducen mayor variabilidad en la distribución de los conjuntos aumentados, creando datos de buena calidad. Sin embargo, el funcionamiento interno de discriminación puede implicar no crear muestras similares a aquellas reales que puedan ser consideradas '*outliers*'. Estas instancias a veces poseen información relevante para la predicción, y esta podría no ser replicada.

No se ha podido concluir si realmente, al realizar DA, los modelos de clasificación se benefician por una ganancia de información (creación de nueva información), o simplemente porque aumentar el número de instancias suaviza el aprendizaje de los modelos, aumentando su rendimiento. Sí se concluye que reducir el balanceo de clases es un aspecto fundamental, pues aumenta la sensibilidad de los clasificadores. Para aprovechar este efecto, el uso de un esquema meta-clasificador puede resultar efectivo, uniéndolo a la especificidad lograda mediante los conjuntos no aumentados. A este respecto, los modelos adaptativos con control del balanceo han resultado efectivos, aunque se ha demostrado que la aplicación de un método más simple y con un menor tiempo de ejecución, como es la adición de ruido con propósito de sobremuestreo presenta un rendimiento igual o mayor en múltiples problemas.

En conclusión, se ha puesto de manifiesto el efecto del DA en problemas bioinformáticos con conjuntos genómicos. Igualmente, se ha demostrado un rendimiento efectivo de los modelos de DL en tareas de predicción y generación de datos, por lo que es recomendable su empleo siempre que las características de la aplicación real (capacidad de cómputo, tiempo de ejecución, características de los datos, etc.) lo permitan.

UNIVERSIDAD DE MÁLAGA

10.3. Trabajo Futuro

Las conclusiones anteriormente comentadas nos llevan a escoger diferentes líneas futuras de investigación con la intención de extender estos trabajos a más problemas de la bioinformática, tanto en mejora de la predicción mediante análisis de datos ómicos, como a problemas de procesamiento de lenguaje natural. También se tiene en cuenta como línea de investigación las posibles mejoras de los modelos desarrollados y su comparación con un mayor número de métodos de DA del estado del arte. Teniendo en cuenta que ya ha sido expuesto durante todo el trabajo de tesis los fundamentos y avances en la aplicación del DA y su aplicación con datos ómicos, aprovecho esta última sección para profundizar en la línea de investigación del DA aplicado al procesamiento de lenguaje natural (NLP) en el campo médico/sanitario.

De forma similar a lo ocurrido en el procesamiento de imágenes por computador, la incorporación del aprendizaje profundo (DL) al NLP ha permitido mejorar significativa y sorprendentemente la eficacia de los algoritmos de clasificación y predicción aplicados en una gran variedad de problemas, como los sistemas conversacionales (Su et al., 2018), traducción automática de textos (Satija, 2018), análisis de sentimientos en redes sociales (Do et al., 2019) y, por supuesto, la clasificación automática de documentos (Mujtaba et al., 2019). Los modelos de DL comúnmente aplicados a NLP son las redes recurrentes, que se benefician de la estructuración temporal de los textos para aumentar el rendimiento respecto a otros modelos. Sin embargo, como en la mayoría de las posibles aplicaciones del mundo real, los datos disponibles para el entrenamiento de estos modelos profundos son a menudo escasos, lo que limita su rendimiento. Dentro del DA en NLP, en los últimos años ha crecido el interés en la aplicación de modelos generativos profundos, tales como las *Generative Adversarial Networks* (GAN) (Goodfellow et al., 2014a; Radford et al., 2015), modelos que han sido ampliamente estudiados en el presente trabajo de tesis. En esta área se encuentran los primeros trabajos en los que se generan únicamente pequeñas secuencias de palabras (Yu et al., 2016); estudios cada vez más avanzados en los que se generan frases y texto complejos con sentido (Lin et al., 2017; Rajeswar et al., 2017; Caccia et al., 2018; Haidar y Rezagholizadeh, 2019); se recrean diálogos reales con personas (Li et al., 2017); y se genera texto sin necesidad de un entrenamiento previo (dÁutume et al., 2019).

Por otra parte, los sistemas sanitarios públicos se enfrentan a numerosos retos entre los que destacan su sostenibilidad, la variabilidad en la práctica asistencial y la necesidad de mejorar la experiencia del paciente, entre otros. La medicina basada en la evidencia tiene como base operativa la investigación clínica, y su principal herramienta son los ensayos clínicos aleatorizados (ECA). No obstante, hoy en día la investigación en salud incluye también la recogida, compilación y análisis de los datos generados al margen de los ECA,

UNIVERSIDAD DE MÁLAGA

en lo que se conoce como "datos de la vida real" o "*Real World Data*" y que, en los últimos años, ha adquirido un creciente y renovado interés más allá de los clásicos estudios observacionales. La recogida generalizada, sistemática y exhaustiva de los datos realizada por los clínicos en registros electrónicos de salud, de forma común en la historia clínica electrónica (HCE), y su transformación en información útil y de valor para el clínico, aporta un cuerpo de conocimientos conocido como "*Real World Evidence*" (RWE) que enriquece y complementa la información proporcionada por los ECA.

La aplicación de los modelos generativos profundos ha traído nuevas oportunidades y grandes expectativas en el análisis de las HCE. Estos registros almacenan información de naturaleza heterogénea en una gran variedad de formatos, incluyendo documentos de texto abierto, como notas clínicas o informes radiológicos que contienen información relacionada con diagnósticos, tratamientos o procedimientos clínicos (Ribelles et al., 2010; Urda et al., 2013). La extracción de datos de estos registros puede aportar información importante para mejorar las prácticas médicas o para fomentar nuevos descubrimientos sobre enfermedades complejas, sin embargo, la naturaleza no estructurada de estos campos de texto abierto hace especialmente difícil la tarea de extracción automática de conceptos relevantes a partir de ellos, siendo además difícil tener un número equilibrado de muestras para los tipos de enfermedades en estudio. En este sentido, gracias a la aplicación de los modelos generativos, se pueden minimizar las desventajas de balanceo y del número reducido de textos disponibles. Esto ayudaría a obtener un mejor rendimiento en la transformación de texto clínico (escrito en lenguaje natural) en datos estructurados, posibilitando su uso en tareas como la planificación de tratamientos, investigación de enfermedades o toma de decisiones en la práctica clínica y en la gestión del sistema sanitario. Con el propósito de aplicar DA, se han publicado varios trabajos recientes en los que se utilizan GAN para sintetizar HCE (Baowaly et al., 2019; Che et al., 2017; Guan et al., 2018), e incluso para realizar ajustes en los sistemas de preguntas y respuestas para pacientes (Tang et al., 2018b).

Teniendo en cuenta los avances logrados en adaptación de modelos GAN para realizar DA con conjuntos de datos no estructurados, y los conocimientos del grupo de investigación en el tratamiento de textos clínicos y NLP, nos proponemos avanzar en esta línea teórica y su aplicación a tareas de PNL sobre textos clínicos en español. Partiendo de los espacios vectoriales resultantes de las transformaciones de los tokens de los textos originales, pasando por diferentes modelos de incrustación, se podrán entrenar modelos GAN para obtener un conjunto de datos aumentado que debería preservar la distribución estadística observada en el espacio vectorial original. Con estos conjuntos de datos aumentados, se podrá proceder con el entrenamiento de modelos *AI-for-NLP* en texto clínico en español para abordar diferentes tareas de NLP (codificación automática, reconocimiento de entidades, etc.).

UNIVERSIDAD DE MÁLAGA

UNIVERSIDAD
DE MÁLAGA

Parte VI

Apéndices

UNIVERSIDAD
DE MÁLAGA

UNIVERSIDAD
DE MÁLAGA

Apéndice A

Problemas estudiados

A.1. Capítulo 4: FNA

La Tabla A.1 muestra algunas características de los datos de referencia utilizados en los experimentos para comprobar el rendimiento del modelo de ajuste progresivo de ruido (FNA). Estos datos de referencia, tomados de la base de datos de UCI (Dua y Graff, 2017) y del conjunto PROBEN1 (Prechelt, 1994) que son utilizados con frecuencia en la literatura, consisten en datos del mundo real que representan problemas considerados de diagnóstico.

- card: El objetivo es predecir la aprobación o no aprobación de una tarjeta de crédito para un cliente por parte de un banco o institución similar. Los nombres y valores de las características están codificadas sin sentido para proteger la confidencialidad de los datos.

- gene: A partir de una ventana de 60 elementos de la secuencia de ADN (nucleótidos), el objetivo es predecir si el medio es un límite intrón/exón (un donante) o un límite exón/intrón (un aceptor), o ninguno de estos (Noordewier et al., 1991). Cada nucleótido, que es un atributo nominal de cuatro valores, se transforma en binario mediante dos entradas binarias (-1, 1).

- heart cleveland: El objetivo es predecir una enfermedad cardíaca, decidiendo si al menos uno de los cuatro vasos principales tiene un diámetro reducido en más del 50 % (Ahmad et al., 2017). Esta decisión binaria se toma en base a datos personales como edad, sexo, hábitos de tabaquismo u obesidad, descripciones subjetivas del dolor del paciente y resultados de varios exámenes médicos, como la presión arterial y los resultados del electrocardiograma. Este conjunto es la unión de cuatro conjuntos de datos, recogidos por la Cleveland Clinic Foundation, el Instituto Húngaro de Cardiología, el V.A. Medical Center Long Beach, y el Hospital Universitario de Zúrich.

UNIVERSIDAD DE MÁLAGA

- horse colic: El objetivo es predecir si un caballo puede o no sobrevivir a un cólico basándose en condiciones médicas recopiladas en un examen veterinario. El conjunto de datos fue creado por Mary McLeish y Matt Cecile de la Universidad de Guelp, Ontario, Canadá. Debido a la gran cantidad de datos no disponibles (NA), estos son codificados explícitamente mediante una entrada separada.

- pima (*pima indians diabetes*): El objetivo del conjunto de datos es predecir de forma diagnóstica si un paciente tiene diabetes o no, basándose en determinadas medidas de diagnóstico incluidas en el conjunto de datos (Smith et al., 1988). En particular, todos los pacientes son mujeres de al menos 21 años de ascendencia india Pima. Este conjunto de datos fue recolectado por el Instituto Nacional de Diabetes y Enfermedades Digestivas y Renales, Bethesda, Maryland, USA.

- thyroid: El objetivo es determinar si un paciente derivado es hipotiroideo (Quinlan et al., 1987). Presenta tres clases de diagnóstico, normal (no hipotiroideo), hiperfunción y funcionamiento subnormal. Debido a que el 92 por ciento de los pacientes no son hipertiroideos, un buen clasificador debe ser significativamente mejor que el 92 %. Conjunto recogido por el Dr. Quinlan del Instituto Garavan, Sydney, Australia.

- waveform: Este conjunto de datos fue generado con la intención de crear un conjunto complejo con 3 clases diferentes (Breiman et al., 1984). Teniendo tres tipos de ondas simples (con la amplitud a la derecha, a la izquierda y en el centro), cada muestra se encuentra generada mediante la combinación de dos formas de ondas. Cada combinación representa pues una clase en el problema.

- wdbc (*Wisconsin diagnostic breast cancer*): El objetivo es determinar si una muestra representa un conjunto de células cancerosas benigna o maligna. Las características se calculan a partir de una imagen digitalizada de una masa de células mamarias, y representan las propiedades de los núcleos celulares presentes en la imagen.

- breast cancer (*Wisconsin breast cancer*): Este conjunto de datos es el original a partir del cual se extrae el conjunto *wdbc*. El objetivo es pues determinar si una muestra representa un conjunto de células cancerosas benigna o maligna (Wolberg y Mangasarian, 1990). Las muestras presentes en el conjunto de datos fueron recopiladas por el Dr. Wolberg del Hospital de la Universidad de Wisconsin, Madison, Wisconsin, USA. Las características del conjunto de datos representan 9 propiedades de las células cancerosas, por ejemplo, el grosor del grupo, la uniformidad del tamaño y la forma de las células, o la cantidad de adhesión marginal.

UNIVERSIDAD DE MÁLAGA

Tabla A.1: Características de los nueve conjuntos de datos de referencia seleccionados para llevar a cabo los experimentos.

Conjunto	Características	Instancias	Clase Mayoritaria (%)
pima diabetes	8	768	65.10
breast cancer	9	699	65.52
waveform*	12	3353	50.58
heart cleveland	13	303	54.12
card	15	690	55.51
thyroid*	21	7200	92.58
wdbc	30	569	62.74
horse colic*	58	364	61.54
gene*	120	3175	51.90

En la Tabla A.1, las columnas muestran el nombre del conjunto de datos de referencia, el número de características e instancias, y el porcentaje de instancias que pertenecen a la clase mayoritaria con el propósito de mostrar el balanceo de las clases presentes en los datos de referencia. Destacar que algunos de los conjuntos de datos están marcados con un asterisco (*) para indicar el hecho de que los datos originales del problema contenían tres categorías de salida, y estos han sido reconvertidos en problemas binarios (dos clases). En la Tabla A.2 se muestran los conjuntos de datos con los porcentajes de las clases minoritarias inicialmente, el método de transformación en problema binario aplicado y el nuevo porcentaje que presenta la clase minoritaria creada. Específicamente, para los conjuntos de datos '*thyroid*', '*horse colic*' y '*gene*', las muestras de las clases minoritarias fueron agrupadas para formar una nueva clase que equilibrara el problema. En el caso del conjunto de datos de '*waveform*', la clase menos representada fue removida, porque el porcentaje de las tres clases era inicialmente bastante similar y las clases representan situaciones muy diferentes. Mencionar además que todas las variables de entrada fueron normalizadas para su tratamiento posterior en el proceso de aumento de datos. Aunque para estos experimentos fueron llevados a cabo sobre problemas binarios, todos los experimentos realizados y los métodos aplicados también se pueden utilizar en clasificación de múltiples clases de forma directa, simplemente utilizando neuronas de salida adicionales para las diferentes categorías de clase.

A.2. Capítulo 5: Modelo clasificador generativo

Se seleccionaron conjuntos de datos de referencia para utilizarlos en los experimentos para comprobar el rendimiento del aumento de datos con modelos generativos profundos (VAEs y GANs) utilizando el esquema del modelo clasificador generativo y las modificaciones de balanceo de datos. Los

UNIVERSIDAD DE MÁLAGA

Tabla A.2: Porcentaje de muestras en las clases minoritarias de los conjuntos de datos de referencia y el método de procesamiento utilizado para la reconversión a problema binario.

Conjunto	Clase 1	Clase 2	Método	Nueva Clase
thyroid	5.11	2.31	Agrupar	7.42
horse colic	24.18	14.28	Agrupar	38.46
gene	24.09	24.01	Agrupar	48.10
waveform	33.92	32.94	Remover	49.42

datos se tomaron de la base de datos UCI (Dua y Graff, 2017) y del conjunto PROBEN1 (Prechelt, 1994). Algunos de ellos ya fueron descritos en la sección A.1, el resto se explica a continuación.

- bal scale (*Balance Scale*): Este conjunto de datos se generó para modelar resultados experimentales psicológicos (Klahr y Siegler, 1978). Cada ejemplo se clasifica como tener la balanza inclinada hacia la derecha, inclinada hacia la izquierda o estar equilibrada. Los atributos son el peso izquierdo, la distancia izquierda, el peso derecho y la distancia derecha. La forma correcta de predecir la clase es encontrar el mayor entre (distancia izquierda * peso izquierda) y (distancia derecha * peso derecha), y si son iguales significa que está equilibrado.

- glass: El objetivo es clasificar tipos de vidrio para saber si han sido utilizados para la creación de ventanas de construcción, ventanas de vehículos, contenedores, vajillas o faros delanteros. Esta tarea está motivada por las necesidades forenses en la investigación criminal (Evett y Spiehler, 1989). Las características representan los resultados de un análisis químico del vidrio (definido en términos de su contenido de óxido) más el índice de refracción.

- ionosphere: El objetivo es clasificar retornos de radar en "buenos", si muestran evidencia de algún tipo de estructura en la ionosfera; y retornos "malos", si no lo hacen porque sus señales pasan a través de la ionosfera (Sigillito et al., 1989). Las señales recibidas fueron procesadas mediante una función de auto-correlación y las instancias describen mediante 2 atributos los pulsos recogidos de la señal electromagnética. Los datos de radar fueron recopilados por un sistema de 16 antenas de alta frecuencia en Goose Bay, Labrador, Canadá.

- iris: Este es quizás el conjunto de datos más conocido que se puede encontrar en la literatura sobre reconocimiento de patrones (Fisher, 1936). El conjunto de datos contiene 3 clases de 50 instancias cada una, donde cada clase se refiere a un tipo de lirio. El objetivo es pues

UNIVERSIDAD DE MÁLAGA

clasificar cada tipo de lirio a partir de las características recogidas de las plantas.

- non bal scale: El conjunto de datos es el mismo que el explicado anteriormente, Balance Scale. La diferencia se encuentra en la supresión de la clase menos representada. El conjunto presenta pues 576 instancias y se encuentra totalmente balanceado. El objetivo es predecir si la balanza se encuentra inclinada hacia la derecha o hacia la izquierda.

- seeds: El conjunto está compuesto por los valores de las propiedades geométricas de granos pertenecientes a tres variedades diferentes de trigo (Kama, Rosa y Canadian) (Charytanowicz et al., 2010). Para la extracción de estas propiedades se utilizó una técnica de rayos X y construir los siete atributos de valor real. Cada variedad de trigo examinada se encuentra compuesta de 70 muestras. Los estudios fueron realizados en el Instituto de Agrofísica de la Academia de Ciencias de Polonia en Lublin.

- sonar: El objetivo consiste en discriminar entre señales de sonar que rebotan en un cilindro de metal y las que rebotan en una roca aproximadamente cilíndrica (Gorman y Sejnowski, 1988). El conjunto de datos se compone pues de 111 muestras obtenidas de hacer rebotar las señales de la sonda en un cilindro de metal, y de 97 muestras obtenidas de hacerlas rebotar en roca. Cada muestra presenta 60 atributos que representa la energía dentro de una banda de frecuencia particular, integrada durante un cierto período de tiempo. Este conjunto de datos fue recolectado por el Dr. Sejnowski, ahora en la Universidad de California en San Diego en colaboración con el Dr. Gorman del Allied-Signal Aerospace Technology Center.

- soybean: El objetivo es reconocer 19 enfermedades diferentes de la soja (Michaiski y Chiiausky, 1980). La clasificación debe realizarse en función de una descripción del haba, por ejemplo, si su tamaño y color son normales; la planta, como el tamaño de las manchas en las hojas, si estas manchas tienen un halo, si el crecimiento de la planta es normal; e información adicional sobre la historia de la vida de la planta, por ejemplo, si se produjeron cambios en el cultivo en el último año o en los dos últimos años, si se trataron las semillas, cómo es la temperatura ambiente.

- wine: El objetivo es reconocer el tipo de vino analizado (Aeberhard et al., 1994). Estos datos son el resultado de un análisis químico de vinos cultivados en la misma región de Italia, pero derivados de tres cultivos diferentes. El análisis determina las cantidades de 13 componentes que se encuentran en cada uno de los tres tipos de vinos.

UNIVERSIDAD DE MÁLAGA

En la Tabla A.3 se muestran las características de los conjuntos de referencia utilizados en los experimentos. Las columnas muestran el nombre del conjunto de datos, el número de clases que presentan, el número de características e instancias, y la proporción de clases. Para esta última columna, en lugar de mostrar el porcentaje de instancias que pertenecen a cada clase, se utilizó una medida que calcula el balanceo en los datos (Ecuación A.1) basada en la entropía de Shannon (H). La medida se calcula a partir del número de instancias n en el conjunto de datos, el número de clases k y el tamaño de cada clase c_i. Un valor de balanceo igual a 1 significa que el conjunto está completamente equilibrado, mientras que los valores cercanos a 0 corresponden a conjuntos completamente desequilibrados. Una medida por debajo de 0.9 ya representa un conjunto desbalanceado, teniendo en un problema binario un valor de 70:30 para la proporción de clases.

$$\text{Balance} = \frac{H}{\log k} = \frac{-\sum_{i=1}^{k} \frac{c_i}{n} \log \frac{c_i}{n}}{\log k} \tag{A.1}$$

Tabla A.3: Características de los diecinueve conjuntos de datos de referencia seleccionados para llevar a cabo los experimentos.

Conjunto	Clases	Características	Instancias	Balanceo
bal scale	3	4	625	0.83
breast cancer	2	9	699	0.93
card	2	15	690	0.99
cleveland	2	13	303	0.99
gene	3	120	3175	0.93
geneN	3	60	3175	0.93
glass	6	9	214	0.84
horse colic	3	58	364	0.84
ionosphere	2	34	351	0.94
iris	3	4	150	1.00
non bal scale	2	4	576	1.00
pima diabetes	2	8	768	0.93
seeds	3	7	210	1.00
sonar	2	60	208	0.99
soybean	19	82	683	0.90
thyroid	3	21	7200	0.28
waveform	3	21	5000	0.99
wdbc	2	30	569	0.95
wine	3	13	178	0.99

Dos de los conjuntos de datos mostrados en la Tabla A.3 no son los originales de las bases de datos UCI y PROBEN1, sino que proceden de realizar ligeras transformaciones a conjuntos obtenidos de estas. La idea de

UNIVERSIDAD DE MÁLAGA

las modificaciones realizadas era aumentar el número de conjuntos de datos utilizados para la experimentación, y al mismo tiempo analizar problemas similares con diferente número de clases y estructura de balanceo. Los conjuntos de datos modificados son '*geneN*', derivado del conjunto de datos '*gene*' de PROBEN1, y el conjunto de '*non bal scale*' modificado a partir del conjunto de datos '*bal scale*' originalmente de la base de datos UCI. En el conjunto de datos '*gene*', cada nucleótido tomado en la ventana de secuencia de genes está representado por un par de entradas que dan como resultado 120 características codificadas de modo que el nucleótido C se codifica como (-1, -1), A como (-1, 1), G como (1, -1) y T como (1, 1). Hemos modificado esta codificación de modo que C se codifica como 0, A como 1, G como 2 y T como 3, lo que da lugar a un conjunto '*geneN*' que tiene 60 características. En el otro caso, el conjunto de datos de '*bal scale*' presenta tres clases con 4 características y 625 instancias, y está desbalanceado, teniendo dos clases principales con una representación del 46.08 % y una clase minoritaria con solo el 7.84 %. El conjunto de datos modificado '*non bal scale*' tiene sólo dos clases, las que son mayoritarias en '*bal scale*', de modo que las instancias se reducen a 576 y el conjunto de datos se vuelve completamente equilibrado. Por último, cabe mencionar que todos los conjuntos de datos de referencia se normalizaron antes de que se llevaran a cabo los experimentos.

Tabla A.4: Número de neuronas presentes en las capas ocultas de la red neuronal aplicada en los experimentos para los conjuntos de datos.

Conjunto	1º Capa (N)	2º Capa (M)	3º Capa (K)
bal scale	1024	1024	1024
breast cancer	1024	1024	128
card	512	64	1024
cleveland	1024	512	128
gene	1024	1024	128
geneN	256	64	1024
glass	1024	1024	1024
horse colic	512	128	1024
ionosphere	256	1024	64
iris	1024	512	128
non bal scale	1024	1024	1024
pima diabetes	1024	1024	1024
seeds	1024	512	128
sonar	64	64	1024
soybean	1024	1024	1024
thyroid	1024	1024	1024
waveform	1024	1024	1024
wdbc	512	512	1024
wine	1024	128	256

UNIVERSIDAD DE MÁLAGA

Se realizó una experimentación previa para elegir una arquitectura de red con el propósito de optimizar el rendimiento en predicción. Se probaron arquitecturas de 2, 3 y 4 capas ocultas, siendo elegida la arquitectura feedforward con 3 capas ocultas (N, M, K), debido a su rendimiento y la limitación de cómputo. Se optimizaron para esta arquitectura parámetros como la tasa de aprendizaje (rango 1e-4/1e-3) o la tasa de abandono (rango 0.1/0.7), y se realizó un lote de pruebas teniendo como neuronas posibles para cada capa oculta los valores (32, 64, 128, 256, 512 y 1024). Existía una limitación de cómputo para poder probar con 2048 neuronas en las capas. En la Tabla A.4 se muestra el número de neuronas seleccionado en las capas ocultas para cada conjunto de datos en la experimentación final.

A.3. Capítulo 6: Predicción de eventos en cáncer

Los datos de referencia utilizados en los experimentos para comprobar la eficiencia del aumento de datos con modelos basados en CGAN, incluyendo el modelo adaptativo y las modificaciones de balanceo, están disponibles gratuitamente y se pueden descargar en el sitio web *The Cancer Genome Atlas* (TCGA) (Tomczak et al., 2015), proporcionados por el *International Cancer Genome Consortium* (ICGC). Los datos corresponden a pacientes vinculados a 18 tipos diferentes de cáncer: carcinoma de vejiga (blca), carcinoma de mama (brca), adenocarcinoma de colon (coad), glioblastoma multiforme (gbm), carcinoma de células escamosas de cabeza y cuello (hnsc), cánceres de riñon (kipan), carcinoma de células claras de riñón (kirc), glioma cerebral de grado inferior (lgg), carcinoma hepatocelular de hígado (lihc), adenocarcinoma de pulmón (luad), carcinoma de células escamosas de pulmón (lusc), carcinoma de ovario (ov), adenocarcinoma de próstata (prad), melanoma cutáneo (skcm), adenocarcinoma de estómago (stad), carcinoma de estómago y esofágico (stes), adenocarcinoma de tiroides (thca) y carcinoma de endometrio (ucec). Estos se describen en la sección 6.1 del Capítulo 6.

Las instancias de estos conjuntos representan pacientes afectados de cáncer, y para cada paciente los datos contienen un total de 20531 variables que corresponden al nivel de expresión de cada gen, por lo que los conjuntos de datos son perfiles de expresión génica de RNA-Seq después de aplicar procedimientos de preprocesamiento para la corrección de lotes y normalización RSEM. Adicionalmente, se realizó una transformación logarítmica (log2) de los niveles de expresión en los datos, para aproximarlos a una distribución normal para su uso con los modelos predictivos, y se realizó una selección de características reduciendo el número de genes y eliminando aquellos que no agregan un valor predictivo suficiente. Para realizar un análisis de predicción, se recopiló la información sobre el estado vital de cada paciente después de finalizar el estudio, información que también está disponible gratuitamente en TCGA. El estado vital supone por tanto la etiqueta presente en los datos

UNIVERSIDAD DE MÁLAGA

y el objetivo a predecir. Si la etiqueta es igual a 0 representa que el paciente estaba "vivo" al finalizar el estudio, y si es igual a 1 representa que el paciente no sobrevivió al cáncer ("muerto"), por lo que los modelos de predicción funcionan en un problema de clasificación binaria.

La Tabla A.5 muestra algunas características de los conjuntos estudiados. Las columnas muestran el nombre del conjunto, el número de variables presentes en el conjunto después de realizar la selección de características, el número de instancias, la proporción de clases y el método de selección de características (FS) utilizado (selección mediante el método LASSO o mediante la importancia Gini con Random Forest). En lugar de mostrar el porcentaje de instancias que pertenecen a cada clase, se opta por mostrar una medida de balanceo basada en la entropía de Shannon. Esta medida se calcula siguiendo la fórmula mostrada en la Ecuación A.2, a partir del número de instancias n en el conjunto de datos, el número de clases k y el tamaño de cada clase c_i. Un valor de balanceo igual a 1 significa que el conjunto está completamente equilibrado, mientras que los valores cercanos a 0 corresponden a conjuntos completamente desequilibrados.

$$\text{Balance} = \frac{H}{\log k} = \frac{-\sum_{i=1}^{k} \frac{c_i}{n} \log \frac{c_i}{n}}{\log k} \tag{A.2}$$

Tabla A.5: Características de los dieciocho conjuntos de datos de referencia seleccionados para llevar a cabo los experimentos.

Conjunto	Características	Instancias	Balanceo	Método FS
blca	114	427	0.99	LASSO
brca	74	1212	0.64	LASSO
coad	12	191	0.71	RF
gbm	19	171	0.73	RF
hnsc	10	566	0.99	LASSO
kipan	102	1020	0.83	LASSO
kirc	88	606	0.92	LASSO
lgg	57	242	0.98	LASSO
lihc	11	423	0.96	LASSO
luad	13	344	0.96	LASSO
lusc	18	552	0.99	LASSO
ov	33	307	0.97	LASSO
prad	27	550	0.13	RF
skcm	21	473	1.00	LASSO
stad	3	450	0.96	LASSO
stes	47	646	0.97	RF
thca	21	568	0.22	RF
ucec	31	201	0.67	RF

UNIVERSIDAD DE MÁLAGA

Para el procesamiento de los datos fue necesario utilizar dos métodos de selección de características debido a que, al aplicar el método LASSO, este seleccionó un número insuficiente de características para algunos de los conjuntos de datos de referencia estudiados, obteniendo porcentajes de aciertos inaceptables y por debajo de la capacidad de predicción obtenida con los conjuntos originales. Esta problemática con el método LASSO puede deberse al hecho de que los conjuntos de datos correspondientes presentan un reducido número de muestras en conjunción con un gran desequilibrio de las clases (balanceo < 0.75). La excepción es el conjunto de datos brca, que presenta un marcado desequilibrio (0.64), pero presenta un mayor número de muestras (1212), lo que ha podido ayudar a LASSO a extraer mejor información del conjunto. Por estas razones, la selección de características para múltiples conjuntos de datos se realizó midiendo la importancia de Gini del método de Random Forest, que se ve menos afectado por la propiedad del desequilibrio al realizar la selección de características.

A.4. Capítulo 7: Niemman-Pick C1

El conjunto de datos utilizados para comprobar el rendimiento de diferentes técnicas de aumento de datos en la predicción de la enfermedad Niemman-Pick Tipo C1, ha sido recogido por un estudio clínico con sede en Reino Unido. El conjunto consta de 13 pacientes NPC1 no tratados (rango de edad 2.7 - 30 años, 6 hombres / 7 mujeres) y 47 portadores heterocigotos parentales correspondientes (rango de edad 25 - 62 años, 19 hombres / 28 mujeres). Los pacientes NPC1 recogidos corresponden únicamente a pacientes que no recibieron ningún agente terapéutico, seleccionados cuidadosamente para evitar posibles complicaciones en el procesamiento y predicción, derivadas de la presencia de resonancias urinarias ^{1}H NMR atribuibles a dichos fármacos y sus metabolitos, en los perfiles metabólicos urinarios explorados. Los datos para este estudio fueron recolectados con el preceptivo consentimiento informado y previamente aprobados por el Comité de Ética en Investigación correspondiente (06/MRE02/85). Las muestras de orina donadas por los participantes fueron almacenadas a -80° C. Cuando estuvieron listas para el análisis, estas se descongelaron y centrifugaron para eliminar las células y los residuos (5.000 rpm durante un período de 10 minutos), y los volúmenes de 0.60 ml de los sobrenadantes derivados se mezclaron con 0.07 ml de óxido de deuterio (^2H_2O). A continuación, estas mezclas de muestras se transfirieron a tubos de NMR de 5 mm de diámetro.

Se empleó un espectrómetro Bruker Avance AV-600 en las instalaciones de la Universidad Queen Mary de Londres, Londres, Reino Unido. Este espectrómetro se utilizó para el análisis de pulso único ^{1}H NMR de las muestras de orina, funcionando para ello a una frecuencia de 600.13 MHz en modo de detección en cuadratura y una temperatura de 298 K. Este proceso se des-

UNIVERSIDAD DE MÁLAGA

cribe con mayor detalle en Ruiz-Rodado et al. (2014). La señal H_2O/HOD (δ = 4.80 ppm) se eliminó mediante el desacoplamiento controlado durante el retardo entre pulsos. Los valores de desplazamiento químico se referenciaron internamente a las resonancias obtenidas del grupo metilo del acetato (s, δ = 1.920 ppm), alanina (d, δ = 1.487 ppm), lactato (d, δ = 1.330 ppm) y creatinina (>NCH3 s, δ = 3.030 ppm). Las resonancias de los metabolitos presentes en los espectros adquiridos se asignaron de forma rutinaria mediante una consideración completa de los valores de desplazamiento químico, los patrones de acoplamiento y las constantes de acoplamiento, y también de fuentes bibliográficas; siendo cotejadas con la *Human Metabolome Data Base* (HMDB) (Wishart et al., 2018). Se utilizaron técnicas espectroscópicas de correlación (COSY) unidimensional y bidimensional, y correlación total (TOCSY), para confirmar las asignaciones urinarias, junto con el "*spiking*" de muestras con pequeños volúmenes (μl) de aproximadamente 5.00 mmol/L de soluciones estándar de biomoléculas auténticas preparadas en tampón fosfato de 0.10 M (pH 7.00).

Los valores de las variables (*buckets*) fueron generados mediante el empleo de macroprocedimientos para *line broadening*, *zero filling*, transformación de Fourier, corrección de fase y corrección de base; junto con la aplicación posterior de una macro separada para la subrutina de procesamiento de "*intelligent bucketing*". Todos los procedimientos se realizaron utilizando el paquete software ACD/Labs Spectrus Processor 2012 (ACD/Labs, Toronto, Ontario, Canadá M5C 1T4). Esta estrategia ISB aseguró que todos los bordes de las variables presentados no coincidieran con los máximos de resonancia de ^{1}H NMR y, por lo tanto, este enfoque elude la división de señales en regiones integrales separadas. Para este propósito, se seleccionó un *bucket* de 0.04 ppm de ancho con un factor de holgura del 50 %. De los espectros ^{1}H NMR urinarios adquiridos se eliminaron los ISB espectrales que contenían señales atribuibles al etanol (δ = 1.22-1.24 (t) y 3.63-3.67 ppm (q)) y que fueron detectados en espectros adquiridos en un pequeño número de muestras de orina de control de portadores heterocigotos, y también las atribuibles a la urea (*broad*, δ = 5.59-5.99 ppm) y la señal residual H_2O/HOD (δ = 4.65-5.16 ppm), a través de la irradiación secundaria descrita anteriormente. Antes de realizar los análisis, se realizó un auto-escalado en las columnas (variable metabólica) para todas las muestras con los perfiles de ^{1}H NMR.

Las muestras atípicas se eliminaron de los conjuntos de datos original con los perfiles de ^{1}H NMR urinario después de realizar un análisis de componentes principales PC3 *versus* PC2 *versus* PC1. Estos incluyeron muestras de clasificación de control de portadores heterocigotos en las que se detectaron metabolitos de sulfato y glucurónido de paracetamol mediante análisis de ^{1}H NMR (es decir, señales bastante prominentes centradas en δ = 7.13 y 7.35 ppm (glucurónido) y δ = 7.31 y 7.45 ppm (sulfato), junto con sus resonancias de función acetamido-NH-CO-CH_3 (δ = 2.15-2.17 ppm). Tras todos

UNIVERSIDAD DE MÁLAGA

estos análisis y procedimientos previos, el conjunto de datos consta de 60 muestras que presentan un total de 33 variables predictoras de metabolitos asignadas por ^{1}H NMR, cuya distribución se resume en la Tabla A.6.

Grupo	**Muestras**	**Porcentaje**
Paciente NPC1	13	21.67 %
Portador heterocigoto	47	78.33 %

Tabla A.6: Distribución de los grupos en el conjunto de datos ^{1}H NMR.

A.5. Capítulo 8: Subtipo molecular del cáncer

El conjunto de datos utilizado para comprobar el rendimiento de diferentes técnicas de aumento de datos en la predicción del subtipo molecular del cáncer, ha sido proporcionado por el Hospital Regional Universitario Carlos Haya y en el Hospital Universitario Virgen de la Victoria de Málaga, España. Estos pertenecen a un estudio clínico de un grupo de 200 pacientes con cáncer de mama pertenecientes a la provincia de Málaga, diagnosticadas y tratadas entre diciembre de 2008 y febrero de 2013 en las diferentes unidades de Oncología Médica de los hospitales Regional Universitario de Málaga, Comarcal de Antequera, Comarcal de Vélez Málaga y Costa del Sol. La mayoría de estas pacientes tuvieron un seguimiento máximo de 5 años, y el punto de corte para el análisis de datos fue con fecha de 13 de Julio del 2015. Todas las participantes otorgaron su consentimiento para que sus datos fueran recabados para diferentes estudios, siendo promovidos por la Sección de Oncología Médica, el Servicio de Inmunología del Hospital Carlos Haya, la Fundación IMABIS y por el Instituto de Investigación Biomédica de Málaga (IBIMA). Para mayor seguridad de la confidencialidad de estas pacientes, los datos tratados se encuentran totalmente anonimizados. Los criterios de la inclusión eran, mujer diagnosticada de cáncer de mama, de origen español y procedente de la provincia de Málaga, y con tiempo transcurrido desde el diagnóstico a la inclusión inferior a 1 año.

Si bien en el estudio original el conjunto de datos estaba integrado por 200 pacientes afectadas por cáncer de mama, para nuestro estudio fue necesario reducir esta cifra a 165, con lo cual se pierde una parte del potencial de los datos. Aun así, gracias a la buena calidad de los datos restantes, los resultados obtenidos son fiables. Esta reducción viene dada por la presencia de ejemplos para los cuales no está disponible alguna de las variables moleculares, señalados como NA (*Not Available*). Concretamente, hay presentes 487 datos no disponibles de 42000 en total, es decir, más del 10 %. Por ello y ante la imposibilidad de utilizar conjuntos de datos con valores no disponibles en algunas de las implementaciones de los modelos de aprendizaje computacio-

UNIVERSIDAD DE MÁLAGA

nal a aplicar, se decidió prescindir de los ejemplos que presentara algún valor no disponible, en lugar de sustituir estos valores, ya que se podrían obtener resultados erróneos o falsos.

Concretamente, el conjunto de datos utilizado consta de muestras con la expresión genética de las pacientes y el subtipo del cáncer que presentan. La expresión genética consta en este caso de variables moleculares que se encuentran codificadas en el complejo mayor de histocompatibilidad (MHC, acrónimo del inglés *Major Histocompatibility Complex*). El MHC se trata de un sistema de genes ligados cuyo papel principal es el control de la expresión de moléculas de la superficie celular que actúan como marcadores de la respuesta inmune. Se ha demostrado que el MHC está presente en todos los animales vertebrados (Klein, 1987).

Variante	Pacientes	Porcentaje
HLADRB-1		
1	35	21.21 %
3	39	23.64 %
4	32	19.39 %
6	1	0.61 %
7	45	27.27 %
8	8	4.85 %
9	1	0.61 %
10	13	7.88 %
11	39	23.64 %
12	1	0.61 %
13	38	23.03 %
14	11	6.67 %
15	38	23.03 %
16	11	6.67 %
MIC-A		
4	26	15.76 %
5	31	18.79 %
5.1	61	36.97 %
6	95	57.58 %
9	34	20.61 %

Variante	Pacientes	Porcentaje
HLA-B		
7	29	17.58 %
8	19	11.52 %
13	4	2.42 %
14	19	11.52 %
15	24	14.55 %
18	36	21.82 %
27	10	6.06 %
35	30	18.18 %
37	3	1.82 %
38	7	4.24 %
39	1	0.61 %
40	12	7.27 %
41	4	2.42 %
42	1	0.61 %
44	42	25.45 %
45	10	6.06 %
47	2	1.21 %
49	9	5.45 %
50	5	3.03 %
51	22	13.33 %
52	7	4.24 %
53	6	3.64 %
55	4	2.42 %
57	6	3.64 %
58	1	0.61 %
67	1	0.61 %

Tabla A.7: Distribución de las variables moleculares HLADBR-1, MIC-A y HLA-B.

Las variables moleculares HLA-B, HLADBR-1 son dos variantes de los antígenos leucocitarios humanos (HLA), los cuales están formados por mo-

UNIVERSIDAD DE MÁLAGA

léculas presentes en la superficie la mayoría de las células de un individuo, incluidos los glóbulos blancos o leucocitos en la sangre. Las variantes de MIC-A son genes pertenecientes al mismo grupo que los HLA y que tienen significativa importancia en la respuesta inmune no específica de los individuos. Se puede observar en la Tabla A.7 como para HLADBR-1 la variante más presente es la nº. 7, en HLA-B son la nº. 44 y 18, y en MIC-A es la A6 con más de un 50 % de presencia entre las pacientes. Que haya variantes poco presentes puede significar que, o bien no van a ser utilizadas por los modelos por baja significancia, o bien serán clave a la hora de predecir el subtipo. También hay que tener en cuenta que las variantes no son excluyentes, es decir, la presencia de una no implica que otra no esté presente.

Se observan en la Tabla A.8 las variantes de CTLA-4, receptores situados en los linfocitos T y encargados de la inmunidad; los genes TAP, genes transportadores de péptidos antigénicos; el receptor tumoral NKG2DT; y los genes LMP, que están relacionados con el proteosoma. Se puede ver que en la mayoría de estas variables, hay una variante cuya presencia es mucho mayor que el resto, y una variable con poca presencia.

Variante	Pacientes	Porcentaje	Variante	Pacientes	Porcentaje
CTLA-4-49			TAP1.1		
AA	87	52.73 %	AA	118	71.52 %
AG	69	41.82 %	AG	44	26.67 %
GG	9	5.45 %	GG	3	1.82 %
CTLA-4-1661			TAP1.2		
AA	113	68.48 %	AA	122	73.94 %
AG	51	30.91 %	AG	42	25.45 %
GG	1	0.61 %	GG	1	0.61 %
NKG2DT72A			TAP2.1		
AA	4	2.42 %	AA	5	3.03 %
AG	50	30.30 %	AG	43	26.06 %
GG	111	67.27 %	GG	117	70.91 %
LMP2			TAP2.2		
CC	93	56.36 %	AA	91	55.15 %
CT	59	35.76 %	AG	58	35.15 %
TT	13	7.88 %	GG	16	9.70 %
LMP7			TAP2.3		
AA	3	1.82 %	CC	16	9.70 %
CA	39	23.64 %	CT	58	35.15 %
CC	123	74.55 %	TT	91	55.15 %

Tabla A.8: Distribución de las variables moleculares CTLA-4, NKG2DT, LMP y TAP.

Los subtipos moleculares del cáncer de mama se pueden dividir en dos grandes grupos según la positividad que presenta para el receptor de estrógeno. Un grupo son las neoplasias de grado bajo, o tumores receptores de

UNIVERSIDAD DE MÁLAGA

estrógeno positivo, que se dividen a su vez en distintos tipos denominados luminales. Estos subtipos moleculares son los subtipos menos agresivos y con un mayor ratio de supervivencia. El otro grupo son las neoplasias de grado alto, o tumores receptores de estrógeno negativo, que se dividen principalmente en basales y sobre expresión de HER2.

En la Tabla A.9 se muestra la presencia de los subtipos moleculares en el conjunto de datos estudiado. Hay que tener en cuenta la división realizada entre ambos grupos según la positividad para el receptor de estrógeno, ya que en el caso de realizar una clasificación binaria se toman como clases los subtipos de grado bajo (Luminal) frente a los subtipos de grado alto (No Luminal). Se puede observar que el subtipo molecular más presente en el conjunto de datos es el Luminal B/Her2 negativo con un 36.97 % de las muestras, seguido del Luminal A con un 32.73 %. Al contrario, los subtipos Triple negativo y Her2 son los menos representados en el conjunto.

Subtipo	Pacientes	Porcentaje
Luminal	135	81.82 %
Luminal A	54	32.73 %
Luminal B negativo	61	36.97 %
Luminal B positivo	20	12.12 %
No Luminal	30	18.18 %
Her2 positivo puro	14	8.48 %
Triple negativo	16	9.70 %

Tabla A.9: Distribución del subtipo molecular en los datos.

Debido a la alta dimensionalidad del conjunto de datos, se realiza un proceso de selección de características previo a la experimentación. De este modo, no solo es probable la mejora de la calidad de los modelos a la hora de la predicción y la aplicación de métodos basados en GAN, sino también se reduce la complejidad y el tiempo de cómputo de los algoritmos profundos que son computacionalmente costosos. El método elegido para realizar la selección es Random Forest (Breiman, 2001), con el cual se obtiene una ordenación de los atributos según la importancia Gini calculada a través de la creación de un número elevado de árboles de decisión.

Haciendo referencia en primer lugar a la aplicación en el enfoque del problema binario, se muestra en la Figura A.1 la ordenación de las 30 variables moleculares del conjunto de datos del estudio, haciendo una distinción sobre las distintas variantes, que obtienen un mayor valor de importancia (izquierda) y las 30 variables que obtienen un menor valor de importancia (derecha). En el caso de las variables con mayor importancia, destacar que solo hay 5 que superan un valor de 1.0, mientras que la gran mayoría presentan un valor por debajo de 0.5, presentando cierta variación en cuanto a

UNIVERSIDAD DE MÁLAGA

las variables moleculares presentes. Si se observan las variables con menor importancia Gini, la totalidad de las variables tienen importancia nula, es decir, en los árboles de decisión creados en el bosque no influyen en la clasificación de los ejemplos. Todas ellas son además variables moleculares HLA-B y HLA-DRB1, si bien la escasa importancia de estas variables se puede dar debido a su mayor número en el conjunto de datos, representando más del 50 % de las variables. En definitiva, tras aplicar este método, el número de variables seleccionadas para el tratamiento del problema binario ha sido de 18, todas aquellas que superan un valor de 0.6 de importancia, que es un valor suficientemente representativo.

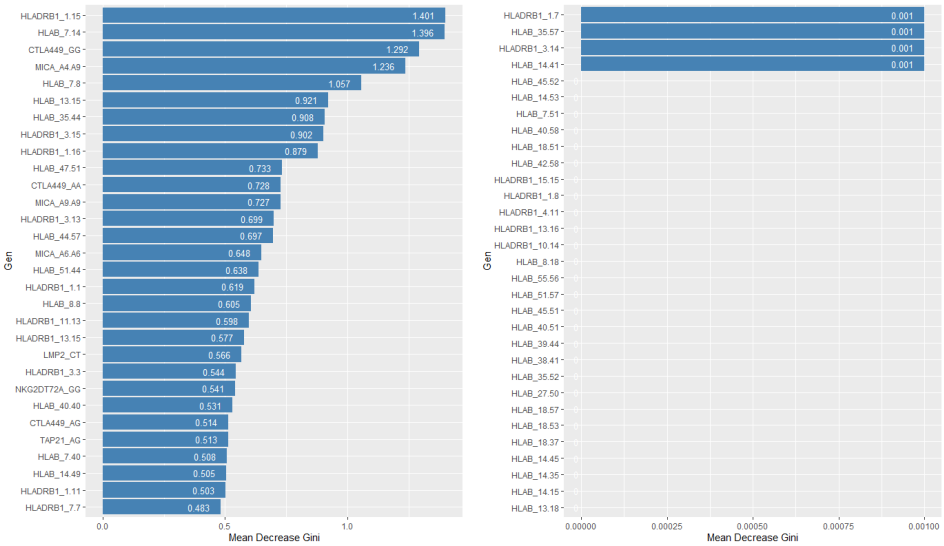

Figura A.1: Valores la importancia Gini de las variables aplicando Random Forest al enfoque binario, mostrando una selección de los 30 mayores valores (izquierda) y los 30 menores (derecha).

En segundo lugar, para el tratamiento multiclase de la predicción del subtipo, se muestra de igual modo en la Figura A.2 la ordenación de las 30 variables moleculares del con un mayor valor de importancia (izquierda) y las 30 variables que obtienen un menor valor de importancia (derecha). Realizando una pequeña comparación con las importancias y las variables obtenidas para el enfoque binario, en este caso, al ser un problema más complejo y al ser necesaria una distinción más concreta del subtipo, los valores de importancia obtenidos son mayores. Destacar especialmente la variable molecular MICA_A6, que presenta una importancia de 2.165, siendo este valor 0.3 puntos mayor que segunda variable más importante. También es importante destacar la presencia de las variantes de las variables moleculares CTLA-4 (CTLA-4-1661 y CTLA-4-49) entre las 30 primeras, donde CTLA-4-49 tiene los tres genotipos posibles (AA, AG, GG) dentro de este grupo. Se

UNIVERSIDAD DE MÁLAGA

puede observar entonces como, además del hecho probado de que estos dos polimorfismos de nucleótido simple se correlacionan con un aumento en la probabilidad de desarrollar cáncer (Bond et al., 2004), también pueden estar muy relacionados con el subtipo del mismo. Tras aplicar el procesamiento de selección, el número de variables seleccionadas para el tratamiento del problema multiclase ha sido de 59, y al igual que en el caso binario, se han seleccionado aquellas que superan un valor de 0.6 de importancia.

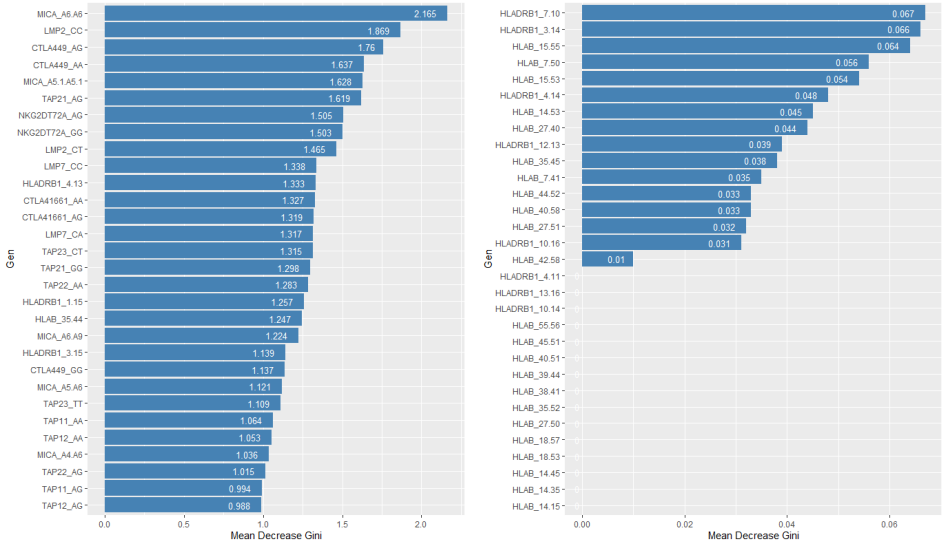

Figura A.2: Valores la importancia Gini de las variables aplicando Random Forest al enfoque multiclase, mostrando una selección de los 30 mayores valores (izquierda) y los 30 menores (derecha).

A.6. Capítulo 9: Esquema Meta Clasificador

Para evaluar el rendimiento de la aplicación de aumento de datos en el esquema meta clasificador, se seleccionaron 7 conjuntos de datos desbalanceados conocidos en la literatura, y 5 conjuntos de datos desbalanceados que corresponden a la expresión génica de pacientes que presentan diferentes tipos de cáncer. Concretamente, se seleccionaron los conjuntos de datos '*pima*', '*ecoli*', '*segment*' y '*satimage*' del repositorio de aprendizaje automático UCI (Dua y Graff, 2017), el conjunto '*newthyroid*' se seleccionó del repositorio KEEL (Alcalá-Fdez et al., 2011) y el conjunto de datos '*oil*' fue proporcionado por Robert Holte (Kubat et al., 1998). El conjunto '*pima*' se encuentra descrito en la sección A.1, el resto se explica a continuación.

- ecoli: El objetivo es predecir la localización de la proteína en bacterias Gram-negativas (Nakai y Kanehisa, 1991, 1992), dada una escasa in-

UNIVERSIDAD DE MÁLAGA

formación de la secuencia de aminoácidos. En el conjunto original hay 7 clases que identifican la localización (citoplasma, membrana interna sin secuencia de señal, membrana interna - secuencia de señales no escindible, membrana interna - secuencia de señal escindible, membrana exterior, lipoproteína membrana externa, lipoproteína membrana interna, perisplasma). Para tener un problema binario se transforma el objetivo a identificar si la proteína se encuentra en el periplasma o no (Douzas et al., 2018). Los estudios fueron realizados en el Instituto de Biología Molecular y Celular, Universidad de Osaka, Japón.

- newthyroid (*thyroid glad*): El objetivo es determinar si un paciente presenta hipertiroidismo (Coomans et al., 1983). En el conjunto original se tienen tres clases correspondientes a eutiroidismo, hipotiroidismo o hipertiroidismo, pero para tener un problema binario, las dos primeras clases se unieron (Douzas et al., 2018). Las muestras representan los valores obtenidos de diferentes pruebas clínicas.

- oil: El objetivo es detectar derrames de hidrocarburos a través de imágenes satelitales (Kubat et al., 1998). Esta tarea viene motivada por la creación de un sistema capaz de ofrecer una alerta temprana de vertidos dado el significativo impacto ambiental. Las instancias son imágenes de satélite procesadas en forma de vector de características de longitud fija para cada región sospechosa.

- satimage (*statlog, Landsat Satellite*): El conjunto de datos consta de los valores multiespectrales de los píxeles en regiones 3x3 en una imagen de satélite. El objetivo es predecir la clasificación asociada con el píxel central en cada vecindario. Concretamente, cada instancia contiene los valores de los píxeles en cuatro bandas espectrales obtenidas de un satélite Landsat MSS, para cada uno de los 9 píxeles de la región 3x3. Originalmente el conjunto de datos presenta 6 clases, pero se seleccionó la clase con menor número de instancias como la clase minoritaria y se unieron el resto de las clases en una (Provost y Fawcett, 2001). Los datos originales de Landsat se generaron a partir de datos comprados a la NASA por el Centro Australiano de Percepción Remota, y fueron procesados por el Dr. Srinivasan de la Universidad de Strathclyde, Glasgow, Reino Unido.

- segment (*image segmentation*): El objetivo es predecir qué representa una imagen segmentada. Las instancias originales están recopiladas de una base de datos de imágenes al aire libre, y están segmentadas para crear una clasificación para cada píxel. Cada instancia es una región de 3x3 representada por diferentes características como el contraste, la intensidad, etc. Las imágenes originalmente representan césped, camino, ventana, ladrillo, cemento, follaje o cielo. Para tener un problema bi-

UNIVERSIDAD DE MÁLAGA

nario, se reconvirtieron las clases del conjunto a 'cemento *versus* resto' (segment1) y 'ventana *versus* resto' (segment2). Las imágenes fueron recopiladas y procesadas por la Dra. Carla Brodley del Grupo de Visión de la Universidad de Massachusetts, Boston, Massachusetts, USA.

Los 5 conjuntos médicos están disponibles gratuitamente en el sitio web *The Cancer Genome Atlas* (TCGA) (Tomczak et al., 2015), proporcionados por el *International Cancer Genome Consortium* (ICGC). Los conjuntos de datos corresponden al glioblastoma multiforme (gbm), adenocarcinoma de colon (coad), carcinoma endometrioide de cuerpo uterino (ucec), adenocarcinoma de tiroides (thca) y adenocarcinoma de próstata (prad). Los conjuntos de datos de cáncer originales son perfiles de expresión génica de RNA-Seq que contienen los valores de 20531 genes. Se realizó una transformación logarítmica de los niveles de expresión en los datos, aproximándolos a una distribución normal. Además, se aplicó un proceso de selección de características utilizando LASSO (Tibshirani, 1996) y la importancia Gini de Random Forest (Breiman, 2001), reduciendo el número de genes. El estado vital de cada paciente es la etiqueta de clase presente en los datos y el objetivo a predecir. Estos conjuntos se describen en la sección 6.1 del Capítulo 6.

La Tabla A.10 muestra la información detallada de los 12 conjuntos de datos de referencia estudiados, incluidos los nombres, el número de características, el número de instancias del conjunto de datos completo, el número de muestras minoritarias, el número de muestras mayoritarias, el equilibrio basado en la entropía de Shannon y el ratio de desequilibrio (IR).

Tabla A.10: Características de los doce conjuntos de datos de referencia seleccionados para llevar a cabo los experimentos.

Conjuntos	Caract.	Instancias	Min.	Mayor.	Balanceo	IR
pima	19	768	268	500	0.93	1.87
newthyroid	5	215	35	180	0.64	5.14
ecoli	7	336	52	284	0.62	5.46
segment1	16	2310	330	1980	0.59	6.00
segment2	16	2310	330	1980	0.59	6.00
satimage	36	6430	625	5805	0.46	9.29
oil	48	937	41	896	0.26	36.74
gbm	19	171	35	136	0.73	3.89
coad	12	191	38	153	0.71	4.03
ucec	31	201	35	166	0.67	4.74
thca	21	568	20	548	0.22	27.40
prad	27	550	10	540	0.13	54.00

UNIVERSIDAD DE MÁLAGA

UNIVERSIDAD
DE MÁLAGA

Apéndice B

Thesis Summary in English

B.1. Introduction

Deep learning (DL) models have obtained greater progress and success compared to traditional learning methods in the last decade. DL is currently the state of the art in computer vision and natural language processing. However, DL models present an important disadvantage, they require thousands of instances to achieve a good level of success. In the modern world, this should not be a problem in most fields of application. Nevertheless, data acquisition remains a difficult and expensive process in bioinformatics. The data augmentation (DA) strategy can avoid this limitation, increasing the number of available samples by applying transformations to the dataset or generating new synthetic samples, but its application in problems with 'non-structured' biomedical datasets is a challenging task.

This research is integrated into the projects developed in the Computational Intelligence and Biomedicine group of the University of Malaga. The research team has a consolidated trajectory in the last 15 years in the application of artificial intelligence in biomedicine. The Plan Nacional de I+D+i and the Junta de Andalucía have supported the group through the execution of different projects, all of them focused on bioinformatics. The group has made progress in the design of neuro-computational learning algorithms for cancer survival analysis (TIN2005-02984), constructive algorithms of artificial neural networks for information processing and data mining in oncology (TIN2008-04985, TIC-4026), and bio-inspired intelligent systems applied to personalized medicine (TIN2010-16556). An indication of the quality of the previous work developed by the research team is the commitment shown by different entities/companies in the transfer of results and technology derived from the execution of the research projects mentioned. The group is an active member of the UGC Oncología Intercentros (UGCOI), the Biomedical Research Institute of Málaga (IBIMA) and the Andalusian Public Foundation for Research in Biomedicine and Health in Málaga (FIMABIS). One of the

UNIVERSIDAD DE MÁLAGA

greatest achievements of the research group is the Galén information system (Ribelles et al., 2010), developed in collaboration with expert clinical staff. Gálen currently stores information relating to more than 40000 cancer patients (1978 - 2020) in the province of Málaga. The ICB group is responsible for the application of AI algorithms to these bioinformatics data within the cancer molecular biology group of the UGCOI. Due to the success of DL, the ICB group has worked on the design and adaptation of DL algorithms for their application to problems in the field of biomedicine under a project of the Plan Nacional I+D+i (TIN2017-88278). We have focused on two strategies to improve the application of these DL models to bioinformatics data, transfer learning and data augmentation. This last approach is the central theme of this Ph.D. thesis.

B.1.1. Motivation and Objective

Artificial neural networks are capable of learning any nonlinear function (Hornik et al., 1989), but their performance was limited for a long time compared to other more interpretable machine learning (ML) methods. Nevertheless, important advances have led during the last decade to the rejuvenation of neural networks with the DL. An unprecedented amount of data is being generated and collected in the modern world. In addition, there has been an exponential increase in computing power. This, together with the development of complex optimization algorithms, have made deep networks stand out as the ideal model for processing large and complex data.

DL has become the state-of-the-art for many ML application areas, so the industry is investing billions of dollars in these artificial intelligence techniques. The convolutional neural networks (CNN) dominate the main areas in the field of computer vision (Ren et al., 2015; Szegedy et al., 2015; Yue et al., 2016) and the recurrent neural networks (RNN) present the best performance in a wide range of tasks in natural language processing (NPL) (Graves et al., 2013; Sutskever et al., 2014; Yao et al., 2019). In a socially relevant field such as biomedicine, DL also shows promising potential to exploit the complex relationships present in large-scale biological data. The performance is evident in certain tasks, such as DNA and RNA sequence analysis (Alipanahi et al., 2015; Umarov et al., 2019), protein structures prediction and reconstruction (Wang et al., 2015; Zeng et al., 2016), drug discovery and drug analysis (Wallach et al., 2015; Karimi et al., 2019), and biomedical image analysis (Ciresan et al., 2013; Havaei et al., 2017; Hu et al., 2020; Shin et al., 2020).

As mentioned before, DL has a complex structure compared to other ML methods, with multiple hidden layers architectures and thousands of parameters. For this reason, DL models require large datasets with thousands of instances to get good results (LeCun et al., 2015; Schmidhuber,

UNIVERSIDAD DE MÁLAGA

2015; Goodfellow et al., 2016). Currently this is not generally a problem; however, there are social impact fields for which the data acquisition process is difficult and costly in terms of time, legal and ethical components, and money. Although the DL performs better than other ML methods for some bioinformatics tasks described above, certain factors present in some bioinformatics tasks could limit the performance of DL models. The first factor is the scarcity of available data, since medical records and samples are sensitive data associated with legal problems. This problem becomes more pressing when rare diseases or specific geographical regions are analysed. The second factor is the complex information represented in biological datasets, which causes complex DL models to suffer from a high risk of overfitting and a lack of generalization. In this regard, there are some approaches that try to solve it, such as transfer learning (Cai et al., 2020) or collecting biological information from various sources (Mamoshina et al., 2016). A third factor is the imbalance present in the datasets, such that samples from one class are greatly outnumbered by samples from the rest. Lastly, DL has limited performance when applied to classification/prediction problems on datasets without 2D/3D patterns, sequential or structured information. It is clearly seen that DL models are the state-of-the-art in image processing and NLP, because CNN and RNN extract structured information.

Thereby, there are still bioinformatics tasks that present limitations. Note that performance limitations affect not only DL models, but also ML models. More specifically, gene expression microarrays, molecular expression or metabolomics datasets are significantly more difficult to acquire and have a greater dispersion compared to biomedical imaging datasets. There are works in the literature that apply DL with this type of datasets (Fakoor et al., 2013; Danaee et al., 2016; Urda et al., 2017; Montesinos-López et al., 2019; Zhu et al., 2020; Pomyen et al., 2020; Talukder et al., 2021), but they present less successful results compared to the efficiency of these algorithms in image recognition and NLP.

To solve the limitation of the reduced number of instances available and the class imbalance, the Data Augmentation (DA) strategy can be used. DA essentially consists of increasing the number of training instances, either by applying transforming instances already present in the dataset or by creating new synthetic instances. Most DA application successes are in image processing as well as DL (Simonyan y Zisserman, 2014; Szegedy et al., 2015; He et al., 2016). Some of the current DA methods are also DL frameworks, such as the Variational Autoencoders (VAE) (Kingma y Welling, 2013) and the Generative Adversarial Networks (GAN) (Goodfellow et al., 2014a). These models are included within the so-called deep generative models and their purpose is to learn the distribution of the dataset to subsequently generate synthetic instances. Several variants of both basic deep generative models have been proposed that show great performance, being able to generate

UNIVERSIDAD DE MÁLAGA

false images that experts may consider real (Radford et al., 2015; Pu et al., 2016; Karras et al., 2018; Zhu et al., 2017). The synthetic images created by the deep generative models have proven to be useful for DA process (Hsu et al., 2017b; Frid-Adar et al., 2018; Han et al., 2018; Waheed et al., 2020).

Figura B.1: Example of photorealistic generated faces using PG-GAN[1].

GAN models are also used in NLP (Xu et al., 2017; Hsu et al., 2017a; Pascual et al., 2017) and with other structured data such as time series or signals (Hsu et al., 2017b; Sheng et al., 2018; Shao et al., 2019). However, applying these DA models to datasets without spatial or temporal structure is a more challenging task. Domain experts can be asked to assess the quality of a generated image or text, even from a time series, and to distinguish a generated synthetic sample from a real one. However, this evaluation with human experts is not feasible when the DA is applied with data without structured information. Even more difficult with the complex bioinformatics data, being almost impossible to identify if a biological sample is real or synthetic, or to assess its quality.

Apart from some oversampling applications, such as SMOTE (Chawla et al., 2002), designed to deal with datasets that have imbalanced classes, the rest of the existing literature before 2015 on applying DA to non-structured data is mainly related to noise injection techniques as a way to avoid overfitting (Reed y Marks, 1998; Zur et al., 2009; Fernández-Navarro et al., 2011; Piotrowski y Napiorkowski, 2013). For this reason, one focus of the Ph.D. thesis is the adaptation of DL generative models to non-structured data problems, and especially to bioinformatics problems. It is interesting to note that in recent years, and in parallel with our research, GAN models are being applied to DA process in this type of problems (Douzas y Bacao, 2018; Liu et al., 2019; dos Santos Tanaka y Aranha, 2019; Marouf et al., 2020; Xiao et al., 2021). This current interest adds value to our work.

To conclude, we can summarize the objective of this Ph.D. thesis as the

[1]Extracted from Karras et al. (2018)

UNIVERSIDAD DE MÁLAGA

design, implementation and evaluation of robust DA methods and models in conjunction with deep learning, in order to improve its performance in the application to current and social impact bioinformatics problems. Specifically, in problems with genomic nature datasets that do not have spatial/temporal structure or information and present a small number of available samples.

B.1.2. Structure of the Summary

The rest of this summary is structured in a similar way to the Ph.D. thesis. First, a review of the state-of-the-art of DA methods and techniques is carried out. These are divided into transformation-based methods, commonly used to perform DA in image processing, and resampling and oversampling methods; and the DL's generative models.

The DA methods developed for the research and the results of their application are presented. Methods based on Gaussian noise addition, a forward noise adjustment scheme, and an adaptive DA model designed for deep generative models were developed. The conclusions drawn from the experimental results are also presented.

The application of DA models and methods developed on three bioinformatics problems is then presented: the events prediction in cancer through RNA-Seq samples analysis; the diagnosis of rare disease Niemann-Pick Type C from metabolomics information; and the molecular subtype breast cancer prediction using gene expression profiles of the immune system.

A meta classifier scheme is also described. This method was designed to take advantage of the sensitivity effects observed after the DA application in bioinformatics problems. Finally, the general conclusions obtained and possible lines of future work are presented.

B.2. State of the Art

Initially, data augmentation was used to reduce overfitting suffered during the training process. There are other techniques for this purpose in DL, such as the dropout method (Srivastava et al., 2014), pre-training (Erhan et al., 2010) and transfer learning (Pan y Yang, 2010). However, the DA aims to reduce overfitting from the base, the size and composition of the training dataset. This section includes a review of the state-of-the-art of DA methods, making a distinction between oversampling and transformation-based methods and the deep generative models.

B.2.1. Transformation-based Methods and SMOTE

From the application of transformation-based methods come the first demonstrations of the effectiveness of DA. These methods are usually applied

UNIVERSIDAD DE MÁLAGA

in image or natural language processing, so some of them are not applicable to problems dealing with unstructured datasets. These methods perform a direct modification of the original instances and usually have a simple implementation. Understanding these transformations provides a useful foundation for further investigation of more complex DA techniques. Of the transformation-based methods, the following should be highlighted: flipping a sample along its horizontal or vertical axis; rotation, clockwise or counterclockwise on an axis between 1 and 359 degrees; shifting data left, right, up, or down, useful for avoiding data positioning bias; cropping to select a section of the instance; color space transformations; addition of noise (random values) to the features present in the original data; kernel filters, a popular technique in image processing to sharpen and blur images; and random erasing of image sections (Zhong et al., 2020) inspired by the dropout regularization method.

Apart from some transformation-based methods, one of the first and most basic methods to augment the training dataset is resampling. Being the Jackknifing method (Quenouille, 1956; Tukey, 1958) the first application of resampling in statistical inference, the most common resampling method is the bootstrap (Efron, 1979, 1982). This technique is based on the generation of training datasets through the uniform random selection of instances and their duplication with replacement. The greatest exponent is the Bagging algorithm (Breiman, 1996), which joins the bootstrap and the application of individual classifiers on the resampled sets. The idea is that integrating different models, mitigating bias and variance, can obtain better results. Although this algorithm is impractical when the applied models are complex and their training is expensive in execution time and memory.

Some DA techniques deal with the specific problem of learning from unbalanced data. The oversampling methods partially eliminate this problem, trying to balance the classes by increasing the number of instances of the minority class. The predominant oversampling technique is the Synthetic Minority Oversampling Technique (SMOTE) (Chawla et al., 2002; Fernández et al., 2018). Instead of applying a simple replication of minority class instances (resampling), the key idea of SMOTE is the generation of new synthetic examples. This algorithm uses the interpolation of several neighboring instances belonging to the minority class for the generation. For this reason, SMOTE is considered to be a method focused on feature space rather than sample space. However, the SMOTE algorithm has certain disadvantages. There is a lack of control over the number of instances created, as it only generates enough instances to balance the dataset. Furthermore, due to interpolation, synthetic instances that violate the geometry present in the dataset for the minority class can be created.

Countless studies and numerous modifications have been made to the original SMOTE method. We highlight the Borderline-SMOTE (Han et al.,

UNIVERSIDAD DE MÁLAGA

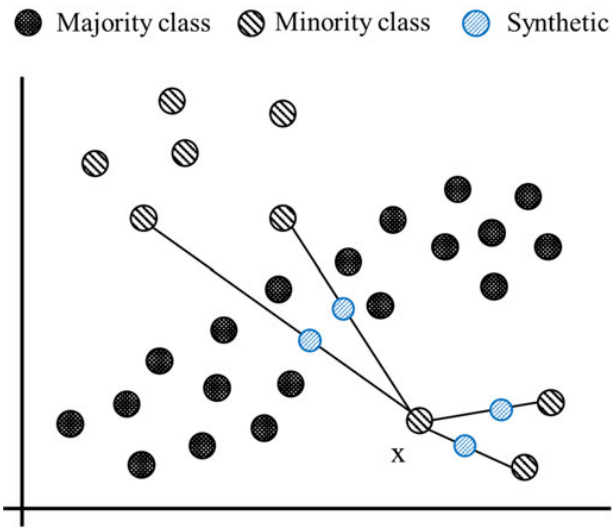

Figura B.2: Example of synthetic sample generation with SMOTE.

2005) method, based on the premise that examples far from the boundary between classes contribute less to learning. To avoid this, the method identifies the instances that are at the limit and generates new ones from them. Another similar method is ADASYN (He et al., 2008), based on the idea of adaptively generating instances of the minority dataset according to its distribution. Specifically, it generates more synthetic samples in regions where the density of minority instances is low, and fewer or none where the density of instances is high. However, these methods have a major disadvantage, they depend on the original configuration of the dataset and usually generate a very small number of instances.

B.2.2. Deep Generative Models

During the last decade, important advances in DL have led to the rejuvenation of neural networks. The new approach has enabled the creation of models capable of obtaining better performance in most of the research fields in which they have been applied, highlighting natural language processing and computer vision. Within the most current DA techniques there are also some models with DL frameworks.

Boltzmann machines (BM) (Ackley et al., 1985) were the first approach to deep generative models. These models can learn important features of an unknown probability distribution; however, it is a difficult and computationally time consuming process. Other deep generative models are the Deep Boltzmann Machine (Salakhutdinov y Hinton, 2009) and the Deep Belief Network (Hinton et al., 2006). Currently these models are rarely used, but they are deservedly recognized as they opened new debates that led to

UNIVERSIDAD DE MÁLAGA

advances and improvements in CNNs, RNNs or autoencoders.

The autoencoder (Bourlard y Kamp, 1988) is an unsupervised learning algorithm that establishes target values to be similar to inputs. The network is designed to learn an approximation to the identity function, not the identity function itself. In this way, autoencoders are designed including restrictions that only allow approximate copies of the inputs, e.g. limiting the number of hidden units in the network (sparse autoencoders (Ranzato et al., 2006)), or adding noise to the input (denoising autoencoder (Vincent et al., 2008)). Due to these restrictions, autoencoders are forced to prioritize which properties of the data should be copied, discovering useful information.

From the standard autoencoder network arise the generative model Variational Autoencoder (VAE) (Kingma y Welling, 2013). The encoder assigns the input data to a lower-dimensional latent variable space, and the decoder recreates the input from the latent space. The main difference between a VAE and an autoencoder is that a multivariate Gaussian distribution is generated in VAE. In this way, two sets of neurons are obtained as output from the encoder, one for the vector of means and other for the standard deviations. Both sets are combined to form the latent vector using an additional variable sampled from a Gaussian distribution. To generate a synthetic sample from VAE, it is necessary to pass noise of size equal to the latent space as input to the decoder. In this way, the decoder will try to recreate an input, thereby obtaining the synthetic sample. The most popular variants are the Denoising VAE (Im et al., 2017), which receives a noisy deformation of the original input; and the Conditional VAE (Kingma et al., 2014), which aggregates information regarding a condition in encoder and decoder. This makes the network aware of the type of sample that needs to be mapped into the latent space of the encoder.

The Generative Adversarial Networks (GAN) (Goodfellow et al., 2014a) are the DL models that lead the way in synthetic data generation. Their impressive performance in synthetic imaging has led to increased attention on their application in DA, multiplying exponentially in recent years. GAN models essentially consist of two connected neural networks, generator and discriminator, trained simultaneously, giving rise to a confrontation between both so that they learn from each other. In this context, the goal of the discriminator is to estimate the probability of the sample belonging to the real distribution or not. On the other hand, the generator takes as input a noisy random distribution and produces as output a distribution assigned to the space of real samples. The purpose of the generator is to create new synthetic samples with features that approximate those present in real samples, so that the discriminator does not distinguish these samples as samples not derived from the real distribution. Regarding the process of generating synthetic samples with GAN, noise is passed as input to the generator and this outputs the generated sample. The discriminator then predicts the pro-

UNIVERSIDAD DE MÁLAGA

bability that the new sample comes from the real distribution. If so, the sample has sufficient quality to fool the network, and can be used for a DA application. A more in-depth explanation of these models and VAE models can be found in Moreno-Barea et al. (2020).

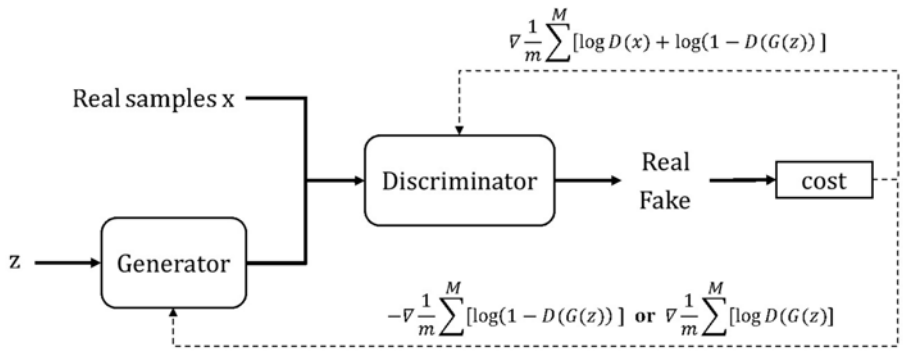

Figura B.3: Schematic diagram of the GAN model.

An essential modification applied to the Vanilla GAN to improve the image processing performance is the replacement of perceptrons by convolutional layers. Since the introduction of the Deep Convolutional GAN (Radford et al., 2015), all GANs were implemented using convolutional layers. However, the Vanilla GAN has certain disadvantages not solved by the use of convolutional layers: the vanishing gradient and the collapse mode. The first arises when the discriminator is too good and always detects which samples are created by the generator. The generator does not receive enough information to move forward, so the gradients gradually disappear. The collapse mode arises when the model is stuck in a local minimum. If a generator produces an output (or a small set of them) that is especially plausible as a real sample, the generator can be stuck producing only that output.

The Wasserstein GAN (Arjovsky et al., 2017) uses the Wasserstein distance (instead of the Jensen-Shannon Divergence) to evaluate the distance between the synthetic data and the real data. Intuitively, it can be seen as the minimum work required to transform one distribution into another. It also modifies the discriminator, renamed critic, which gets a score directly and performs several iterations for each iteration of the generator. Finally, it performs an operation called weight clipping, which is an adjustment of the critic weights so that the values stay within a controlled range. A variant that offers better performance is the WGAN model with gradient penalty (WGAN-GP) (Gulrajani et al., 2017). The weight clipping operation is replaced by the use of a gradient penalty term. The WGAN-GP model presents greater stability and greater independence between networks. Even so, WGAN-GP has higher computational complexity and increases the possibility of collapse mode.

UNIVERSIDAD DE MÁLAGA

Another variant of the Vanilla GAN is the Conditional GAN (CGAN) (Mirza y Osindero, 2014). As with the CVAE model with respect to VAE, the difference of the CGAN model with respect to GAN is the addition of the sample label. This additional information is assumed to help the CGAN model know what features to look for, introducing a bias at the nodes and leading to better results. A similar model is the *Auxiliary Classifier* GAN (ACGAN) (Odena et al., 2017). ACGAN provides the class label to the generator, but not to the discriminator. The discriminator must predict whether the given sample is real or false, but it must also predict the class label of the sample. The architecture of the discriminator and the auxiliary classifier use the same network with two sets of outputs. The first output is a probability that the sample is real, and the second output is the probability that the sample belongs to each class. However, ACGAN suffers from certain failures when generating samples and assigning labels due to sharing the same network. ACGAN can generate nearly identical samples for most classes by collapse mode (Miyato y Koyama, 2018). It also presents a performance problem when the dataset does not have a spatial or temporal structure, because the potential for duality of purposes lies in the use of convolutional layers in the discriminator, aided by the spatial information of the data.

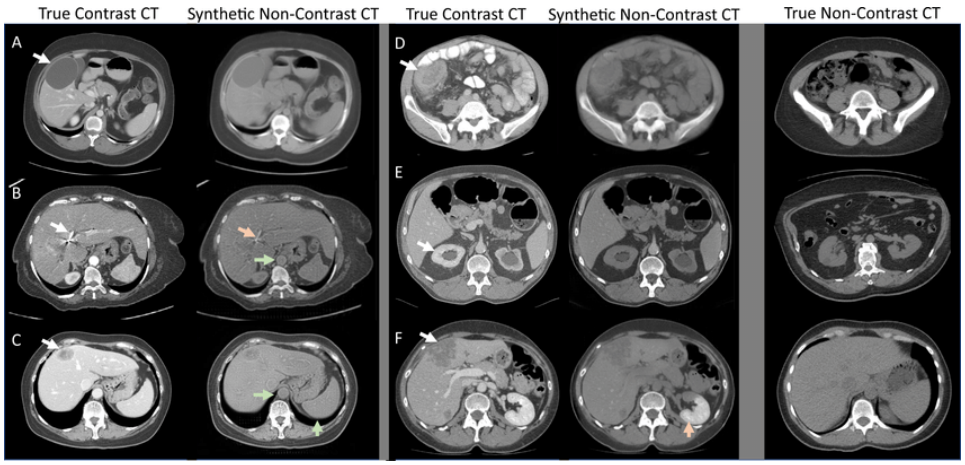

Figura B.4: Generation of non-contrast brain CT scans from contrast-enhanced images[2].

Regarding the DA application of deep generative models in bioinformatics, most studies focus on medical imaging (X-rays, CT scans, MRIs) (Wu et al., 2018; Sandfort et al., 2019; Frid-Adar et al., 2018; Han et al., 2018; Zhao et al., 2018; Waheed et al., 2020). However, its application in other domains has not been as successful and related work is more difficult to find. If we focus on datasets that are not images in biomedical problems, we can men-

[2]Extracted from Sandfort et al. (2019)

UNIVERSIDAD DE MÁLAGA

tion the works of Liu et al. (2019); Chen et al. (2020); Marouf et al. (2020); and the works of Chaudhari et al. (2020); Xiao et al. (2021) with genomic data in cancer. Considering these, there is a clear idea that the use of deep generative models for DA in domains where the samples are not images or time series (structured data), is a challenging task. The works on biomedical problems where genomic data are handled, which are significantly more dispersed, expensive and limited, are scarce and recent, but they show that these models can be beneficial to increase the prediction rate in classification. For this reason, we decided to design our own DA methods and apply these and state-of-the-art DA techniques to specific bioinformatics problems.

B.3. Designed Data Augmentation Methods

The first task performed in the Ph.D. thesis after reviewing the state-of-the-art, was the design and development of DA methods. A simple approach addressed in the first place is the development of methods based on addition of noise. Specifically, three noise addition methods are proposed, two based on the random injection of Gaussian noise and one method that creates new samples by adjusting the noise level for individual input variables previously classified by their level of relevance. Subsequently, the design of a DA model adaptable to the use of deep generative models was carried out, including additional functionalities to carry out a control over the balance of the generated set. This adaptive model was applied to variations of the VAE and GAN models with small datasets without spatial or temporal structure.

B.3.1. Noise Addition Methods

To perform DA there are some methods with approaches which may be considered simple, such as shifting, flipping, or clipping. A good initial approximation can be performed by using a simple method such as the addition of noise, based on a modification of original instances with an established degree of this factor. Noise injection methods have been used mainly to avoid overfitting in neural networks (Matsuoka, 1992; Reed y Marks, 1998; Zur et al., 2009; Piotrowski y Napiorkowski, 2013). However, it has not usually been considered as a proper DA model to create new synthetic data. Although the approach is simple, the design and application of a procedure based on the addition of noise can be extended, complicating its operation for adaptation towards different datasets, and with the ability to obtain effective results.

The first method developed to create new samples through a progressive and incremental procedure was called the Forward Noise Adjustment method (FNA) (Moreno-Barea et al., 2018). The motivation of this method lies in the difference in importance that the features of the datasets present when carrying out a classification process. Thus, there are features that in a

UNIVERSIDAD DE MÁLAGA

practical way do not affect the classification process, either because their values do not vary or because they are not relevant. Random Forest (Breiman, 2001) was used to measure the features importance.

The FNA method receives as input the dataset and the noise levels, whose values are between 0.1 and 1.0. Firstly, a stratified 10-fold cross-validation procedure is performed, 8 folds for training, 1 fold to choose the best noise level (validation) and the remaining fold to test performance. This last fold is not used during the adjustment so that the test results are totally independent. Having the features ordered descending according to their relevance, for each feature and noise level the algorithm goes through each of the folds, adding noise to the selected copies according to the corresponding parameters. The generated noisy data is concatenated with the original training set and a classifier model is trained. In this case, a deep neural network with 3 hidden layers with 32, 64 and 32 neurons respectively was used, with Leaky ReLU (Maas et al., 2013) as activation function, a combination of dropout (Srivastava et al., 2014) and L^2 norm as regularization technique, and the Adam algorithm (Kingma y Ba, 2014) as optimization algorithm.

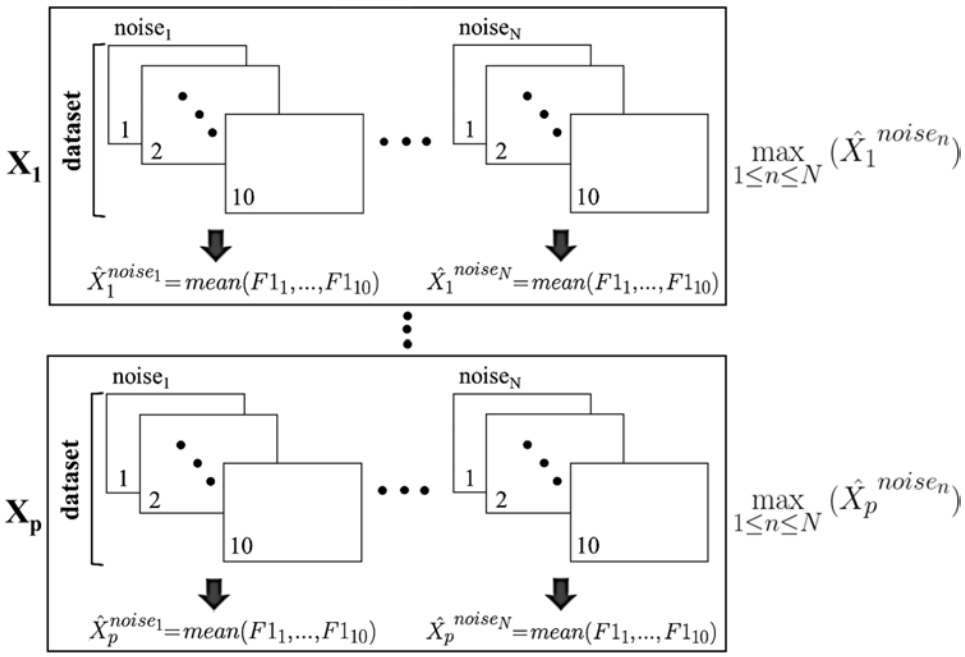

Figura B.5: Scheme of the FNA procedure proposed for estimating the optimal level of noise.

Once the model has been trained, the F_1 measure is obtained, being representative of the prediction performance with certain independence of the balance of the dataset. When all folds are tested for each noise level, the associated F_1 measure is the average. When all noise levels are tested

UNIVERSIDAD DE MÁLAGA

for the corresponding feature, the associated F_1 is the value associated with the noise level that maximizes validation. If the new value is greater than or equal to the best value obtained, then the algorithm creates new samples from the synthetic set by adding the noise level that has maximized the F_1 metric. When all features are tested, the algorithm returns the results obtained for each: the optimal noise level found using the validation folds and the F_1 measure averaged over their corresponding test folds.

An extensive number of experimental simulations were carried out testing the FNA method in a DA process, using nine well-known datasets, commonly used in machine learning from UCI (Dua y Graff, 2017) and PROBEN1 (Prechelt, 1994) databases. These datasets represent binary problems with a number of samples between 8 and 120. The effect was tested by increasing the training samples with a percentage of 100 %, 200 % and 500 %. This implies that the size of the training set is double, triple and sextuple respectively. The analyses indicate in the best case a 1.65 % average relative increase for the prediction accuracy (F_1 measure), corresponding to the situation in which the dataset was increased six times (500 % DA). The behavior of the proposed method was analysed as a function of the number of input variables, number of instances and degree of class unbalance of the problems considered and noticed no clear tendency with relationship to these important parameters. Although, it could be shown that the DA application not only leads to better prediction performance, but also accelerates the convergence of DL models.

Tabla B.1: F1 relative differences (RD) obtained measured for the nine benchmark datasets using different levels of Data Augmentation (100 %, 200 %, and 500 %).

Dataset	RD 100 % DA	RD 200 % DA	RD 500 % DA
heart cleveland	5.845	5.945	**6.606**
horse colic	0.801	1.789	**2.044**
wdbc	0.835	1.130	**2.024**
card	0.869	**0.978**	0.607
breast cancer	0.113	**0.201**	0.113
pima diabetes	1.090	**2.367**	1.275
gene	0.262	0.303	**0.499**
waveform	0.699	1.050	**1.180**
thyroid	0.355	0.464	**0.484**
Average	1.234	1.580	**1.648**

An advantage of FNA is that it can be applied with different feature selection methods and classification models. It is modular, so there is independence with the noise adjustment procedure. Any other performance evaluation metric could also be considered. However, this method has the great disadvantage of consuming a heavy computation time. Approximately 5 days are needed for each parameter set test (combination of noise levels) for

UNIVERSIDAD DE MÁLAGA

a 100 % level of increase. The experiments were run using a PC running Windows 10 64-bit OS, equipped with an Intel Core i7-6700K 4.00 GHz CPU, 32 GB RAM, and an NVIDIA® Titan Xp GPU with 12 GB RAM. The software used to perform the experiments was CUDA® Toolkit 9.0, cuDNN v7.0, Python v3.5.2, and TensorFlow v1.8 with GPU support (Abadi et al., 2016). In addition, although the results obtained with the method are satisfactory, the improvement in prediction achieved is not equivalent to the DA methods applied with images problems (Simonyan y Zisserman, 2014; Szegedy et al., 2015; He et al., 2016).

Due to the huge computation time required by the FNA procedure, it was necessary to implement other noise addition methods for comparison with deep generative models. In this way, two simpler methods were developed that require significantly less computation time. The noise addition method designed performs an increase in the number of available samples based on the random selection of "training" samples. For each sample, a copy is made, the features of which are then modified to a maximal value of 25 %. If the feature is chosen for modification, noise that arises from a random normal distribution with a standard deviation/variance of 1.0, is added to the original value. The value of standard deviation/variance is equal to 0.2 when the method is applied to datasets whose values are normalized between $[0, 1]$. The resulting "noisy" value must not exceed the limits established for its feature. In this manner, the new value must be greater than the minimum value present in the feature, a process that sets a lower bound. The value must also be less than the maximum value present in the feature setting an upper bound. A standard deviation of 1.0 or 0.2 (depending on the case) is sufficient to create a sample that does not stray too far from the sample.

A variation of this noise addition method has also been designed. This method, abbreviated in the different studies as "NOISE Bal", differs from the noise addition method described above, in that it performs the random selection only on the training samples of the minority class. Thus, only synthetic samples that belong to this class are created. This makes it an oversampling method, but unlike SMOTE-based methods, instead of creating a sufficient number of samples to perfectly balance the dataset, the designed method can create as many samples as desired. The remainder of the method follows the same process as the standard noise addition approach.

B.3.2. Adaptation of Deep Generative Models

A disadvantage of deep generative models is the labelling process of samples in generation for supervised learning. Except for the conditional versions, CVAE (Kingma et al., 2014; Sohn et al., 2015) and CGAN (Mirza y Osindero, 2014), the rest of the models are not designed with a supervised purpose. In this way, taking into account that these are essentially used in the ge-

UNIVERSIDAD DE MÁLAGA

neration of synthetic images from only a noisy input, a visual inspection of the generated image by an expert is necessary to assign it a label. However, experts cannot label a generated sample for unstructured datasets.

The proposed method uses a classifying model as an expert, in a scheme adaptable to any method or model (Moreno-Barea et al., 2020). In this regard, the Auxiliary Classifier GAN (ACGAN) (Odena et al., 2017) model should be noted, which gained popularity at the same time that we developed our method. The principle of this model is to use the discriminator network to predict whether it corresponds to a real or false sample, and at the same time make a prediction of the label that should be assigned. This is similar to the adaptive process developed, however, ACGAN suffers from certain failures when generating samples and assigning them labels because they share the same network for two purposes. Furthermore, it is not a model that can be adapted, for example, to a VAE or a WGAN. For these reasons, the development of the adaptive model is interesting and novel.

The adaptive model includes two blocks, a "Generator" and a classifier called "Generative Classifier". As already mentioned, in the investigation we deal with supervised problems involving labelled samples, so the DA process includes two steps in relation to this: first, the Generator is used to create new synthetic data and, second, a Generative Classifier is used to label the created samples. The whole process essentially consists of a loop between these two main threads, where an included control statement acts as a discriminator, checking if the Generative Classifier decides that the created sample is considered as noise or as a real sample. In the case of the sample is considered real, the Generative Classifier directly assigns a label that will allow its application in a supervised classification scheme. If noise is considered, the Generator must create a new sample, repeating the loop.

The training of the Generative Classifier model uses the original training set so that the model is able to assign the classes from real samples. However, the model also incorporates noisy samples into the training that are generated from two different sources: a) by using a uniform random distribution; b) by adding Gaussian noise. Samples created by both methods have a label that indicates that they are considered as noisy samples. These two different sources of noise are used because the first normal distribution applied is a way to teach the classifier that this is "purely" noise, while the second source creates samples related to the original distribution. The validation set is used in the model to perform an early stop on the model, in addition to evaluating its behavior. Due to the design of the general scheme of the process, any supervised learning algorithm can be used as a classifier model. The level of noise it receives can also be varied, or previous DA can be performed to improve its stability.

On the other hand, the Generator model uses only the training set, maintaining total independence with the validation and test sets. As mentioned,

UNIVERSIDAD DE MÁLAGA

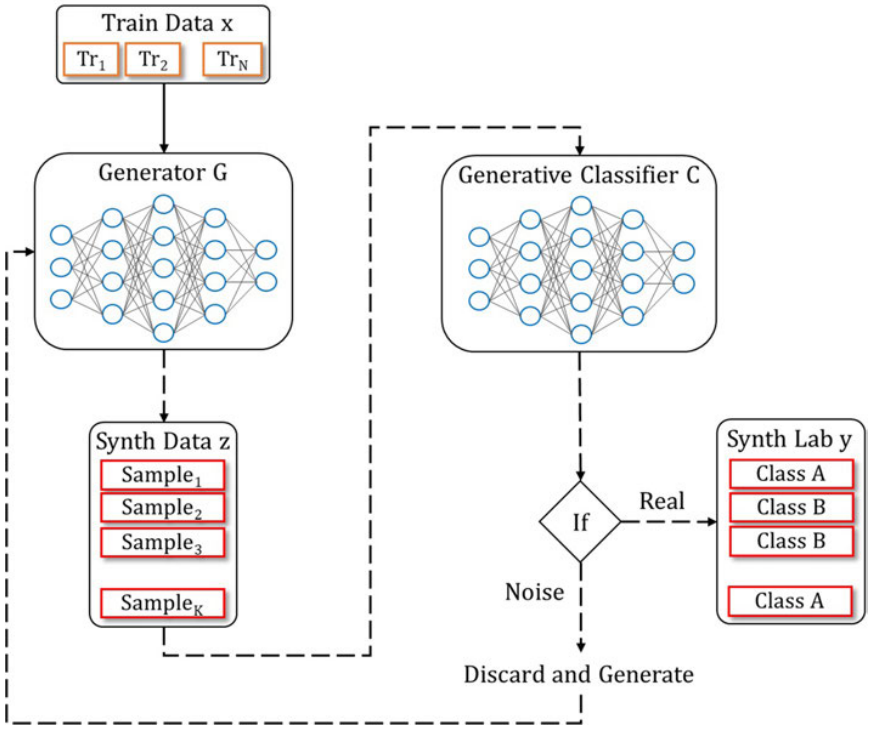

Figura B.6: The generative DA process for the creation of synthetic data. When the generative model is trained, synthetic samples are generated and the generative classifier predicts the appropriate labels. If the label predicted represents noise, sample is discarded and it generates another new sample.

this Generator model can be replaced by any deep generative model or other method that creates synthetic samples. Once the Generator has been trained, synthetic data generation can be applied followed by assignment of the corresponding labels using the Generative Classifier model. The Generative Classifier and Generator training processes can be performed in any order or in a parallel process. The only requirement is that the training of both processes has been completed before starting the assignment of labels.

The adaptive model is also applied to conditional models, although these have the label implicit in the generation. It is necessary to take into account the concept of security of a DA method, which refers to its probability of preserving the class label after performing a transformation or generation method. For example, flipping is safe if applied to a recognition task between cats and dogs. However, it is not safe for other tasks, such as recognizing the digits 6 *versus* 9. Deep conditional generative models often suffer from this problem. For example, in handwritten number generation, a CGAN from noise and label 6 can create a synthetic image that looks more like a 5 or even 8, but is assigned label 6. The only quality measure of these models is the

UNIVERSIDAD DE MÁLAGA

distinction by the discriminator of real appearance, which is true, but does not have a security measure like our method, where an expert classifier could assign a more appropriate label. Although, according to our experiments, in most situations the assigned label is the same. Furthermore, all DA models can carry some amount of distortion.

When different variations of the VAE and GAN models were directly applied to the generation of data, it was observed that many times they generate synthetic a larger number of synthetic samples for one of the category classes, usually for the larger one. This may cause an imbalance in the augmented set, leading to a possible collapse in the classifier model. In order to avoid this problem, and after experimenting with different options, two methods named 'Multiclass' and 'Balanced Multiclass' were introduced. Both modifications use as many generative models as classes are present in the original dataset, instead of a single generative model for the entire dataset. In this way, the first step consists in dividing the input data according to classes. For each class, one model will be trained taking all samples that belong to the category of the model together with a random selection of samples from the other classes, trying to reach a 20 % proportion of samples that do not belong to the model category.

The 'Multiclass' modification essentially works by keeping the proportion of samples of each class as in the original data. The 'Balanced Multiclass' aims to generate more samples for less represented classes, i.e., after the DA process is applied the augmented set will be more balanced. The number of samples generated by each of the generators depends on the implemented method used, on the number of samples desired and on the proportion of classes in the original set. These modifications have a double purpose, the first being to modify the bias of the generative models. A model that observes more samples of one class than others should focus better on generating higher quality samples for this class, while also learning the difference from the rest. Secondly, the introduced methods do not intend to force the model to generate only synthetic samples of a class, but to better adapt the distribution of samples of this class, trying to avoid the collapse that happens when the model is fed with all the data.

The architecture (number of layers and neurons layout) of the VAE and GAN models was selected after an extensive review of the literature and an optimization process. Validations prior to experimentation were carried out to refine the architecture, number of neurons and the rest of the parameters. To test the performance of the exposed models, experiments were carried out with 19 datasets taken from UCI (Dua y Graff, 2017) and PROBEN1 (Prechelt, 1994). The experimentation scheme designed includes two processes. First, the synthetic data generation process is carried out and they are added in the desired percentage to the previously separated validation and training sets. A 10-fold cross-validation procedure is implemented with

UNIVERSIDAD DE MÁLAGA

these sets. Once the training is finished, the prediction is carried out with the test set that was saved separately for an honest external test. The result is the average accuracy obtained for the 10 folds, repeating the process 10 times with different seeds to reduce random effects. Accuracy was used as a metric to prove if the DA through these models achieves an improvement in the total number of well-classified samples. Since the datasets are well balanced, accuracy is not a wrong metric. Data increase levels were tested with a percentage of 50 %, 100 % and 200 %.

Tabla B.2: Test accuracy using 50 %, 100 % and 200 % of augmented data, and maximum relative difference ($\widehat{RD}$) for all benchmark problems ordered according to the number of instances.

		Data Augmentation			
Dataset	Original	50 %	100 %	200 %	$\widehat{RD}$
iris	0.9126	0.9323	0.9327	**0.9427**	3.295
wine	0.9633	**1.0000**	**1.0000**	**1.0000**	3.810
sonar	0.6245	0.7519	**0.7781**	0.7676	24.595
seeds	0.8917	**0.8938**	0.8921	0.8933	0.237
glass	0.6351	0.6572	**0.6749**	0.6658	6.264
cleveland	0.8493	0.8636	0.8690	**0.8769**	3.248
ionosphere	0.8862	0.8945	0.9051	**0.9118**	2.892
horse colic	0.6574	0.6593	0.6660	**0.6697**	1.875
wdbc	0.9705	**0.9789**	0.9778	0.9769	0.870
non bal scale	0.9835	0.9869	0.9855	**0.9886**	0.521
bal scale	0.9258	0.9252	0.9250	**0.9441**	1.975
soybean	0.9241	**0.9283**	0.9281	0.9256	0.457
card	0.8536	0.8591	0.8541	**0.8618**	0.962
breast cancer	0.9736	**0.9754**	0.9754	0.9741	0.188
pima diabetes	0.7537	0.7566	**0.7578**	0.7555	0.543
geneN	0.8499	0.8833	**0.8839**	0.8768	3.996
gene	0.9008	**0.9051**	0.9028	0.8985	0.479
waveform	0.8525	**0.8634**	0.8630	0.8632	1.273
thyroid	0.9780	**0.9808**	0.9806	0.9767	0.281
Mean	0.8624	0.8787	0.8817	**0.8826**	3.040

The most relevant results shown that DA can lead to an increase in prediction accuracy of approximately 3 % (all tested method are evaluated, choosing the best one according to the validation error). If only one method should be applied, our results show that the best performing method among all datasets tested is the CVAE with the 'Balanced Multiclass' scheme and 50 % of DA, method that leads to a 1.03 % relative difference increase in accuracy (RD). An analysis of the number of instances and the accuracy gain obtained indicates a moderate negative correlation between both. This supports the hypothesis that the application of DA is more beneficial in those problems with a smaller number of available samples.

UNIVERSIDAD DE MÁLAGA

Tabla B.3: Best option model and their results for three cases: considering all datasets together or selecting the model for two groups of problems according to the number of instances.

Data	Method	$\overline{\#inst}$	Acc Val	RD	Acc Test	RD
Sets 1-19	CVAE_BM$_{50}$	1271	0.92	0.40	0.87	1.03
Sets 1-9	WGAN-GP_BM$_{100}$	283	0.90	0.73	0.85	4.12
Sets 10-19	VAE_M$_{100}$	2259	0.94	0.43	0.91	0.77

Thus, better results can be obtained if problems are grouped according to the number of instances available for training. In these case WGAN-GP with the 'Balanced Multiclass' scheme with 100 % DA was the best option for problems with lower number of samples (mean number of instances equal to 283) leading to an RD of 4.12 %. While standard VAE with the 'Multiclass' scheme and 100 % DA was the best alternative for datasets with a number of samples larger than 600 (mean number of instances equal to 2259). Nevertheless, it is worth noting that the results show that not all datasets benefit on the same degree, and results with lowest values close to 0.2 % to very high values up to 24.6 % increase in prediction accuracy were observed.

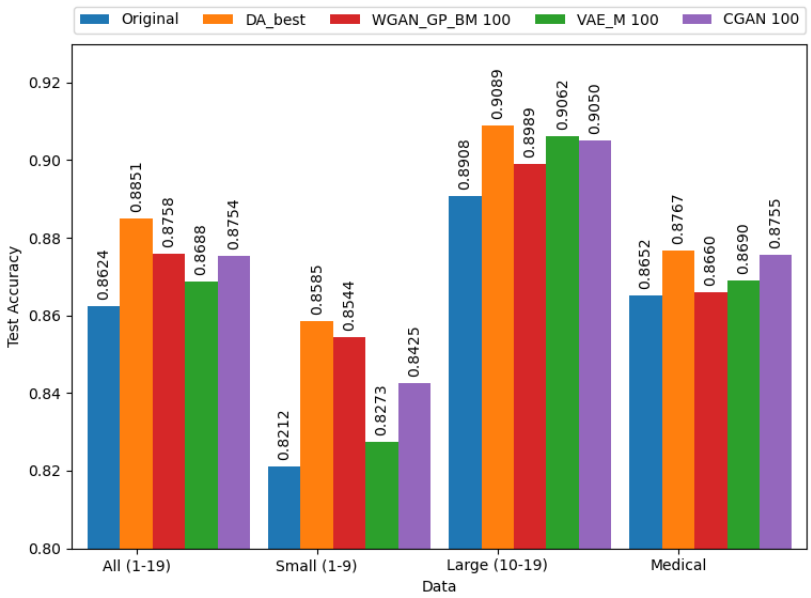

Figura B.7: Predictive ability using the original data, the best option for DA, WGAN-GP with 'Balanced Multiclass' 100 % DA, VAE with 'Multiclass' 100 % DA and CGAN with 100 % DA. Results are reported for four groups: all sets; grouped according to the number of instances; and the medical field.

UNIVERSIDAD DE MÁLAGA

The achieved performance using a logistic regression as a classifier model was also analysed. With this, an improvement of 4 % is obtained with the DA application. However, the maximum average accuracy is obtained by applying 50 % DA with a value of 0.8642, much lower than that obtained from the DL models when DA is used, and only slightly higher (0.8624) than the accuracy value obtained with the original datasets and DL.

The general conclusion that can be drawn from the results obtained is that DA techniques based on VAE and GAN are useful approaches to increase the prediction performance with small datasets, since they generate good quality data, which lead in average to a relative prediction increase of 3 %. Also that the conditional models (CVAE and CGAN), thanks to the addition of the class label in the generation process, seem to be able to generate samples of a higher quality, especially in their application with problems related to medicine, and are less influenced by data balancing changes. Although the profit with this type of problems is more difficult (barely 1 %). In relation to this, the experiments served to propose the application of the developed methods to genomic data, since until then no great advantages had been obtained from their application with DA and DL (Liu et al., 2019).

B.4. DA Application in Bioinformatics

The application area selected for the Ph.D. thesis is bioinformatics. As mentioned before, this research is integrated within the projects developed in the Computational Intelligence and Biomedicine (ICB) group, which has extensive experience in the application of artificial intelligence in the field of biomedicine. Currently, the ICB group is responsible for the application of AI algorithms to bioinformatics data within the cancer molecular biology group of the Intercenter Clinical Oncology Management Unit (UGCOI). The treatment of genomic data in bioinformatics presents a great problem. These are significantly difficult to obtain, have a greater dispersion in values and features, and suffer from the so-called curse of dimensionality, a large number of features compared to a very small number of samples. For these reasons, we wanted to test the effect of data augmentation in the application of deep learning models and other machine learning techniques in prediction/classification tasks in bioinformatics.

B.4.1. Events Prediction in Cancer

Cancer is one of the biggest health problems facing the society. According to the study conducted by GLOBOCAN (Sung et al., 2021), approximately 19 million new cases of cancer were detected and almost 10 million deaths from this disease occurred worldwide in 2020. In addition, significant delays in cancer diagnosis and treatment have occurred in 2020 and 2021 due to

UNIVERSIDAD DE MÁLAGA

the COVID-19 pandemic (Kutikov et al., 2020; Maringe et al., 2020). These delays can lead to increased cases of late-stage disease and increased mortality. Therefore, it is important to create new efficient diagnostic systems and improve current systems.

In this sense, the cancer events prediction such as the prognosis or survival prediction, including the time-to-event prediction, remain challenging problems in the clinical domain even after decades of effort by researchers (Nicholson et al., 2001; Kourou et al., 2015). The research carried out focuses on the vital status prediction of the patient in a fixed-time interval. In this way, the additional objective of predicting the cancer aggressiveness without taking into account the elapsed time is achieved. For this purpose, only gene expression profiles of mRNA sequenced from cancer patients are analysed. Technological progress in recent decades has led to the accessibility of high-throughput transcriptomic data from cancer patients. Researchers have found that transcriptomic data, especially gene expression profiles, can be very useful for cancer analysis (Valdes-Mora et al., 2018).

Different statistical models are used to perform this type of prediction task, including models in machine learning. A commonly used model is the neural network. The first neural network for this purpose was developed by Faraggi y Simon (1995), and now with DL, genomic data are used for events prediction. Although it remains a challenge (Ching et al., 2018; Kim et al., 2020), mainly because the amount of data available for deep learning models in bioinformatic problems is insufficient, leading to serious problems of overfitting in the models. For this reason, the impact generated by the increase in the amount of the genomic data available through DA methods was studied, trying to improve the performance in cancer vital status prediction in a fixed-time interval (Moreno-Barea et al., 2022c).

Specifically, a study was performed with mRNA-Seq data for 18 types of cancer freely available on the The Cancer Genome Atlas (TCGA) website (Tomczak et al., 2015), provided by the International Cancer Genome Consortium (ICGC). The samples represent cancer patients and contain 20531 variables corresponding to the expression level of each gene. A logarithmic transformation (log2) of the expression levels was performed to approximate them to a normal distribution for use the datasets with predictive models. In addition, a feature selection was carried out using LASSO (Tibshirani, 1996) and the Gini importance from Random Forest (Breiman, 2001). Feature selection reduces the number of genes, improving the datasets applicability with deep generative models.

A comparison was made between various DA methods, using the addition of Gaussian noise, SMOTE, the standard CGAN model and the CGAN model in the adaptive framework of the generative classifier model, including balance data modifications. To select the architecture, neurons and other parameters in the configuration of the models, including the deep neural

UNIVERSIDAD DE MÁLAGA

network used for the prediction, a previous experimentation and a review of models from the literature were carried out.

In the experimentation process, the dataset is firstly divided into training set and test set, in order to maintain total independence between data generation, classifier model training and prediction evaluation. The test set is kept separate to obtain an honest result of the performance. Next, the synthetic data generation process is carried out with the corresponding generative model on the preprocessed sets with feature selection. Once the process is finished, a stratified cross-validation procedure is implemented to obtain 10 folds, 9 for training and 1 for validation, and have a better estimation of the results. The training subsets are augmented by adding synthetic samples in the desired percentage. Then the classifier is trained and the prediction is made with the test data that was separated. The result is the average of the results obtained for the 10 folds. This process is repeated 10 times with different seeds to reduce random effects.

Tabla B.4: Test accuracy obtained (cols. 2-5) with the original dataset, when a feature selection method is applied, with DA methods applied, and the best RD obtained ($\widehat{RD}$). Also, test AUC obtained (cols. 6-8) with feature selection, with DA and the best RD obtained.

	Accuracy				AUC		
Dataset	Original	FS	DA	$\widehat{RD}$	FS	DA	$\widehat{RD}$
gbm	0.6353	0.7794	0.8185	5.02	0.6532	0.6703	2.62
coad	0.7015	0.7421	0.7529	1.45	0.7212	0.7666	6.30
ucec	0.7250	0.7955	0.8405	5.66	0.7432	0.7850	5.62
lgg	0.6802	0.8313	0.8413	1.20	0.9136	0.9312	1.93
ov	0.5192	0.7370	0.7921	7.47	0.7875	0.8134	3.29
luad	0.6071	0.6368	0.6694	5.13	0.6703	0.7124	6.28
lihc	0.6248	0.6690	0.6918	3.40	0.6716	0.7066	5.22
blca	0.5566	0.7412	0.7636	3.03	0.8209	0.8289	0.98
stad	0.6093	0.6623	0.6708	1.28	0.6718	0.6952	3.48
skcm	0.5213	0.6872	0.7377	7.34	0.7519	0.7677	2.09
prad	0.8918	0.9772	0.9810	0.39	0.5962	0.5999	0.62
lusc	0.5485	0.5624	0.5857	4.15	0.5669	0.5953	5.00
hnsc	0.5919	0.7000	0.7388	5.54	0.7710	0.7863	1.98
thca	0.9106	0.9640	0.9655	0.16	0.8007	0.8256	3.10
kirc	0.7239	0.7836	0.8041	2.62	0.8361	0.8481	1.44
stes	0.5747	0.5895	0.6033	2.35	0.5986	0.6080	1.57
kipan	0.7402	0.8265	0.8278	0.17	0.8515	0.8671	1.82
brca	0.8072	0.8531	0.8567	0.42	0.7615	0.7741	1.66
Mean	0.6667	0.7521	0.7745	3.40	0.7326	0.7545	3.06

Accuracy and area under the ROC curve (AUC) were used as metrics for vital status prediction performance. Accuracy was measured to analyse how the DA models directly affect the percentage of well-classified samples.

UNIVERSIDAD DE MÁLAGA

In addition, most of the datasets are well-balanced (11 of 18 sets have a value greater than 0.9) or does not present an excessive imbalance (14 of 18 have a value greater than 0.7), being 1 equivalent to a perfect balance. The particularity of this metric was taken into account during the analysis, being complemented with the AUC. Regarding DA levels, the effect was verified with sample creation percentages of 25 %, 50 %, 100 % and 200 %.

Results show that DA techniques application lead to an increase in accuracy of approximately 3 %, evaluating all the tested methods and choosing the best according to the validation error. This improvement has been achieved with respect to the datasets after carrying out the feature selection process, essential to increase the prediction potential in biomedical datasets. The improvement in accuracy with respect to the original 'raw' data is approximately 11 %. The results show that CGAN-based models improves the generalization results that can be obtained with other traditional DA techniques such as SMOTE and noise addition. With the CGAN-based methods, an increase in the classifier model accuracy is obtained (2.79 % RD), while with SMOTE and the addition of noise there is a decrease in the average performance. These conclusions have been corroborated by analysing AUC, obtaining an increase of approximately 3 % using the adaptive model.

Tabla B.5: Test accuracy RD and AUC RD obtained with each DA model selected according to the best validation result for each dataset.

	Accuracy RD			AUC RD		
Dataset	CGAN	SMOTE	NOISE	CGAN	SMOTE	NOISE
gbm	**5.02**	-0.45	-1.21	**2.62**	-6.03	1.39
coad	**1.45**	-5.82	-0.50	**6.30**	-3.87	-5.21
ucec	1.98	1.29	**5.66**	**5.62**	2.54	4.29
lgg	**1.20**	-1.03	0.70	**1.93**	-0.45	-0.91
ov	**7.47**	2.62	-6.56	**3.29**	-1.40	0.15
luad	**5.13**	0.35	-1.48	**6.28**	-2.85	0.58
lihc	1.19	1.64	**3.40**	**5.21**	1.67	0.70
blca	**3.03**	0.24	-0.76	**0.97**	0.02	0.72
stad	0.79	**1.28**	-0.91	1.62	-3.29	**3.48**
skcm	**7.34**	-0.37	-2.94	**2.10**	-0.17	1.40
prad	0.31	**0.39**	0.04	**0.62**	-0.27	-0.54
lusc	**4.15**	0.15	-4.98	**5.01**	-3.81	2.03
hnsc	**5.54**	1.50	-5.90	**1.98**	1.40	0.16
thca	0.05	-0.03	**0.16**	**3.11**	-1.09	2.19
kirc	**2.62**	0.01	2.04	1.40	-0.80	**1.44**
stes	**2.35**	-0.43	0.03	**1.57**	-1.35	0.17
kipan	**0.17**	-2.85	-1.31	**1.83**	-2.07	-1.89
brca	**0.42**	-0.90	0.09	**1.65**	-2.85	-2.15
Mean	**2.79**	-0.13	-0.80	**2.95**	-1.37	0.44

UNIVERSIDAD DE MÁLAGA

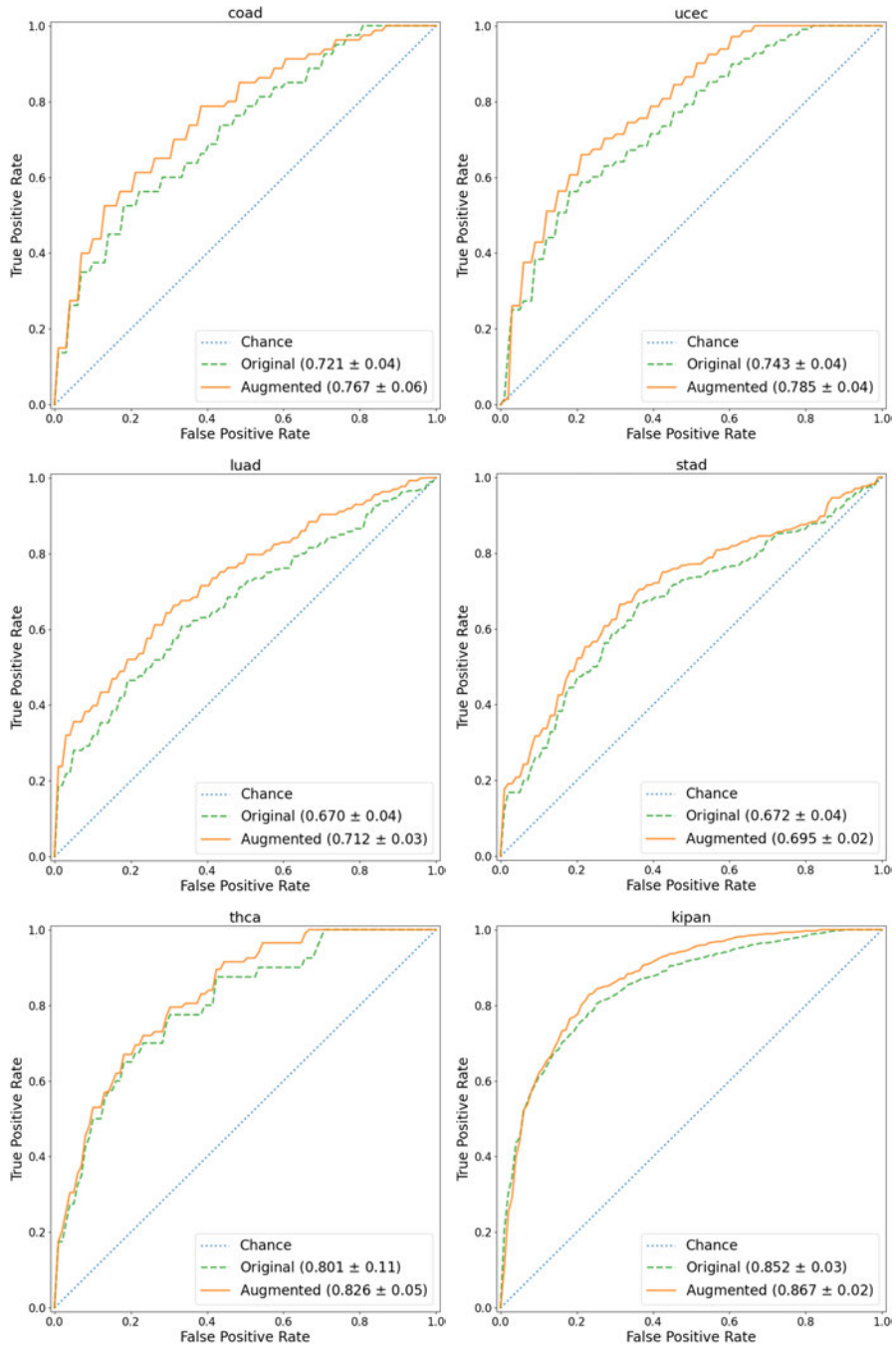

Figura B.8: ROC curves obtained using the original dataset and the data augmentation application. Dotted line represents chance; dashed line represents the approach with original data; and solid line represents data augmentation.

UNIVERSIDAD DE MÁLAGA

The analysis performed regarding the number of instances and balance of the datasets, shows a slight negative correlation between the increase in the number of instances and the accuracy gain for the GAN-based models, and a positive correlation between the increase in the balance of classes and accuracy gain. These results may be because the datasets with a higher number of instances or more unbalanced have a high accuracy base value, reducing the margin for improvement. These analyses also show a possible limitation in the application of DA in real applications, the decrease in performance if only one method is selected. This selection should be made depending on the balance or number of instances. Selection according to these characteristics allows an improvement in the performance achieved by selecting a single method, although it is not as high as would be desirable.

Graphical and statistical analyses were performed to examine the quality of the generated samples and reveal characteristics that explain the performance achieved. The results obtained through analysis of principal components (PCA) and statistical metrics such as the Fréchet Inception Distance (FID), the Standard Distance Deviation (SDD) and the shift of the mean centre of the distribution, corroborate that the samples generated by the DA methods have a good quality. The results show that the noise addition technique generates samples with a great fit and similarity with the original training set distributions, obtaining the lowest FID value (0.164). The results show that SMOTE has a potential to increase the variability of samples within of the minority class, but with a great restriction in the dispersion. The analysis indicates that the GAN-based methods add more variability to the datasets, generating samples that fit the real distribution but also some that are outside. The samples generated with the CGAN model have a lower similarity with the original sets, reaching a higher FID value (1.19). On the other hand, the samples generated with the adaptive model ModCGAN present lower values, indicating a greater similarity with the original samples. The FID values are almost half of those obtained with CGAN, indicating an improvement in the quality of the samples generated by our model from a statistical point of view.

Finally, the predictive models trained only with synthetic samples also indicate that the quality of the samples generated with the DA methods is high, obtaining a 0.7535 of accuracy, slightly higher than that obtained with the datasets without increasing (0.7521). These results indicate that the addition of noise generates samples with greater similarity and a more adjusted distribution to the original sets, reaching the best results in prediction with purely synthetic sets, although its impact with augmented sets is much lower. From these analyses it can be concluded that it is not only necessary that the synthetic samples have quality and present a good similarity with the real data, but also that they must provide variability to the augmented sets to increase the efficiency of the prediction models. This could allow better

UNIVERSIDAD DE MÁLAGA

Tabla B.6: Average Fréchet Inception Distance ($\widehat{FID}$) and Standard Distance Deviation (SDD) obtained with the samples generated by each DA model, for each dataset. Lower values of $\widehat{FID}$ represent greater similarity.

	SMOTE		NOISE		CGAN		ModCGAN	
Dataset	$\widehat{FID}$	SDD	$\widehat{FID}$	SDD	$\widehat{FID}$	SDD	$\widehat{FID}$	SDD
gbm	*0.150*	0.46	0.179	0.60	0.407	*0.44*	0.206	0.55
coad	0.076	0.34	*0.072*	0.30	0.451	0.36	0.147	*0.29*
ucec	*0.287*	*0.44*	0.299	0.53	1.305	0.57	0.741	0.54
lgg	1.015	0.56	*0.351*	*0.50*	2.082	0.52	1.359	0.58
ov	0.264	0.41	*0.103*	*0.40*	0.745	0.52	0.256	0.47
luad	0.040	*0.27*	*0.028*	0.33	0.283	0.35	0.046	0.35
lihc	0.030	*0.26*	*0.018*	0.33	0.089	0.35	0.021	0.27
blca	1.441	0.42	*0.541*	0.61	3.170	0.58	2.057	*0.40*
stad	*0.002*	0.21	0.003	0.26	0.112	*0.16*	0.042	0.21
skcm	0.175	*0.36*	*0.041*	0.40	0.277	0.42	0.107	0.38
prad	0.269	0.39	*0.047*	0.37	1.020	0.34	1.142	*0.24*
lusc	0.086	0.39	*0.029*	0.40	0.280	0.41	0.051	*0.37*
hnsc	0.028	*0.26*	*0.011*	0.32	0.111	0.28	0.015	0.28
thca	0.223	0.56	*0.222*	0.58	0.801	0.69	0.411	*0.52*
kirc	0.378	0.44	*0.341*	0.59	3.413	*0.41*	1.359	0.43
stes	0.171	*0.49*	*0.143*	0.57	1.440	0.65	0.576	0.60
kipan	*0.212*	0.47	0.319	0.49	3.448	*0.40*	1.818	0.45
brca	*0.128*	0.49	0.206	*0.49*	2.064	0.58	1.052	0.53
Mean	0.276	*0.40*	*0.164*	0.45	1.194	0.45	0.634	0.42

prediction of test samples that the model has never seen, and that may not be similar to the training samples. In conclusion, DA techniques constitute a suitable approach to increase the performance of the vital status prediction in a fixed-time interval from RNA-Seq gene expression profiles.

B.4.2. Niemann–Pick C1 Prediction

Niemann–Pick type C disease (NPC, OMIM 257220) is a very rare neurodegenerative lysosomal storage disease (LSD) caused by mutations in two genes NPC1 and NPC2 (Vanier, 2010). Usually, NPC disease presents in childhood with clumsiness, ataxia, learning difficulties, vertical gaze paralysis, and dysphagia, together with cataplexy, epilepsy, and hepatosplenomegaly (simultaneous enlargement of the liver and spleen). In severe cases, respiratory dysfunction may be another clinical feature.

Metabolomic strategies are valuable for the diagnosis and prognostic monitoring of such lysosomal diseases. The non-invasive nature of metabolomics and the close link of this type of data with the phenotype, make it an ideal tool for pharmaceutical and preventative health. In clinical studies, it allows the drug development for conditions investigated. Metabolomics is also ap-

UNIVERSIDAD DE MÁLAGA

plicable to the discovery of biomarkers as a support for decision-making. In metabolomics, biomarkers are small molecules, known as metabolites. As an example, selected metabolites and their concentrations can be used to determine the status of different groups of samples based on their detection in control group samples, or in those collected from patients with a specified disease. High-resolution proton (^{1}H) nuclear magnetic resonance (NMR) urine samples such as those analysed in this work contain informative metabolites that can be easily analysed for the purpose of discovering new biomarkers (Grootveld y Silwood, 2005).

The use of composites of bioanalytical techniques and computational intelligence techniques for metabolomics is evolving and becoming more popular (Marshall y Powers, 2017). However, the prediction performance achieved is usually reduced because the available metabolomics datasets are often limited in the number of samples. In addition, the datasets present are heavily unbalanced when carrying out studies on rare diseases, since there are numerous control samples available with respect to samples with the disease. In this regard, NPC diseases caused by both mutations affect approximately 1:100000 live births (Geberhiwot et al., 2018), being considered a rare disease, so its media impact is less than cancer. In such cases, the prior collection and the ethical consent required for such collections are often highly challenging hurdles to surmount. Additionally, obtaining a sufficient number of biofluid samples for NMR or other analyses adds to this complexity.

For this reason, the use of DA techniques can be beneficial. These techniques were used to improve the predictive performance on urine samples dataset donated by 13 untreated NPC1 disease and 47 heterozygous (parental) carrier control participants, a dataset collected by an UK-based clinical cohort. Experimentation is also proposed to detect unusual metabolic patterns in patients with NPC1 disease via an analysis of the ^{1}H NMR profiles of their urine samples, making a comparison with the biomarkers obtained using only the original dataset. The purpose is twofold: to measure the quality of the generated data, and to identify new biomarkers with these data, or due to the change in the balance (Moreno-Barea et al., 2022b,a).

The DA methods applied are the standard noise addition method and the oversampling noise addition method (Noise Bal), SMOTE, the standard CGAN model and two combinatorial strategies. These strategies combine samples generated from two different DA methods. Thus, the 'Comb 1' method performs a combination between the samples generated using the CGAN model and SMOTE; and 'Comb 2' performs a combination between the samples generated using the CGAN model and the Noise Bal method, when a DA percentage level of 500 % is applied with Noise Bal.

The experimentation process for the DA application in the problem is similar to the previous ones. First, a stratified division into training and test sets is performed to maintain complete independence between the data ge-

UNIVERSIDAD DE MÁLAGA

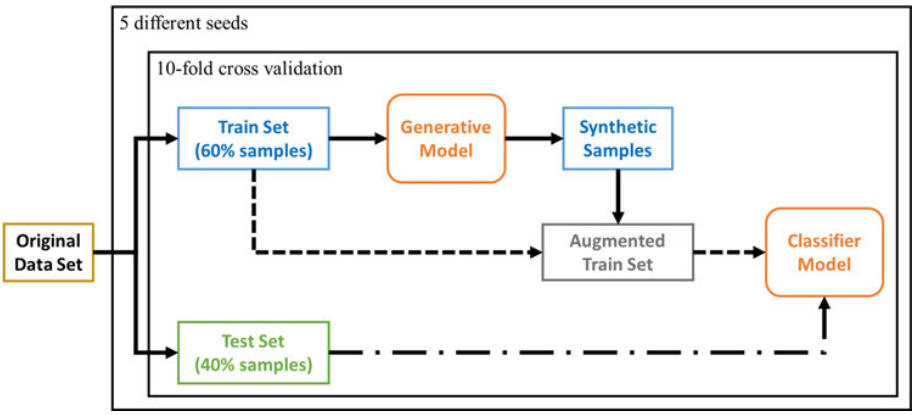

Figura B.9: Flow diagram of the whole experimentation process. The dataset is divided in a cross-validation process. A generative model is fed with the Training set, and synthetic samples are created and added. Data are transformed to principal components (PCs) by following a PCA. Once the classifier is trained, the prediction is made with the Test set. The process is repeated with 5 different seeds.

neration process, model training and evaluation. A division of 60 % of data for training and 40 % for testing was established in view of the small number of samples present in the benchmark dataset. In addition, the entire process of dividing the data and its application is performed by a cross-validation procedure to provide an improved estimate of the results acquired. The data generation process then employs the training set to create the desired number of synthetic samples. Once the synthetic samples have been created, a principal component analysis (PCA) is performed on the training dataset, obtaining the scores vectors and transforming all the data sub-sets (training, test and synthetic). This approach considers the percentages of the variance explained by each principal component (PC), in order to select a sufficient number of components. After some preliminary tests, 14 PCs were selected and the transformation of the sub-sets modifies the samples with the scores vectors. In this manner, the samples present values that are represented by PC vectors instead of the original variable values. This process is performed to remove any high levels of correlation (multicollinearity) between the variables present in the metabolomics dataset. With the transformed sub-sets, the augmented training set is used to train the classifier model. Once the training is completed, prediction of the test set is conducted, obtaining the metrics, which are then averaged for cross-validation. The entire process is repeated 5-fold with different seeds in order to alleviate random effects.

Generative models are tested with 7 different levels of DA percentages (25 %, 50 %, 100 %, 200 %, 500 %, 1000 %, 2000 %). In this case, experiments with high DA percentages were carried out because the number of available

UNIVERSIDAD DE MÁLAGA

samples is very small. The percentages also influences the selection of classifier and generative models. The standard version of CGAN is the only deep generative model selected. Modified versions that entail a higher computation time with such high percentages are not taken into account.

Logistic regression, a SVM system and a Random Forest were used as classification models. The results show that DA lead to an increase in prediction performance regardless of the approach used as classifier model. Specifically, the logistic regression approach obtained the highest accuracy value through a more conventional analysis performed on the original dataset (88.42), and it reaches approximately a 1 % improvement with the augmented set via standard noise addition. However, this improvement is small, although given the features of the problem, the percentage of well-classified samples is not representative. Since the dataset is very unbalanced, it is easy to obtain a high accuracy value simply by predicting all samples as control. It is more important that the largest number of patients with the disease be diagnosed as such and not as control patients. Therefore, the most representative prediction metrics for this problem are sensitivity and F_1 score. The results with the logistic regression shows that when performing data augmentation, the balanced noise addition method (Noise Bal) and 2000 % DA, an approximate 19 % improvement in sensitivity and a 6 % improvement in F_1 score were obtained.

Tabla B.7: Average test results acquired with the application of the different classifiers. Comb 1 is a combination of samples created with the CGAN and SMOTE strategies. Comb 2 is a combination of samples created with the CGAN and NOISE Bal approaches.

Method	Perc.	Accuracy	Specificity	Sensitivity	Precision	F_1
None	None	85.53	**97.40**	40.40	**82.24**	53.19
CGAN	117	87.30	93.51	63.73	72.07	67.62
NOISE	700	87.92	96.45	55.47	80.38	65.40
SMOTE	100	88.25	94.49	64.53	75.62	69.62
NOISE Bal	500	**88.44**	93.05	70.93	73.14	**71.96**
NOISE Bal	2000	84.92	86.64	**78.67**	61.95	68.99
Comb 1	83	86.58	90.21	72.80	66.96	69.55
Comb 2	67	87.00	90.95	72.00	68.07	69.84

The mean values obtained with the DA methods in the experiments with the different approaches can be considered the most relevant results, since they exhibit the most effective results obtained with a certain independence of the classifier model. Although they are slightly biased by the poor sensitivity performance obtained with Random Forest and the SVM system. Results demonstrate that the best performing method is Noise Bal and 500 % DA. This method leads to the highest mean values of accuracy (88.44) and F_1 (71.96) compared to employ only the original dataset (85.53 and 53.19 res-

UNIVERSIDAD DE MÁLAGA

pectively). The obtained PLS-DA cross-validation results also support this decision. The augmented set with involving data created using the Noise Bal model and 500 % DA provide approximately a 9 % improvement in accuracy, and a 0.3 (out of 1) improvement in predictive capacity (Q^2).

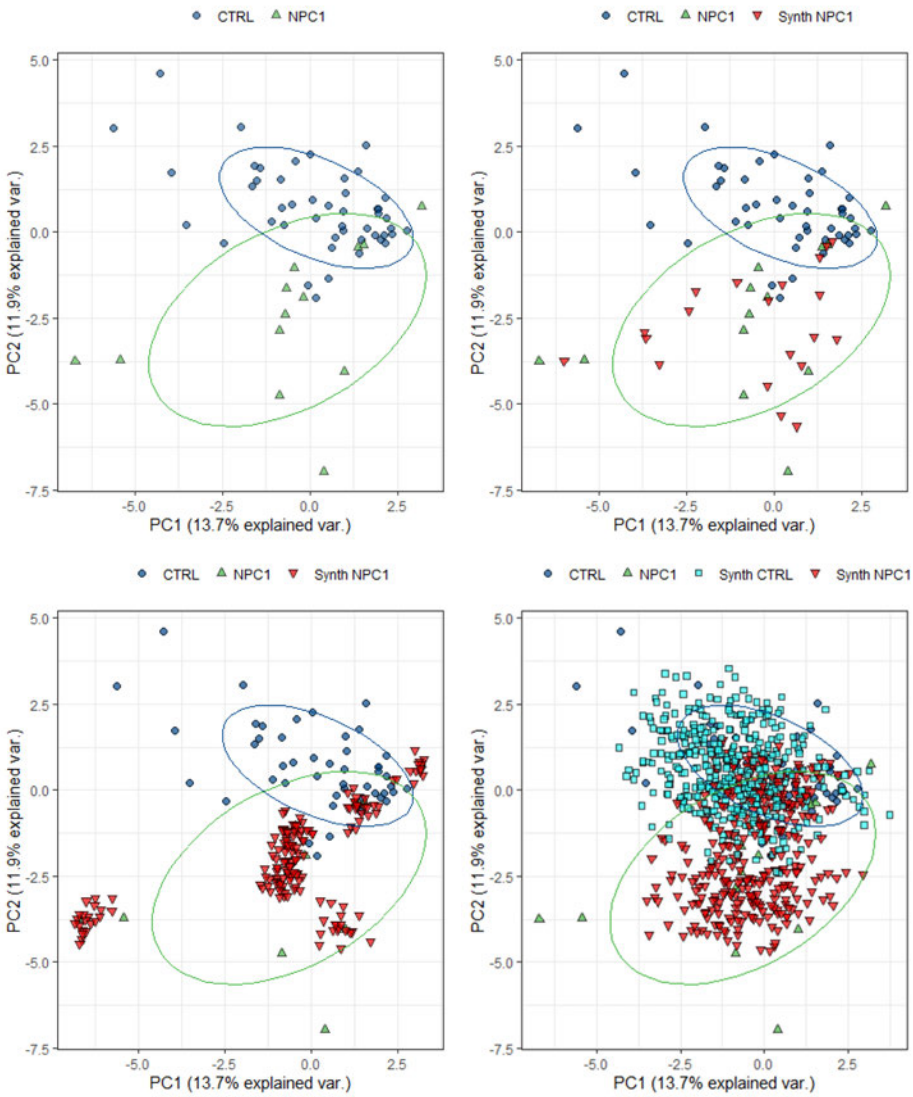

Figura B.10: PCA scores plot with: the original NPC1 disease dataset (top-left); the samples augmented with SMOTE (top-right); the samples augmented via the addition of noise with 2,000 % (bottom-left); and the samples augmented via CGAN with 2,000 % (bottom-right). Colour codes: green triangles, NPC1 disease urine; dark blue circles, heterozygous carrier control urine; red inverse triangles, generated NPC1 disease urine samples; light blue squares, generated heterozygous carrier control urine samples.

UNIVERSIDAD DE MÁLAGA

In order to determine the ability of the DA methods to replicate metabolic information, an analysis was performed using conventional forms of multivariate data analysis. The PCA and PLS-DA results for component 2 *versus* component 1 clearly show the differences in the generation of data by each method applied. The noise addition methods implemented generate synthetic samples that are grouped around the original samples that they modify. Hence, these methods have a clear capacity to replicate the information if the applied error is not excessive. Regarding SMOTE, two separate clusters with a small convergence zone were observed in the original dataset. This avoids the disadvantages that generation may present and it reflects a good replication capacity. The analysis of the CGAN model shows the ability of this model to replicate information that fits the distribution of the "real" samples. However, a disadvantage can also be concluded from these results. Given the internal function of the discriminator network, samples that are far from the core of the distribution (outliers) are considered as "false" samples and are discarded by the discriminator. The PLS-DA results are affected by the synthetic samples created by this model; this is clearly shown by the component 2 *versus* component 1 scores plot analysis.

Tabla B.8: PLS-DA VIP values obtained with the different DA methods and the relative rank position of each metabolite variable.

Metabolite	Original	SMOTE	Noise Bal	CGAN
3-Aminoisobutyric Acid	1.89 (1)	1.51 (5)	1.45 (4)	1.22 (8)
Quinolinic Acid	1.45 (2)	1.60 (3)	1.33 (6)	1.05 (11)
Trimethylamine	1.41 (3)	1.81 (1)	1.68 (2)	0.79 (16)
Trigonelline	1.21 (4)	1.12 (8)	1.23 (8)	1.79 (1)
Succinic Acid	1.20 (5)	1.13 (7)	1.07 (11)	0.70 (21)
3-Hydroxyisobutyric Acid	1.19 (6)	1.56 (4)	1.48 (3)	0.93 (13)
N-Acetylneuraminic Acid	1.13 (7)	0.93 (15)	1.26 (7)	0.93 (12)
Nicotinamide Riboside	1.12 (8)	0.68 (20)	1.44 (5)	1.51 (4)
L-Valine	1.11 (9)	1.38 (6)	1.23 (9)	0.66 (22)
Creatine	1.08 (10)	0.78 (18)	0.65 (22)	1.66 (3)
Glucuronic Acid	1.07 (11)	1.10 (9)	1.17 (10)	1.38 (7)
Propylene Glycol	1.05 (12)	0.43 (28)	0.40 (27)	1.14 (9)
L-Glutamine	1.03 (13)	0.96 (13)	0.77 (19)	0.70 (20)
Trimethylamine-N-oxide	1.03 (14)	1.05 (11)	0.93 (16)	1.49 (6)
2-H-3-Methylbutyric Acid	1.02 (15)	1.02 (12)	0.99 (13)	1.13 (10)
Acetic Acid	1.00 (16)	1.05 (10)	0.98 (14)	0.77 (17)
3-Hydroxyisovaleric Acid	0.99 (17)	1.69 (2)	1.86 (1)	0.58 (25)
N1-Methylnicotinamide	0.99 (18)	0.86 (17)	1.04 (12)	1.51 (5)
Pyroglutamic Acid	0.84 (19)	0.95 (14)	0.89 (17)	1.70 (2)
L-Alanine	0.43 (28)	0.16 (30)	0.24 (31)	0.87 (14)

Finally, variable importance in projection (VIP) scores with respect to component 1 were obtained using the PLS-DA technique (Farrés et al., 2015). The VIP results conclude that DA methods are able to replicate impor-

UNIVERSIDAD DE MÁLAGA

tant metabolomic information to find biomarkers, and support other results obtained in this field (Ruiz-Rodado et al., 2014). It appears that urinary BCAAs such as valine, the BCAA and thymine degradation product 3-aminoisobutyrate, 3-hydroxyisovalerate and 3-hydroxyisobutyrate, and perhaps also quinolinate and trimethylamine, may be employable as valuable biomarkers for the diagnosis of NPC1 disease.

B.4.3. Cancer Molecular Subtype Prediction

Breast cancer represents a global health issue with a high incidence each year. Breast cancer is the most common tumor diagnosed in women, with an estimated 2.3 million cases worldwide, 11.7 % of all diagnosed cancers in 2020 (Sung et al., 2021). Fortunately, the breast cancer mortality is lower than other types of cancer, with a mortality rate close to 6.9 %, thanks to improvements in early detection techniques and less aggressive therapies. Regarding the difference according to geographic region, the incidence rates are 88 % higher in developed countries than in developing countries. However, developing countries have 17 % higher mortality rates in comparison.

One of the most interesting topics to combat this disease is the ability of the immune system to act against cancer cells. The immune system, affected by the local microenvironment, determines the generation of an efficient response against foreign antigens An important function of the immune system is to confer protection against external pathogens such as viruses, bacteria and parasites, however, a debated topic was cancer prevention as a primary function of the immune system (Coley, 1893; Ehrlich, 1908). Based on a greater knowledge of tumor immunology, interest in tumor immunosurveillance resurfaced at the end of the 20th century, and several studies revealed that the molecular components of the immune system are involved in the development of primary tumors (Burnet, 1971; Thomas, 1982; Kaplan et al., 1998; van den Broek et al., 1996). In the last decade, immunotherapy has become a powerful clinical strategy for cancer treatment. The number of immunotherapy drugs and treatments has been increasing, and countless research have been conducted to understand the effects of the immune system on the development and establishment of tumors (Pardoll, 2012; Sharma et al., 2017; Martin et al., 2020).

Conventionally, the classification of breast cancer was based on the histological characteristics found under light microscopy. With immunohistochemical and genetic analyses of tumor biopsies, a new classification based on molecular subtype emerged. According to studies of gene expression patterns related to prognosis or risk of metastasis (Perou et al., 2000; Sørlie et al., 2001), breast cancer can be divided into two large groups based on estrogen receptor positivity. One group is composed of low-grade neoplasms, or estrogen receptor positive tumors, which are further divided into diffe-

UNIVERSIDAD DE MÁLAGA

rent subtypes called Luminal. A second group is composed of high-grade neoplasms, or estrogen receptor negative tumors, which are mainly divided into basal and HER2 gene overexpression. This classification is important in clinical practice because it allows avoiding unnecessary treatments and distinguishing the tumor aggressiveness. Its implementation is mandatory for each patient and it is a very useful tool.

For this reason, the application of different DA techniques was proposed with the aim of improving the prediction performance of the molecular subtype of breast cancer when analysing a dataset with gene expression profiles of the immune system of the patients. The dataset present three low-grade subtypes, luminal A (LA), luminal B HER2 negative (LBH-) and luminal B HER2 positive (LBH+)); and two high-grade subtypes (HER2-positive pure (Her2) and triple-negative (TN)). Two molecular subtype prediction problems were then addressed, a binary problem with a classification according to whether the subtype is low-grade (Luminal) or high-grade (Non-Luminal); and a multiclass problem with a classification in the 5 subtypes. The dataset belongs to a clinical study of patients with breast cancer from the province of Malaga, diagnosed and treated between December 2008 and February 2013 at the Intercenter Clinical Oncology Management Unit (UGCOI).

The DA methods applied are the standard noise addition method and the oversampling noise addition method (Noise Bal), SMOTE, the standard CGAN model, the standard ACGAN model, and the CGAN model in the adaptive framework of the generative classifier model (ModCGAN), including the Balanced Multiclass modification (Moreno-Barea et al., 2020). To deal with the oversampling methods in the multiclass problem, it was necessary to design a specific procedure. In the multiclass prediction of the molecular subtype, there are 3 minority classes distributed with 12.1 % (LBH+), 8.5 % (Her2) and 9.7 % (TN), compared to two majority classes distributed with 36.9 % (LBH-) and 32.7 % (LA). For this reason, the Noise Bal method randomly selects samples from only the 3 minority classes for noise addition and synthetic sample creation. Regarding SMOTE, this method is applied with the differentiation 'LBH+ *vs.* LBH-', 'Her2 *vs.* LBH-' and 'TN *vs.* LBH-'. In this way, synthetic samples are created only for the three minority classes.

Before the experimentation, a feature selection was performed using Gini importance of the Random Forest method to reduce the number of variables. This process was performed independently for each problem treated (binary/multiclass). The resulting dataset is considered as the original dataset. In the experimentation, a principal component analysis is also performed, obtaining the component score vectors and transforming the dataset. This PCA strategy is applied because the gene expression variables only indicate the presence or absence of the molecular variables encoded in the major histocompatibility complex. The original variables are all binary, implying

UNIVERSIDAD DE MÁLAGA

greater difficulty in DA application and prediction of the molecular subtype.

Once the PCs transformation is complete, a stratified cross-validation procedure divides de dataset into training, validation and test sets. A division of 50 % for training, 10 % for validation and 40 % for testing is established due to the reduced number of samples present for some subtypes in the benchmark dataset. The generation process uses only the training set to create the desired number of synthetic samples. After the synthetic samples are created, the augmented training set is used to train the classifier model. The prediction of the test set is then performed, obtaining the performance metrics that are then averaged for the cross-validation. The whole process is repeated 10 times with different seeds to alleviate random effects. Generative models are tested with 6 different levels of DA percentages (50 %, 100 %, 200 %, 500 %, 750 %, 1000 %). In this case, the original dataset has 165 samples and these percentages are enough to verify the effect of the DA in the classification.

Tabla B.9: Test results obtained for the binary subtype prediction using a deep neural network as a classifier with the different DA methods.

Method	Perc.	Acc	Sens	Spec	F_1	MCC	AUC
Original	None	0.8141	0.3058	0.9270	0.3478	0.2701	0.6042
CGAN	500	0.8288	0.3567	0.9337	0.4133	0.3436	0.5899
ACGAN	1000	0.8303	0.3667	0.9333	0.4202	0.3511	0.6145
ModCGAN	1000	0.8262	0.3342	**0.9356**	0.3241	0.3172	0.6030
Mod_BM	1000	0.8338	0.4008	0.9300	0.4229	0.3800	0.6613
NOISE	100	0.8214	0.3175	0.9333	0.3747	0.2993	0.5796
NOISE Bal	1000	**0.8424**	**0.5242**	0.9131	**0.5331**	**0.4535**	**0.7302**
SMOTE	None	0.8094	0.2717	0.9289	0.3190	0.2444	0.5155

Tabla B.10: Test results obtained for the multiclass subtype prediction using a deep neural network as a classifier with the different DA methods.

Method	Perc.	Acc	LA	LBH-	LBH+	Her2	TN
Original	None	0.3224	7.83	11.71	0.44	0.86	0.44
CGAN	200	0.3359	8.24	12.1	0.5	0.83	0.5
ACGAN	750	0.3352	**8.53**	11.96	0.31	0.67	0.65
ModCGAN	500	0.3420	8.44	**12.82**	0.55	0.42	0.34
Mod_BM	500	0.3359	7.96	12.2	0.44	0.85	**0.72**
NOISE	750	**0.3423**	8.3	12.74	0.31	0.86	0.38
NOISE Bal	500	0.3239	7.64	11.55	**0.61**	**1.14**	0.44
SMOTE	100	0.3089	7.6	10.83	0.39	1.09	0.48

A deep neural network, a SVM system and a random forest were used as classification models. Results reflect a greater difficulty in multiclass prediction of the molecular subtype compared to the binary approach as expected (Ha et al., 2019; Ma et al., 2019). The differences are wide regardless of the

UNIVERSIDAD DE MÁLAGA

classifier model or the DA technique applied. With a deep neural network as classifier model, a maximum accuracy of 84.2 % is obtained for the binary problem, against just 34.2 % accuracy for the multiclass problem. The analyses show that the noise addition method with the oversampling purpose (Noise Bal) leads to an increase of approximately 3.4 % in accuracy for the binary approach and 6.2 % for the multiclass problem. This improvement in accuracy is sufficient for both approaches. However, since the prediction of the minority class in the binary approach implicitly has a higher value, since it is identified with the more aggressive high-grade subtypes (Non-Luminal), the most representative prediction metrics are the F_1 score and the Mathews Correlation Coefficient (MCC). MCC is a metric that indicates the overall performance of a classifier model in correct predictions (Matthews, 1975; Baldi et al., 2000). In this respect, an improvement of 18.5 % in F_1 and 18.3 % in MCC with respect to the original dataset results is obtained by applying the Noise Bal method in the binary problem.

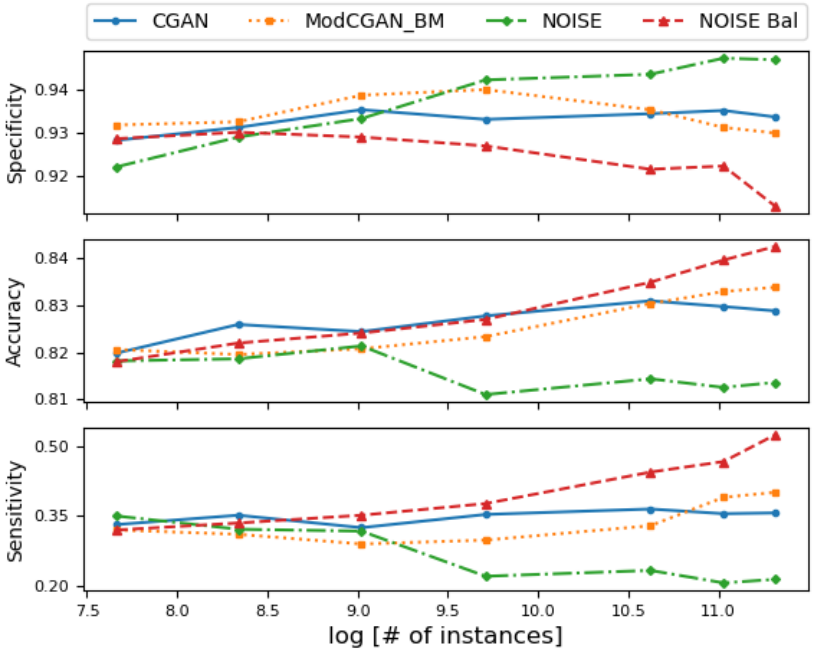

Figura B.11: Accuracy, specificity and sensitivity obtained using a deep neural network as a classifier with different DA models with respect to the log of the number of instances created. Code lines: solid blue, CGAN model; orange dots, ModCGAN_BM; green dashes and dots, Noise; red dashes, Noise Bal.

These results were replicated by using a Random Forest and a SVM system. With these methods, higher values are obtained with the DA application for all the performance metrics, especially sensitivity. With a SVM system and the application of the Noise Bal method in the binary problem,

UNIVERSIDAD DE MÁLAGA

an accuracy of 84.9 %, a value of 0.64 for F_1 and 0.55 for MCC are obtained. These represent an improvement of 3.8 %, 36.4 % and 33.5 % respectively with respect to the results without augmentation. In the multiclass problem, a maximum accuracy value of 37.5 % is reached, which is a poor result. The analyses show that the DA application improves the classification of minority subtypes, although to a much lesser degree than is desirable. In the experiments carried out, most of the samples are predicted as belonging to the majority classes LBH- and LA.

In addition to predictive performance analysis, principal component analysis (PCA) and different statistical metrics (FID, SDD and Shift) are used to analyse the characteristics and quality of the generated samples. The results acquired from the analyses support the conclusions obtained with other bioinformatics problems. The analyses show that by the CGAN-based models application, the generated samples fit the distribution of the real data. ModCGAN with balance modification (ModCGAN_BM) performs a better fit than the other deep generative models. However, samples that could be considered outliers are observed in the original set. With the deep generative models, samples are not generated near the outliers, since the discriminator considers these as false. Meanwhile, the Noise Bal strategy generates samples close to the "outlier" samples if they belong to the training set, which explains its better prediction performance. These conclusions are supported by the similarity and dispersion metrics, obtaining the noise addition and SMOTE methods the best FID values and dispersion values close to the original dataset. In this case, the dispersion metric value is highly influenced by the outliers, so the deep generative models obtain low SDD values and a larger shift from the original distribution. The only CGAN-based model that obtains good values for these metrics is ModCGAN_BM. This obtains good similarity, acceptable dispersion and a low shift.

Tabla B.11: Fréchet Inception Distance ($\widehat{FID}$), Standard Distance Deviation (SDD) and shift of the mean center of the distribution (*Shift*) obtained with the samples generated by each DA model with respect to the real distributions of the binary classes.

	Class 0			Class 1		
Method	FID	SDD	Shift	FID	SDD	Shift
None	-	0.88	-	-	2.85	-
CGAN	25.66	1.02	1.50	85.24	1.28	2.81
ACGAN	57.87	1.65	1.56	54.34	1.30	0.88
ModCGAN	71.00	2.74	1.34	86.20	2.27	0.98
ModCGAN_BM	38.20	1.23	0.43	46.57	2.04	0.50
Noise	13.88	1.03	0.03	39.08	2.64	0.50
Noise Bal	0.00	0.00	0.00	13.78	3.47	0.58
SMOTE	0.00	0.00	0.00	23.31	2.18	0.68

UNIVERSIDAD DE MÁLAGA

In conclusion, prediction of the molecular subtype of breast cancer using gene expression profiles is an extremely complex problem. If it is simplified to the binary prediction of the low- grade or high-grade subtype, the results obtained are better. Likewise, the application of DA techniques implies an increase in prediction performance for both approaches. For the binary problem, the Noise Bal method has shown the best performance, leading to a 3.6 % increase in accuracy, with particular attention to increased sensitivity, resulting in an 18-36 % improvement in F_1, 18-33 % in MCC and 6-20 % in AUC. Due to the increase in sensitivity observed with the DA in this and the previous problems, the development of a method that takes benefit of this advantage together with the higher specificity achieved by the non-augmented dataset is interesting.

B.5. Meta-Classifier Method

Class imbalance in datasets is a common problem in the application of machine learning algorithms to many real-world problems, as in most cases data categories are not equally represented, with some classes having more instances than others, and in particular with relevant categories being underrepresented. When machine learning algorithms in a supervised task are applied to these datasets, the classification algorithms are affected by class imbalance (Japkowicz y Stephen, 2002; Kubat et al., 1997). Most of predictive algorithms focus on improving classification accuracy, however, reaching a high predictive accuracy in a problem with a high percentage of majority samples is trivial; e.g. in a binary problem with unbalanced classes 2:98 (2 minority class samples, 98 majority class), the classifier should only predict all samples as belonging to the majority class to obtain an accuracy of 98 %. This implies a poor classifier performance and a potential risk depending on the application domain.

Techniques proposed by researchers to deal with imbalanced datasets can be divided into algorithm level, cost-sensitive, and data level methods. These last techniques are the most popular and can be categorised into undersampling techniques, which eliminate samples of the majority class (Drummond et al., 2003; Tomek, 1976), and oversampling techniques, which generate samples for the minority class (Chawla et al., 2002; Douzas et al., 2018). As has been analysed throughout this Ph.D. thesis, with these techniques, as with data augmentation in general, classifier models can greatly improve the ability to correctly predict minority class samples (sensitivity). However, this usually also leads to a large loss of majority class prediction (specificity) and a reduction in overall performance.

A meta-classifier scheme was designed with the purpose of improving the prediction capacity for minority samples, but reducing the loss for majority samples (Moreno-Barea et al., 2022d). The proposed scheme makes use of

UNIVERSIDAD DE MÁLAGA

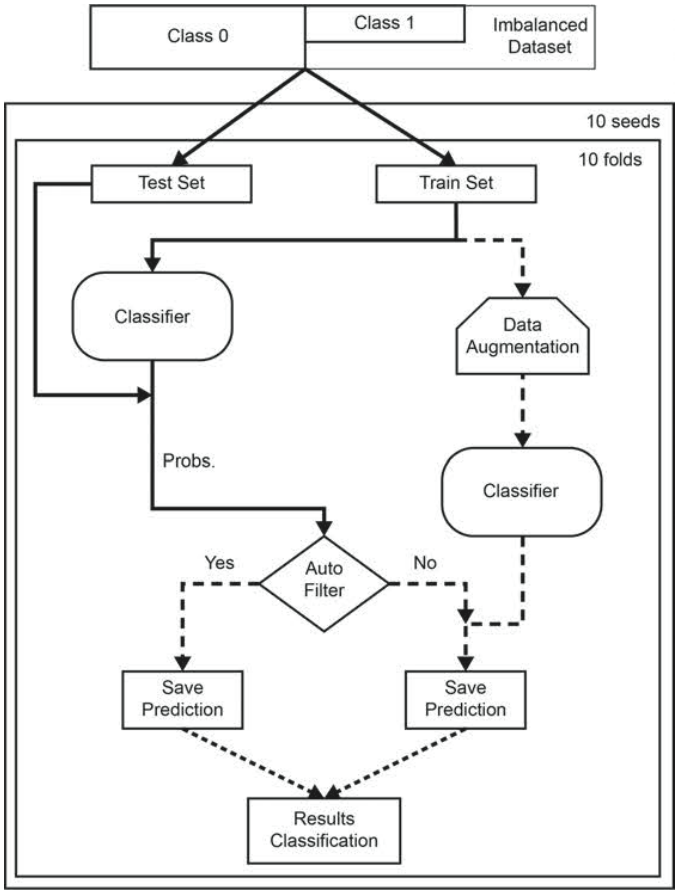

Figura B.12: Flow diagram of the whole experimentation process with the meta-classifier scheme for data augmentation.

two classifiers, one trained with the original dataset and the other trained employing DA, with a filtering process to select the test samples. From the first classifier model, the output probabilities for the prediction with the test set are obtained. In this way, we expect the model to suffer from the imbalance of the data, making a prediction that comes close to predicting the majority of the test samples as belonging to the majority class. To minimise this effect and to be able to take advantage of the use of DA, a filtering of the classifier's output probabilities is carried out before class assignment.

The developed automatic filtering procedure separates the test samples that the classifier clearly indicates to which class their belongs. For this purpose, the procedure uses threshold values for the majority and minority classes, which are compared with the output probabilities. However, the process only advances if a minimum number of samples remain outside of the threshold. Therefore, the threshold is increased in each iteration, more slowly as its value is higher. In this way, a more precise distinction can be made

UNIVERSIDAD DE MÁLAGA

Algorithm 6 Automatic Filtering Procedure

```
procedure INPUT(probs: classifier output probabilities of test samples to belong to each class.)
    Initialize the threshold values T_maj = 0.55 for the majority class (0) and T_min = 0.7 for the minority class (1).

    while samplesM.length< 2 do
        for j = 1 → N do                                   ▷ loop over test outputs
            if probs_j[0] ⩾ T_maj or probs_j[1] ⩾ T_min then
                Index j is added to samplesK
            else
                Index j is added to samplesM
            end if
        end do
        Update T_maj increasing the value for the next iteration and depending on the current value
    end do
    return samplesK: set of samples keeping their classifier output; samplesM: set of samples to be tested with the augmented classifier.
```

for prediction outputs, allowing the threshold to reach a value that allows a sufficient number of samples to continue in the meta-classifier process.

Once the filtering procedure has been carried out successfully, the DA method is applied to the original training set and a second classifier is trained with the augmented set. The application of DA should help this second model to better predict the samples belonging to the minority class, according to the studies and research previously carried out. Thus, a prediction is obtained for the test samples that continued in the process. Taking into account all the predictions, we obtain the test metrics to measure the classification quality. It is necessary to understand that this model does not include an output vote, but a combination of them, since each classifier obtains the prediction for a subset of test samples, depending on whether they were selected by automatic filtering.

The different approaches were applied to seven well-known unbalanced datasets from the UCI and KEEL repositories, and five real biomedical datasets obtained from the TCGA repository for fixed time event prediction, specifically vital status in cancer. As DA models, the deep generative models CGAN, ACGAN and ModCGAN were used, as well as traditional models of standard noise addition, noise addition for the purpose of oversampling, SMOTE and undersampling. Results for biomedical datasets using a system of SVMs showed an increase in accuracy and AUC using the meta-classifier scheme. It achieves an increase of 1.14 % in predictive capacity and 2.87 %

UNIVERSIDAD DE MÁLAGA

in AUC compared to the values obtained with the original data. Regarding sensitivity and specificity, the results confirm that the application of DA (simple or meta-classifier) improves sensitivity at the cost of obtaining worse specificity. This fact is more visible using the single augmentation approach with an 8.49 % increase in sensitivity and a -0.68 % loss in specificity. Lastly, the MCC results show that for most datasets, it is more beneficial use the meta-classifier scheme than simple augmentation. A great improvement in the overall prediction is observed with an increase of 11.46 % compared to the MCC obtained from the original data.

Tabla B.12: Test results using a deep neural network as a classifier model, trained with the original data, the augmented data and the meta-classifier scheme for the benchmark problems.

Dataset	Method	Acc	Sens	Spec	F_1	MCC	AUC
gbm	Original	0.7746	0.2407	0.9105	0.3024	0.1871	0.6703
	Augmented	0.7838	0.2314	0.9244	0.3028	0.2025	0.6806
	MetaClass	0.7888	0.2407	0.9284	0.3163	0.2209	0.6841
coad	Original	0.8223	0.3633	0.9334	0.4434	0.3561	0.7172
	Augmented	0.8164	0.3913	0.9192	0.4536	0.3531	0.7363
	MetaClass	0.8314	0.3900	0.9382	0.4741	0.3921	0.7363
ucec	Original	0.8201	0.3879	0.9104	0.4271	0.3240	0.7526
	Augmented	0.8290	0.3743	0.9240	0.4307	0.3382	0.7747
	MetaClass	0.8296	0.3857	0.9224	0.4390	0.3454	0.7658
thca	Original	0.9580	0.1075	0.9890	0.1523	0.1488	0.6521
	Augmented	0.9594	0.1113	0.9903	0.1614	0.1631	0.6257
	MetaClass	0.9629	0.1075	0.9940	0.1690	0.1919	0.6651
prad	Original	0.9785	0.0575	0.9956	0.0886	0.0966	0.5864
	Augmented	0.9778	0.0975	0.9941	0.1378	0.1415	0.5894
	MetaClass	0.9802	0.0900	0.9967	0.1420	0.1665	0.5703
Comp.	Augmented	0.29	0.98	0.26	1.45	1.71	0.83
	MetaClass	0.90	1.14	0.82	2.53	4.09	1.27

Results with a deep neural network as classifier also indicate that using the meta-classifier scheme is beneficial. It obtains better average values compared to the results obtained with the original data and the simple augmentation. A 4.09 % increase in MCC stands out compared to the original data, compared to the 1.71 % obtained with the simple augmentation. However, the improvement using DA is less than that obtained with SVM. This again highlights the difficulty present in deep neural networks to improve their performance only through DA in unstructured datasets.

A comparison between the different DA methods applied in each approach, simple or meta-classifier, is also made. For both SVM and neural network, the results show that the application of the DA methods with the meta-classifier scheme improves the test metrics in most cases, compared to its application with the simple augmentation. The best overall prediction

UNIVERSIDAD DE MÁLAGA

Tabla B.13: Results obtained using a deep neural network as a classifier model for the augmented data and meta-classifier scheme with each DA method, compared with original data results.

Method		Acc	Sens	Spec	F_1	MCC	AUC
CGAN	Aumento	-0.26	-0.66	0.02	-0.79	-1.22	-2.17
	MetaClass	0.75	0.05	0.83	1.32	3.00	-0.72
ACGAN	Aumento	-0.20	-0.46	0.05	-0.52	-0.91	-1.12
	MetaClass	0.66	0.19	0.72	1.35	2.79	-0.31
ModCGAN	Aumento	0.12	-0.45	0.44	0.17	0.42	-2.03
	MetaClass	0.55	0.43	0.69	1.66	2.76	0.40
NOISE	Aumento	0.09	-2.40	0.63	-1.67	-0.76	-2.48
	MetaClass	0.55	-0.62	0.73	0.50	2.03	-1.18
NOISE Bal	Aumento	-0.25	3.25	-0.78	2.73	2.00	0.46
	MetaClass	0.76	1.34	0.67	2.63	3.86	0.87
SMOTE	Aumento	-0.60	-1.18	-0.17	-1.37	-1.79	-4.05
	MetaClass	0.42	0.16	0.68	1.31	2.04	-2.03
UNDER	Aumento	-2.62	3.44	-2.34	-1.10	-2.68	-0.95
	MetaClass	0.52	0.10	0.70	1.29	2.20	-1.53

performance is achieved with the addition of noise with oversampling purpose, which obtains a large improvement in sensitivity and a slight improvement in specificity. Regarding the deep generative models, these obtain poor results with the SVM, being improved in their use with the neural network. The ModCGAN model developed for this study with a logistic regression as a Generative Classifier, constitutes the best deep generative model.

With the analysis performed, we can support the hypothesis that the classifiers trained with augmented data greatly increase the sensitivity achieved but worsen the specificity to the same extent. If a deep neural network is used as a classifier, this effect can be reversed depending on the configuration of the augmented data. With our proposal, the difference between sensitivity and specificity seems to decrease and an improvement is obtained regardless of the DA method applied.

B.6. Conclusions

This Ph.D. thesis has covered the application of data augmentation (DA) in problems in the bioinformatics field, for the analysis of genomic sets. For this purpose, different deep learning (DL) models have been used for prediction and data generation processes. General DA methods have been developed, improving the state-of-the-art, as well as specific applications in different relevant medical problems. To do this, noise addition, oversampling and deep generative models have been used. Deep models are fundamentally based on Generative Adversarial Networks (GAN).

UNIVERSIDAD DE MÁLAGA

DL models obtain impressive results in both classification and synthetic imaging. However, we verified that the performance of DL models is reduced when they are applied with genomic sets. This is probably due to not being able to extract spatial information from the data. Thus, in some cases the prediction results are not as high as those obtained using traditional machine learning techniques, like support vector machines or Random Forest, which also have the advantage of requiring exponentially less computational time.

This is also reflected in the creation of samples using deep generative models. In order to be able to use these models with unstructured data and obtain better results, it has been necessary to develop an adaptive model and control the balance of new augmented datasets. In addition, the effect of these deep models is given by certain characteristics of the datasets, as derived from the analyses performed. Thus, the use of GAN models would be recommended if the number of variables is not extremely high, which can be alleviated by the use of feature selection methods. It can also be applied if the number of available samples is not excessively low, causing certain problems in the application of the adaptive model, and if the variables are not binary. Although this last disadvantage can be avoided by a transformation in principal components (PCA).

Regarding data generation process, GAN-based models introduce greater variability in the distribution of the augmented datasets, creating good quality data. However, the internal workings of the discriminator sometimes involve not creating samples that are similar to those real ones that could be considered outliers. These instances may have relevant information for the prediction, and this might not be replicated by the GAN-based models.

It cannot be concluded whether, when performing DA, the classification models benefit from an information gain (creation of new information), or simply because the increase in the number of instances smoothes the learning of the models, increasing their performance. It is concluded that reducing the balance is a fundamental aspect, since it increases the sensitivity of the classifiers models. To take advantage of this effect, the use of a meta-classifier scheme can be effective, combining the increased sensitivity with the high specificity achieved by non-increased dataset. In this regard, the adaptive models with balance control are effective. Although it was shown that the application of a simpler method and with a shorter execution time, such as the addition of noise with oversampling purpose, presents a similar or even better performance in multiple problems.

In conclusion, the effect of DA in bioinformatic problems with genomic datasets has been demonstrated. Likewise, an effective performance of DL models has been demonstrated in prediction and data generation tasks, so their use is recommended as long as the characteristics of the real application allow it (available computing capacity, desired execution time, intrinsic characteristics of the data, etc.).

UNIVERSIDAD DE MÁLAGA

B.7. Future Works

The aforementioned conclusions lead us to choose different future lines of research to extend these works to more bioinformatics problems, omics data analysis and natural language processing (NLP); to investigate possible improvements of the developed models and to compare them with more state-of-the-art DA methods. The fundamentals and advances in the application of DA to omics data analysis have been exposed throughout this Ph.D. thesis. Thus, these cannot be described beyond a concrete framework provided by the problem. For this reason, I take advantage of this last section to delve into DA applied to NLP in the medical/health field.

Similar to computer image processing, the addition of DL to NLP has significantly improved the efficiency of classification and prediction algorithms applied to a wide variety of problems However, as in most real-world applications, the data available for training these deep models is often scarce, limiting the performance. The interest of the DA application with deep generative models in NLP has grown in recent years (Yu et al., 2016; Lin et al., 2017; Caccia et al., 2018; dÁutume et al., 2019).

Currently, health research includes the collection, compilation and analysis of data generated. This type of information is known as "Real World Data" (RWD) and has acquired in recent years a growing and renewed interest beyond the classic observational studies. The generalized, systematic and exhaustive collection of data carried out by clinicians in electronic health records (EHR), and its transformation into useful and valuable information for the clinician, provides a body of knowledge that enriches and complements medical information.

The application of deep generative models has brought new opportunities and great expectations in the analysis of EHRs. These registries store information of a heterogeneous nature in a wide variety of formats that can provide important information to improve medical practices or promote new discoveries about complex diseases (Ribelles et al., 2010; Urda et al., 2013). However, the unstructured nature of these open text fields makes the task of automatically extracting relevant concepts from them especially difficult. It is also difficult to have a balanced number of samples for the types of diseases under study. In this sense, the application of deep generative models can minimize the balance and reduced number of available texts disadvantages.

Starting from the vector spaces of the transformed tokens of the original texts, passing through different embedding models, GAN models can be trained to obtain an augmented dataset that should preserve the statistical distribution observed in the original vector space. With these augmented datasets, it will be possible to proceed with the training of *AI-for-NLP* models in Spanish clinical text to address different NLP tasks (automatic coding, recognition of named entities, etc.).

UNIVERSIDAD DE MÁLAGA

UNIVERSIDAD
DE MÁLAGA

Apéndice C

Publicaciones que avalan la tesis

En este apéndice, resumimos la información asociada con las publicaciones que respaldan la presente tesis de doctorado. Para las revistas se ha utilizado el ranking *Journal Citation Reports* (JCR), mientras que para las conferencias se ha utilizado el *GII-GRIN-SCIE Conference Rating* (GGS), el ranking CORE y el *Research Impact Score*.

C.1. Revistas indexadas JCR

1. Moreno-Barea, F. J., Jerez, J. M., & Franco, L. (2020). Improving classification accuracy using data augmentation on small data sets. *Expert Systems with Applications*, 161, 113696.

   - Factor de Impacto: 8.665 (2021)
   - Categoría:
     - COMPUTER SCIENCE, ARTIFICIAL INTELLIGENCE: Q1
     - OPERATIONS RESEARCH & MANAGEMENT SCIENCE: Q1
     - ENGINEERING, ELECTRICAL & ELECTRONIC: Q1
   - DOI: 10.1016/j.eswa.2020.113696
   - Citado por: 56 (2022)

2. Moreno-Barea, F. J., Franco, L., Elizondo, D., & Grootveld, M. (2022). Application of data augmentation techniques towards metabolomics. *Computers in Biology and Medicine*.

   - Factor de Impacto: 6.698 (2021)

UNIVERSIDAD DE MÁLAGA

- Categoría:
  - COMPUTER SCIENCE, INTERDISCIPLINARY APPLICATIONS: Q1
  - ENGINEERING, BIOMEDICAL: Q1
  - MATHEMATICAL & COMPUTATIONAL BIOLOGY: Q1
  - BIOLOGY: Q1
- DOI: 10.1016/j.compbiomed.2022.105916

C.2. Conferencias Internacionales

1. Moreno-Barea, F. J., Jerez, J. M., & Franco, L. (2022). GAN-based Data Augmentation for prediction improvement using gene expression data in cancer. In *International Conference on Computational Science* (ICCS 2022). Lecture Notes in Computer Science. Springer, Cham.
   - GGS class: 3
   - CORE rating: A
   - Research Impact Score: 1.49
   - DOI: 10.1007/978-3-031-08757-8_3

2. Moreno-Barea, F. J., Franco, L., Elizondo, D., & Grootveld, M. (2022). Data augmentation techniques to improve metabolomic analysis in Niemann-Pick type C disease. In *International Conference on Computational Science* (ICCS 2022). Lecture Notes in Computer Science. Springer, Cham.
   - GGS class: 3
   - CORE rating: A
   - Research Impact Score: 1.49
   - DOI: 10.1007/978-3-031-08757-8_8

3. Moreno-Barea, F. J., Jerez, J. M., & Franco, L. (2022). Meta Classifier Scheme with Data Augmentation in imbalanced datasets. In *2022 IEEE symposium series on computational intelligence* (SSCI). IEEE.
   - GGS class: *En progreso*
   - CORE rating: *En progreso*
   - Research Impact Score: 3.3
   - DOI: Aceptado para publicación

UNIVERSIDAD DE MÁLAGA

4. Moreno-Barea, F. J., Strazzera, F., Jerez, J. M., Urda, D., & Franco, L. (2018). Forward noise adjustment scheme for data augmentation. In *2018 IEEE symposium series on computational intelligence* (SSCI) (pp. 728-734). IEEE.

   - SCImago Journal Rank (SJR): 0.241
   - Categoría:
     - COMPUTER SCIENCE, ARTIFICIAL INTELLIGENCE
     - MATHEMATICS, THEORETICAL COMPUTER SCIENCE
   - DOI: 10.1109/SSCI.2018.8628917
   - Citado por: 82 (2022)

UNIVERSIDAD DE MÁLAGA

UNIVERSIDAD
DE MÁLAGA

Bibliografía

Abadi, M., Agarwal, A., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., Isard, M., Kudlur, M., Levenberg, J., Monga, R., Moore, S., Murray, D. G., Steiner, B., Tucker, P., Vasudevan, V., Warden, P., Wicke, M., Yu, Y. y Zheng, X. TensorFlow: a system for large-scale machine learning. En *12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 16)*, páginas 265–283. 2016. https://dl.acm.org/doi/10.5555/3026877.3026899.

Ackley, D. H., Hinton, G. E. y Sejnowski, T. J. A learning algorithm for boltzmann machines. *Cognitive science*, vol. 9(1), páginas 147–169, 1985. https://doi.org/10.1207/s15516709cog0901_7.

Adolphs, R., Tranel, D., Damasio, H. y Damasio, A. Impaired recognition of emotion in facial expressions following bilateral damage to the human amygdala. *Nature*, vol. 372(6507), página 669, 1994. https://doi.org/10.1038/372669a0.

Aeberhard, S., Coomans, D. y De Vel, O. Comparative analysis of statistical pattern recognition methods in high dimensional settings. *Pattern Recognition*, vol. 27(8), páginas 1065–1077, 1994. https://doi.org/10.1016/0031-3203(94)90145-7.

Ahmad, T., Munir, A., Bhatti, S. H., Aftab, M. y Raza, M. A. Survival analysis of heart failure patients: A case study. *PloS one*, vol. 12(7), página e0181001, 2017. https://doi.org/10.1371/journal.pone.0181001.

Alcalá-Fdez, J., Fernández, A., Luengo, J., Derrac, J., García, S., Sánchez, L. y Herrera, F. Keel data-mining software tool: data set repository, integration of algorithms and experimental analysis framework. *Journal of Multiple-Valued Logic & Soft Computing*, vol. 17, 2011.

Alipanahi, B., Delong, A., Weirauch, M. y Frey, B. Predicting the sequence specificities of dna- and rna-binding proteins by deep learning.

UNIVERSIDAD DE MÁLAGA

Nature Biotechnology, vol. 33(8), páginas 831–838, 2015. https://doi.org/10.1038/nbt.3300.

AMAZON. Amazon aws. 2017. https://s3-ap-south-1.amazonaws.com/av-blog-media/wp-content/uploads/2017/03/06100746/grad.png.

AMIT, Y. y GEMAN, D. Shape quantization and recognition with randomized trees. *Neural computation*, vol. 9(7), páginas 1545–1588, 1997. https://doi.org/10.1162/neco.1997.9.7.1545.

ANGERMUELLER, C., PÄRNAMAA, T., PARTS, L. y STEGLE, O. Deep learning for computational biology. *Molecular systems biology*, vol. 12(7), página 878, 2016. https://doi.org/10.15252/msb.20156651.

ARBIB, M. A. From monkey-like action recognition to human language: An evolutionary framework for neurolinguistics. *Behavioral and brain sciences*, vol. 28(2), páginas 105–124, 2005. https://doi.org/10.1017/s0140525x05000038.

ARCHER, K. J. y KIMES, R. V. Empirical characterization of random forest variable importance measures. *Computational Statistics & Data Analysis*, vol. 52(4), páginas 2249–2260, 2008. https://doi.org/10.1016/j.csda.2007.08.015.

ARJOVSKY, M., CHINTALA, S. y BOTTOU, L. Wasserstein Generative Adversarial Networks. En *International Conference on Machine Learning*, páginas 214–223. 2017. https://dl.acm.org/doi/10.5555/3305381.3305404.

BALDI, P., BRUNAK, S., CHAUVIN, Y., ANDERSEN, C. A. y NIELSEN, H. Assessing the accuracy of prediction algorithms for classification: an overview. *Bioinformatics*, vol. 16(5), páginas 412–424, 2000. https://doi.org/10.1093/bioinformatics/16.5.412.

BALDI, P. y SADOWSKI, P. J. Understanding dropout. *Advances in neural information processing systems*, vol. 26, 2013. https://dl.acm.org/doi/10.5555/2999792.2999926.

BAOWALY, M. K., LIN, C.-C., LIU, C.-L. y CHEN, K.-T. Synthesizing electronic health records using improved generative adversarial networks. *Journal of the American Medical Informatics Association*, vol. 26(3), páginas 228–241, 2019. https://doi.org/10.1093/jamia/ocy142.

BENGIO, Y. Learning deep architectures for ai. *Foundations and trends in Machine Learning*, vol. 2(1), páginas 1–127, 2009. https://dl.acm.org/doi/10.1561/2200000006.

UNIVERSIDAD DE MÁLAGA

Bengio, Y., Lamblin, P., Popovici, D. y Larochelle, H. Greedy layer-wise training of deep networks. En *Advances in neural information processing systems*, páginas 153–160. 2007. https://dl.acm.org/doi/10.5555/2976456.2976476.

Berkson, J. Application of the logistic function to bio-assay. *Journal of the American statistical association*, vol. 39(227), páginas 357–365, 1944. https://doi.org/10.2307/2280041.

Bond, G. L., Hu, W., Bond, E. E., Robins, H., Lutzker, S. G., Arva, N. C., Bargonetti, J., Bartel, F., Taubert, H., Wuerl, P. et al. A single nucleotide polymorphism in the mdm2 promoter attenuates the p53 tumor suppressor pathway and accelerates tumor formation in humans. *Cell*, vol. 119(5), páginas 591–602, 2004. https://doi.org/10.1016/j.cell.2004.11.022.

Bourlard, H. y Kamp, Y. Auto-association by multilayer perceptrons and singular value decomposition. *Biological cybernetics*, vol. 59(4), páginas 291–294, 1988. https://doi.org/10.1007/bf00332918.

Bradley, A. P. The use of the area under the roc curve in the evaluation of machine learning algorithms. *Pattern recognition*, vol. 30(7), páginas 1145–1159, 1997. https://doi.org/10.1016/S0031-3203(96)00142-2.

Brand, R. Microdata protection through noise addition. En *Inference control in statistical databases*, páginas 97–116. Springer, 2002. https://doi.org/10.1007/3-540-47804-3_8.

Breiman, L. Bagging predictors. *Machine learning*, vol. 24(2), páginas 123–140, 1996. https://doi.org/10.1007/bf00058655.

Breiman, L. Random forests. *Machine learning*, vol. 45(1), páginas 5–32, 2001. https://doi.org/10.1023/a:1010933404324.

Breiman, L., Friedman, J. H., Olshen, R. A. y Stone, C. J. *Classification and regression trees*. Routledge, 1984. https://doi.org/10.1201/9781315139470.

van den Broek, M. E., Kägi, D., Ossendorp, F., Toes, R., Vamvakas, S., Lutz, W. K., Melief, C., Zinkernagel, R. M. y Hengartner, H. Decreased tumor surveillance in perforin-deficient mice. *The Journal of experimental medicine*, vol. 184(5), páginas 1781–1790, 1996. https://doi.org/10.1084/jem.184.5.1781.

Burges, C. J. A tutorial on support vector machines for pattern recognition. *Data mining and knowledge discovery*, vol. 2(2), páginas 121–167, 1998. https://doi.org/10.1023/a:1009715923555.

UNIVERSIDAD DE MÁLAGA

BURNET, F. Immunological surveillance in neoplasia. *Immunological Reviews*, vol. 7(1), páginas 3–25, 1971. https://doi.org/10.1111/j.1600-065x.1971.tb00461.x.

BYRD, R. H., HANSEN, S. L., NOCEDAL, J. y SINGER, Y. A stochastic quasi-newton method for large-scale optimization. *SIAM Journal on Optimization*, vol. 26(2), páginas 1008–1031, 2016. https://doi.org/10.1137/140954362.

CACCIA, M., CACCIA, L., FEDUS, W., LAROCHELLE, H., PINEAU, J. y CHARLIN, L. Language GANs Falling Short. *CoRR*, 2018. https://doi.org/10.48550/arXiv.1811.02549.

CAI, C., WANG, S., XU, Y., ZHANG, W., TANG, K., OUYANG, Q., LAI, L. y PEI, J. Transfer learning for drug discovery. *Journal of Medicinal Chemistry*, vol. 63(16), páginas 8683–8694, 2020. https://doi.org/10.1021/acs.jmedchem.9b02147.

CANNY, J. A computational approach to edge detection. *IEEE Transactions on pattern analysis and machine intelligence*, vol. 8(6), páginas 679–698, 1986. https://doi.org/10.1109/tpami.1986.4767851.

CASTANEDA, G., MORRIS, P. y KHOSHGOFTAAR, T. M. Evaluation of maxout activations in deep learning across several big data domains. *Journal of Big Data*, vol. 6(1), páginas 1–35, 2019. https://doi.org/10.1186/s40537-019-0233-0.

CASTRO, C. L. y BRAGA, A. P. Novel cost-sensitive approach to improve the multilayer perceptron performance on imbalanced data. *IEEE transactions on neural networks and learning systems*, vol. 24(6), páginas 888–899, 2013. https://doi.org/10.1109/tnnls.2013.2246188.

CHARYTANOWICZ, M., NIEWCZAS, J., KULCZYCKI, P., KOWALSKI, P. A., ŁUKASIK, S. y ŻAK, S. Complete gradient clustering algorithm for features analysis of x-ray images. En *Information technologies in biomedicine*, páginas 15–24. Springer, 2010. https://doi.org/10.1007/978-3-642-13105-9_2.

CHAUDHARI, P., AGRAWAL, H. y KOTECHA, K. Data augmentation using mg-gan for improved cancer classification on gene expression data. *Soft Computing*, vol. 24(15), páginas 11381–11391, 2020. https://doi.org/10.1007/s00500-019-04602-2.

CHAUDHARY, A. A visual survey of data augmentation in nlp. 2020. https://amitness.com/2020/05/data-augmentation-for-nlp.

UNIVERSIDAD DE MÁLAGA

Chawla, N. V., Bowyer, K. W., Hall, L. O. y Kegelmeyer, W. P. SMOTE: synthetic minority over-sampling technique. *Journal of artificial intelligence research*, vol. 16, páginas 321–357, 2002. https://doi.org/10.1613/jair.953.

Che, Z., Cheng, Y., Zhai, S., Sun, Z. y Liu, Y. Boosting Deep Learning Risk Prediction with Generative Adversarial Networks for Electronic Health Records. En *2017 IEEE International Conference on Data Mining (ICDM)*, páginas 787–792. 2017. https://doi.org/10.1109/icdm.2017.93.

Chellappa, R., Wilson, C. L. y Sirohey, S. Human and machine recognition of faces: A survey. *Proceedings of the IEEE*, vol. 83(5), páginas 705–741, 1995. https://doi.org/10.1109/5.381842.

Chen, J., Mowlaei, M. E. y Shi, X. Population-scale Genomic Data Augmentation Based on Conditional Generative Adversarial Networks. En *Proceedings of the 11th ACM International Conference on Bioinformatics, Computational Biology and Health Informatics*, páginas 1–6. 2020. https://doi.org/10.1145/3388440.3412475.

Chicco, D. y Jurman, G. The advantages of the matthews correlation coefficient (mcc) over f1 score and accuracy in binary classification evaluation. *BMC genomics*, vol. 21(1), páginas 1–13, 2020. https://doi.org/10.1186/s12864-019-6413-7.

Chinchor, N. The statistical significance of the muc-4 results. En *Proceedings of the 4th conference on Message understanding*, páginas 30–50. Association for Computational Linguistics, 1992.

Ching, T., Zhu, X. y Garmire, L. X. Cox-nnet: an artificial neural network method for prognosis prediction of high-throughput omics data. *PLoS computational biology*, vol. 14(4), página e1006076, 2018. https://doi.org/10.1371/journal.pcbi.1006076.

Chong, J., Soufan, O., Li, C., Caraus, I., Li, S., Bourque, G., Wishart, D. S. y Xia, J. MetaboAnalyst 4.0: towards more transparent and integrative metabolomics analysis. *Nucleic acids research*, vol. 46(W1), páginas W486–W494, 2018. https://doi.org/10.1093/nar/gky310.

Chung, J., Gülçehre, C., Cho, K. y Bengio, Y. Gated feedback recurrent neural networks. En *Proceedings of the 32nd International Conference on International Conference on Machine Learning - Volume 37*, página 2067–2075. 2015. https://dl.acm.org/doi/10.5555/3045118.3045338.

UNIVERSIDAD DE MÁLAGA

CIRESAN, D., GIUSTI, A., GAMBARDELLA, L. M. y SCHMIDHUBER, J. Deep neural networks segment neuronal membranes in electron microscopy images. En *Advances in neural information processing systems*, páginas 2843–2851. 2012.

CIRESAN, D., GIUSTI, A., GAMBARDELLA, L. M. y SCHMIDHUBER, J. Mitosis detection in breast cancer histology images with deep neural networks. En *International Conference on Medical Image Computing and Computer-assisted Intervention*, páginas 411–418. Springer, 2013. https://doi.org/10.1007/978-3-642-40763-5_51.

CLARIFI/TECHNOLOGY. 2017. https://www.clarifai.com/.

CLEVERT, D.-A., UNTERTHINER, T. y HOCHREITER, S. Fast and accurate deep network learning by exponential linear units (elus). *CoRR*, 2015. https://doi.org/10.48550/arXiv.1511.07289.

COLEY, W. B. The treatment of malignant tumors by repeated inoculations of erysipelas: with a report of ten original cases. *The American Journal of the Medical Sciences*, vol. 105(5), páginas 487–510, 1893.

COOMANS, D., BROECKAERT, I., JONCKHEER, M. y MASSART, D. Comparison of multivariate discrimination techniques for clinical data—application to the thyroid functional state. *Methods of Information in Medicine*, vol. 22(02), páginas 93–101, 1983. https://doi.org/10.1055/s-0038-1635425.

CORTES, C. y VAPNIK, V. Support-vector networks. *Machine learning*, vol. 20(3), páginas 273–297, 1995. https://doi.org/10.1007/bf00994018.

COUGNOUX, A., CLUZEAU, C., MITRA, S., LI, R., WILLIAMS, I., BURKERT, K., XU, X., WASSIF, C., ZHENG, W. y PORTER, F. Necroptosis in Niemann–Pick disease, type C1: a potential therapeutic target. *Cell death & disease*, vol. 7(3), páginas e2147–e2147, 2016. https://doi.org/10.1038/cddis.2016.16.

COX, D. R. Regression models and life-tables. *Journal of the Royal Statistical Society: Series B (Methodological)*, vol. 34(2), páginas 187–202, 1972.

DANAEE, P., GHAEINI, R. y HENDRIX, D. A. A deep learning approach for cancer detection and relevant gene identification. En *Pacific Symposium on Biocomputing. Pacific Symposium on Biocomputing*, vol. 22, página 219. NIH Public Access, 2016. https://doi.org/10.1142/9789813207813_0022.

DÁUTUME, C. D. M., ROSCA, M., RAE, J. y MOHAMED, S. Training language GANs from Scratch. *CoRR*, 2019. https://doi.org/10.48550/arXiv.1905.09922.

UNIVERSIDAD DE MÁLAGA

Defazio, A., Bach, F. y Lacoste-Julien, S. Saga: A fast incremental gradient method with support for non-strongly convex composite objectives. En *International Conference on Neural Information Processing Systems*, páginas 1646–1654. 2014. https://dl.acm.org/doi/10.5555/2968826.2969010.

Díaz-Uriarte, R. y De Andres, S. A. Gene selection and classification of microarray data using random forest. *BMC bioinformatics*, vol. 7(1), página 3, 2006. https://doi.org/10.1186/1471-2105-7-3.

Dietze, E. C., Sistrunk, C., Miranda-Carboni, G., O'regan, R. y Seewaldt, V. L. Triple-negative breast cancer in African-American women: disparities versus biology. *Nature Reviews Cancer*, vol. 15(4), páginas 248–254, 2015. https://doi.org/10.1038/nrc3896.

Do, H. H., Prasad, P., Maag, A. y Alsadoon, A. Deep learning for aspect-based sentiment analysis: a comparative review. *Expert systems with applications*, vol. 118, páginas 272–299, 2019. https://doi.org/10.1016/j.eswa.2018.10.003.

Domingos, P. Metacost: A general method for making classifiers cost-sensitive. En *Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining*, páginas 155–164. 1999. https://doi.org/10.1145/312129.312220.

Douzas, G. y Bacao, F. Effective data generation for imbalanced learning using conditional generative adversarial networks. *Expert Systems with Applications*, vol. 91, páginas 464–471, 2018. https://doi.org/10.1016/j.eswa.2017.09.030.

Douzas, G., Bacao, F. y Last, F. Improving imbalanced learning through a heuristic oversampling method based on k-means and smote. *Information Sciences*, vol. 465, páginas 1–20, 2018. https://doi.org/10.1016/j.ins.2018.06.056.

Drummond, C., Holte, R. C. et al. C4.5, class imbalance, and cost sensitivity: why under-sampling beats over-sampling. En *Workshop on learning from imbalanced datasets II*, vol. 11, páginas 1–8. Citeseer, 2003.

Dua, D. y Graff, C. UCI machine learning repository. 2017.

Duchi, J., Hazan, E. y Singer, Y. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, vol. 12, páginas 2121–2159, 2011. https://dl.acm.org/doi/10.5555/1953048.2021068.

UNIVERSIDAD DE MÁLAGA

DUGAS, C., BENGIO, Y., BÉLISLE, F., NADEAU, C. y GARCIA, R. Incorporating second-order functional knowledge for better option pricing. En *Advances in Neural Information Processing Systems*, vol. 13, páginas 472–478. MIT Press, 2001. https://dl.acm.org/doi/10.5555/3008751.3008817.

EFRON, B. Bootstrap methods: Another look at the jackknife. *The Annals of Statistics*, páginas 1–26, 1979. https://doi.org/10.1214/aos/1176344552.

EFRON, B. *The jackknife, the bootstrap and other resampling plans*. SIAM, 1982. https://doi.org/10.1137/1.9781611970319.

EHRLICH, P. *Üeber den jetzigen Stand der Karzinomforschung*. Ned Tijdschr Geneeskd, 1908.

EIBEN, A. E., RAUE, P.-E. y RUTTKAY, Z. Genetic algorithms with multi-parent recombination. En *International conference on parallel problem solving from nature*, páginas 78–87. Springer, 1994. https://doi.org/10.1007/3-540-58484-6_252.

ERHAN, D., COURVILLE, A., BENGIO, Y. y VINCENT, P. Why does unsupervised pre-training help deep learning? En *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, páginas 201–208. JMLR Workshop and Conference Proceedings, 2010.

EVETT, I. W. y SPIEHLER, E. J. *Rule Induction in Forensic Science*, página 152–160. Halsted Press, 1989. ISBN 0470212608. https://dl.acm.org/doi/abs/10.5555/67040.67055.

FAJARDO, V. A., FINDLAY, D., JAISWAL, C., YIN, X., HOUMANFAR, R., XIE, H., LIANG, J., SHE, X. y EMERSON, D. On oversampling imbalanced data with deep conditional generative models. *Expert Systems with Applications*, vol. 169, página 114463, 2021. https://doi.org/10.1016/j.eswa.2020.114463.

FAKOOR, R., LADHAK, F., NAZI, A. y HUBER, M. Using deep learning to enhance cancer diagnosis and classification. En *Proceedings of the International Conference on Machine Learning*. 2013.

FAN, W., STOLFO, S. J., ZHANG, J. y CHAN, P. K. Adacost: misclassification cost-sensitive boosting. En *International Conference on Machine Learning*, vol. 99, páginas 97–105. 1999. https://dl.acm.org/doi/10.5555/645528.657651.

FARAGGI, D. y SIMON, R. A neural network model for survival data. *Statistics in medicine*, vol. 14(1), páginas 73–82, 1995. https://doi.org/10.1002/sim.4780140108.

UNIVERSIDAD DE MÁLAGA

Farrés, M., Platikanov, S., Tsakovski, S. y Tauler, R. Comparison of the variable importance in projection (VIP) and of the selectivity ratio (SR) methods for variable selection and interpretation. *Journal of Chemometrics*, vol. 29(10), páginas 528–536, 2015. https://doi.org/10.1002/cem.2736.

Fedus, W., Rosca, M., Lakshminarayanan, B., Dai, A. M., Mohamed, S. y Goodfellow, I. Many paths to equilibrium: Gans do not need to decrease a divergence at every step. En *International Conference on Learning Representations*. 2018. https://doi.org/10.48550/arXiv.1710.08446.

Fernández, A., Garcia, S., Herrera, F. y Chawla, N. V. Smote for learning from imbalanced data: progress and challenges, marking the 15-year anniversary. *Journal of artificial intelligence research*, vol. 61, páginas 863–905, 2018. https://doi.org/10.1613/jair.1.11192.

Fernández-Navarro, F., Hervás-Martínez, C. y Gutiérrez, P. A. A dynamic over-sampling procedure based on sensitivity for multi-class problems. *Pattern Recognition*, vol. 44(8), páginas 1821–1833, 2011. ISSN 0031-3203. https://doi.org/10.1016/j.patcog.2011.02.019.

Fisher, R. A. The use of multiple measurements in taxonomic problems. *Annals of eugenics*, vol. 7(2), páginas 179–188, 1936. https://doi.org/10.1111/j.1469-1809.1936.tb02137.x.

Frid-Adar, M., Diamant, I., Klang, E., Amitai, M., Goldberger, J. y Greenspan, H. GAN-based synthetic medical image augmentation for increased CNN performance in liver lesion classification. *Neurocomputing*, vol. 321, páginas 321–331, 2018. https://doi.org/10.1016/j.neucom.2018.09.013.

Fukushima, K. Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological Cybernetics*, vol. 36(4), páginas 193–202, 1980. https://doi.org/10.1007/bf00344251.

Galar, M., Fernandez, A., Barrenechea, E., Bustince, H. y Herrera, F. A review on ensembles for the class imbalance problem: bagging-, boosting-, and hybrid-based approaches. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 42(4), páginas 463–484, 2011. https://doi.org/10.1109/tsmcc.2011.2161285.

Galar, M., Fernández, A., Barrenechea, E., Bustince, H. y Herrera, F. Ordering-based pruning for improving the performance of

UNIVERSIDAD DE MÁLAGA

ensembles of classifiers in the framework of imbalanced datasets. *Information Sciences*, vol. 354, páginas 178–196, 2016. https://doi.org/10.1016/j.ins.2013.07.007.

GARG, S. y RAMAKRISHNAN, G. Bae: Bert-based adversarial examples for text classification. En *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, páginas 6174–6181. 2020. https://doi.org/10.18653/v1/2020.emnlp-main.498.

GEBERHIWOT, T., MORO, A., DARDIS, A., RAMASWAMI, U., SIRRS, S., MARFA, M. P., VANIER, M. T., WALTERFANG, M., BOLTON, S., DAWSON, C. ET AL. Consensus clinical management guidelines for Niemann-Pick disease type C. *Orphanet journal of rare diseases*, vol. 13(1), páginas 1–19, 2018. https://doi.org/10.1016/j.eswa.2017.09.030.

GERS, F. A., SCHMIDHUBER, J. y CUMMINS, F. Learning to forget: Continual prediction with lstm. *Neural computation*, vol. 12(10), páginas 2451–2471, 2000. https://doi.org/10.1049/cp:19991218.

GLOROT, X. y BENGIO, Y. Understanding the difficulty of training deep feedforward neural networks. En *Aistats*, number 9, páginas 249–256. 2010.

GLOROT, X., BORDES, A. y BENGIO, Y. Deep sparse rectifier neural networks. En *Aistats*, number 15, página 275. 2011.

GOLDHIRSCH, A., WINER, E. P., COATES, A., GELBER, R., PICCART-GEBHART, M., THÜRLIMANN, B., SENN, H.-J., ALBAIN, K. S., ANDRÉ, F., BERGH, J. ET AL. Personalizing the treatment of women with early breast cancer: highlights of the St Gallen International Expert Consensus on the Primary Therapy of Early Breast Cancer 2013. *Annals of oncology*, vol. 24(9), páginas 2206–2223, 2013. https://doi.org/10.1093/annonc/mdt303.

GOODFELLOW, I., BENGIO, Y. y COURVILLE, A. *Deep learning*. MIT press, 2016. http://www.deeplearningbook.org.

GOODFELLOW, I., POUGET-ABADIE, J., MIRZA, M., XU, B., WARDE-FARLEY, D., OZAIR, S., COURVILLE, A. y BENGIO, Y. Generative Adversarial Nets. En *Advances in Neural Information Processing Systems*, páginas 2672–2680. 2014a. https://dl.acm.org/doi/10.5555/2969033.2969125.

GOODFELLOW, I. J., BULATOV, Y., IBARZ, J., ARNOUD, S. y SHET, V. D. Multi-digit number recognition from street view imagery using deep convolutional neural networks. *CoRR*, 2013a. https://doi.org/10.48550/arXiv.1312.6082.

UNIVERSIDAD DE MÁLAGA

Goodfellow, I. J., Mirza, M., Xiao, D., Courville, A. y Bengio, Y. An empirical investigation of catastrophic forgetting in gradient-based neural networks. En *International Conference on Learning Representations*. 2014b. https://doi.org/10.48550/arXiv.1312.6211.

Goodfellow, I. J., Warde-Farley, D., Mirza, M., Courville, A. y Bengio, Y. Maxout networks. *International Conference on Machine Learning*, vol. 28, páginas 1319–1327, 2013b.

Gorman, R. y Sejnowski, T. J. Analysis of hidden units in a layered network trained to classify sonar targets. *Neural Networks*, vol. 1(1), páginas 75–89, 1988. https://doi.org/10.1016/0893-6080(88)90023-8.

Graves, A. Generating sequences with recurrent neural networks. *CoRR*, 2013. https://doi.org/10.48550/arXiv.1308.0850.

Graves, A., Mohamed, A. R. y Hinton, G. Speech recognition with deep recurrent neural networks. En *2013 IEEE international conference on acoustics, speech and signal processing*, páginas 6645–6649. IEEE, 2013. https://doi.org/10.48550/arXiv.1303.5778.

Graves, A. y Schmidhuber, J. Framewise phoneme classification with bidirectional lstm and other neural network architectures. *Neural Networks*, vol. 18(5), páginas 602–610, 2005. https://doi.org/10.1016/j.neunet.2005.06.042.

Graves, A. y Schmidhuber, J. Offline handwriting recognition with multidimensional recurrent neural networks. En *Advances in neural information processing systems*, páginas 545–552. 2008. https://dl.acm.org/doi/10.5555/2981780.2981848.

Grønbech, C. H., Vording, M. F., Timshel, P. N., Sønderby, C. K., Pers, T. H. y Winther, O. scvae: variational auto-encoders for single-cell gene expression data. *Bioinformatics*, vol. 36(16), páginas 4415–4422, 2020. https://doi.org/0.1093/bioinformatics/btaa293.

Grootveld, M. y Silwood, C. J. 1H NMR analysis as a diagnostic probe for human saliva. *Biochemical and Biophysical Research Communications*, vol. 329(1), páginas 1–5, 2005. https://doi.org/10.1016/j.bbrc.2005.01.112.

Grosicki, E. y El Abed, H. Icdar 2009 handwriting recognition competition. En *ICDAR'09. 10th International Conference on Document Analysis and Recognition, 2009.*, páginas 1398–1402. IEEE, 2009. https://doi.org/10.1109/icdar.2009.184.

UNIVERSIDAD DE MÁLAGA

GUAN, J., LI, R., YU, S. y ZHANG, X. Generation of Synthetic Electronic Medical Record Text. En *2018 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, páginas 374–380. 2018. https://doi.org/10.1109/bibm.2018.8621223.

GULRAJANI, I., AHMED, F., ARJOVSKY, M., DUMOULIN, V. y COURVILLE, A. Improved Training of Wasserstein GANs. En *Advances in Neural Information Processing Systems*, páginas 5769–5779. 2017. https://dl.acm.org/doi/10.5555/3295222.3295327.

GUYON, I. y ELISSEEFF, A. An introduction to variable and feature selection. *The Journal of Machine Learning Research*, vol. 3, páginas 1157–1182, 2003. https://dl.acm.org/doi/10.5555/944919.944968.

HA, R., MUTASA, S., KARCICH, J., GUPTA, N., PASCUAL VAN SANT, E., NEMER, J., SUN, M., CHANG, P., LIU, M. Z. y JAMBAWALIKAR, S. Predicting breast cancer molecular subtype with mri dataset utilizing convolutional neural network algorithm. *Journal of Digital Imaging*, vol. 32(2), páginas 276–282, 2019. https://doi.org/10.1007/s10278-019-00179-2.

HAHNLOSER, R. H., SARPESHKAR, R., MAHOWALD, M. A., DOUGLAS, R. J. y SEUNG, H. S. Digital selection and analogue amplification coexist in a cortex-inspired silicon circuit. *Nature*, vol. 405(6789), páginas 947–951, 2000. https://doi.org/10.1038/35016072.

HAIDAR, M. A. y REZAGHOLIZADEH, M. TextKD-GAN: Text Generation using Knowledge Distillation and Generative Adversarial Networks. En *Canadian Conference on Artificial Intelligence*, páginas 107–118. 2019. https://doi.org/10.1007/978-3-030-18305-9_9.

HAN, C., HAYASHI, H., RUNDO, L., ARAKI, R., SHIMODA, W., MURAMATSU, S., FURUKAWA, Y., MAURI, G. y NAKAYAMA, H. GAN-based synthetic brain MR image generation. En *2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018)*, páginas 734–738. IEEE, 2018. https://doi.org/10.1109/isbi.2018.8363678.

HAN, H., WANG, W.-Y. y MAO, B.-H. Borderline-smote: a new over-sampling method in imbalanced data sets learning. En *International conference on intelligent computing*, páginas 878–887. Springer, 2005. https://doi.org/10.1007/11538059_91.

HAN, W., HUANG, Z., LI, S. y JIA, Y. Distribution-sensitive unbalanced data oversampling method for medical diagnosis. *Journal of medical Systems*, vol. 43(2), página 39, 2019. https://doi.org/10.1007/s10916-018-1154-8.

UNIVERSIDAD DE MÁLAGA

Hanley, J. A. y McNeil, B. J. The meaning and use of the area under a receiver operating characteristic (roc) curve. *Radiology*, vol. 143(1), páginas 29–36, 1982. https://doi.org/10.1148/radiology.143.1.7063747.

Hart, P. The condensed nearest neighbor rule (corresp.). *IEEE transactions on information theory*, vol. 14(3), páginas 515–516, 1968. https://doi.org/10.1109/tit.1968.1054155.

Havaei, M., Davy, A., Warde-Farley, D., Biard, A., Courville, A., Bengio, Y., Pal, C., Jodoin, P.-M. y Larochelle, H. Brain tumor segmentation with deep neural networks. *Medical image analysis*, vol. 35, páginas 18–31, 2017. https://doi.org/10.1016/j.media.2016.05.004.

He, H., Bai, Y., Garcia, E. A. y Li, S. Adasyn: Adaptive synthetic sampling approach for imbalanced learning. En *2008 IEEE international joint conference on neural networks (IEEE world congress on computational intelligence)*, páginas 1322–1328. IEEE, 2008. https://doi.org/10.1109/ijcnn.2008.4633969.

He, H. y Garcia, E. A. Learning from imbalanced data. *IEEE Transactions on knowledge and data engineering*, vol. 21(9), páginas 1263–1284, 2009. https://doi.org/10.1109/tkde.2008.239.

He, K., Zhang, X., Ren, S. y Sun, J. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. *CoRR*, 2015. https://doi.org/10.48550/arXiv.1502.01852.

He, K., Zhang, X., Ren, S. y Sun, J. Deep Residual Learning for Image Recognition. En *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, páginas 770–778. 2016. https://doi.org/10.1109/cvpr.2016.90.

Hebb, D. O. *The organization of behavior: A neuropsychological theory*. John Wiley & Sons, 1949. https://doi.org/10.1002/sce.37303405110.

Hecht-Nielsen, R. Theory of the backpropagation neural network. *International Joint Conference on Neural Networks*, vol. 1(Supplement-1), páginas 445–448, 1989. https://doi.org/10.1109/ijcnn.1989.118638.

Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B. y Hochreiter, S. Gans trained by a two time-scale update rule converge to a local nash equilibrium. En *Advances in Neural Information Processing Systems*, vol. 30. 2017. https://dl.acm.org/doi/10.5555/3295222.3295408.

Hinton, G. E. y McClelland, J. L. Learning representations by recirculation. En *Neural information processing systems*, páginas 358–366. 1987. https://dl.acm.org/doi/abs/10.5555/2969644.2969682.

UNIVERSIDAD DE MÁLAGA

HINTON, G. E., OSINDERO, S. y TEH, Y.-W. A fast learning algorithm for deep belief nets. *Neural Comput.*, vol. 18(7), páginas 1527–1554, 2006. ISSN 0899-7667. https://doi.org/10.1162/neco.2006.18.7.1527.

HINTON, G. E. y SALAKHUTDINOV, R. Reducing the dimensionality of data with neural networks. *Science*, vol. 313(5786), páginas 504–507, 2006. https://doi.org/10.1126/science.11276.

HINTON, G. E., SRIVASTAVA, N., KRIZHEVSKY, A., SUTSKEVER, I. y SALAKHUTDINOV, R. R. Improving neural networks by preventing co-adaptation of feature detectors. *CoRR*, 2012a. https://doi.org/10.48550/arXiv.1207.0580.

HINTON, G. E., SRIVASTAVA, N. y SWERSKY, K. Neural Networks for Machine Learning Lecture 6a Overview of mini-batch gradient descent. *Cited on*, vol. 14, página 8, 2012b.

HO, T. K. The random subspace method for constructing decision forests. *IEEE transactions on pattern analysis and machine intelligence*, vol. 20(8), páginas 832–844, 1998. https://doi.org/10.1109/34.709601.

HOCHREITER, S., BENGIO, Y., FRASCONI, P. y SCHMIDHUBER, J. Gradient flow in recurrent nets: the difficulty of learning long-term dependencies. A field guide to dynamical recurrent neural networks. IEEE Press, 2001. https://doi.org/10.1109/9780470544037.ch14.

HOCHREITER, S. y SCHMIDHUBER, J. Long short-term memory. *Neural computation*, vol. 9(8), páginas 1735–1780, 1997. https://doi.org/10.1162/neco.1997.9.8.1735.

HOPFIELD, J. J. Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the national academy of sciences*, vol. 79(8), páginas 2554–2558, 1982. https://doi.org/10.1073/pnas.79.8.2554.

HORNIK, K., STINCHCOMBE, M. y WHITE, H. Multilayer feedforward networks are universal approximators. *Neural networks*, vol. 2(5), páginas 359–366, 1989. https://doi.org/10.1016/0893-6080(89)90020-8.

HOTELLING, H. Analysis of a complex of statistical variables into principal components. *Journal of educational psychology*, vol. 24(6), página 417, 1933. https://doi.org/10.1037/h0071325.

HOU, W.-H., WANG, X.-K., ZHANG, H.-Y., WANG, J.-Q. y LI, L. A novel dynamic ensemble selection classifier for an imbalanced data set: An application for credit risk assessment. *Knowledge-Based Systems*, vol. 208, página 106462, 2020. https://doi.org/10.1016/j.knosys.2020.106462.

UNIVERSIDAD DE MÁLAGA

Hsu, C.-C., Hwang, H.-T., Wu, Y.-C., Tsao, Y. y Wang, H.-M. Voice conversion from unaligned corpora using variational autoencoding wasserstein generative adversarial networks. En *18th Annual Conference of the International Speech Communication Association (INTERSPEECH 2017)*, páginas 3364–3368. 2017a. https://doi.org/10.21437/Interspeech.2017-63.

Hsu, W.-N., Zhang, Y. y Glass, J. Unsupervised domain adaptation for robust speech recognition via variational autoencoder-based data augmentation. En *2017 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, páginas 16–23. IEEE, 2017b. https://doi.org/10.1109/asru.2017.8268911.

Hu, Q., Whitney, H. M. y Giger, M. L. A deep learning methodology for improved breast cancer diagnosis using multiparametric mri. *Scientific reports*, vol. 10(1), páginas 1–11, 2020. https://doi.org/10.1038/s41598-020-67441-4.

Hubel, D. H. y Wiesel, T. N. Receptive fields of single neurones in the cat's striate cortex. *The Journal of physiology*, vol. 148(3), páginas 574–591, 1959. https://doi.org/10.1113/jphysiol.1959.sp006308.

Hubel, D. H. y Wiesel, T. N. Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. *The Journal of physiology*, vol. 160(1), páginas 106–154, 1962. https://doi.org/10.1113/jphysiol.1962.sp006837.

Hubel, D. H. y Wiesel, T. N. Receptive fields and functional architecture of monkey striate cortex. *The Journal of physiology*, vol. 195(1), páginas 215–243, 1968. https://doi.org/10.1113/jphysiol.1968.sp008455.

Ilievski, V. The delicacy of data augmentation in natural language processing. 2020. https://medium.com/ideas-at-igenius/the-delicacy-of-data-augmentation-in-natural-language-processing-nlp-2ef07e9ad1c0.

Im, D. J., Ahn, S., Memisevic, R. y Bengio, Y. Denoising Criterion for Variational Auto-Encoding Framework. En *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, página 2059–2065. 2017. https://dl.acm.org/doi/10.5555/3298483.3298537.

Ioffe, S. y Szegedy, C. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. En *International Conference on Machine Learning*, páginas 448–456. 2015. https://dl.acm.org/doi/10.5555/3045118.3045167.

UNIVERSIDAD DE MÁLAGA

ISOLA, P., ZHU, J.-Y., ZHOU, T. y EFROS, A. A. Image-to-image translation with conditional adversarial networks. En *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, páginas 1125–1134. 2017. https://doi.org/10.1109/cvpr.2017.632.

IVAKHNENKO, A. G. The group method of data handling-a rival of the method of stochastic approximation. *Soviet Automatic Control*, vol. 13(3), páginas 43–55, 1968.

IVAKHNENKO, A. G. Polynomial theory of complex systems. *IEEE transactions on Systems, Man, and Cybernetics*, vol. 1(4), páginas 364–378, 1971. https://doi.org/10.1109/tsmc.1971.4308320.

JAPKOWICZ, N. Assessment metrics for imbalanced learning. *Imbalanced learning: Foundations, algorithms, and applications*, páginas 187–206, 2013. https://doi.org/10.1002/9781118646106.ch8.

JAPKOWICZ, N. y STEPHEN, S. The class imbalance problem: A systematic study. *Intelligent data analysis*, vol. 6(5), páginas 429–449, 2002. https://doi.org/10.3233/ida-2002-6504.

JARRETT, K., KAVUKCUOGLU, K., LECUN, Y. ET AL. What is the best multi-stage architecture for object recognition? En *IEEE 12th International Conference on Computer Vision, 2009*, páginas 2146–2153. IEEE, 2009. https://doi.org/10.1109/iccv.2009.5459469.

JIA, Y., SHELHAMER, E., DONAHUE, J., KARAYEV, S., LONG, J., GIRSHICK, R., GUADARRAMA, S. y DARRELL, T. Caffe: Convolutional architecture for fast feature embedding. En *Proceedings of the 22nd ACM international conference on Multimedia*, páginas 675–678. ACM, 2014. https://doi.org/10.1145/2647868.2654889.

JIAO, X., YIN, Y., SHANG, L., JIANG, X., CHEN, X., LI, L., WANG, F. y LIU, Q. Tinybert: Distilling bert for natural language understanding. En *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings*, páginas 4163–4174. 2020. https://doi.org/10.18653/v1/2020.findings-emnlp.372.

JOLLIFFE, I. T. y CADIMA, J. Principal component analysis: a review and recent developments. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 374(2065), página 20150202, 2016. https://doi.org/10.1098/rsta.2015.0202.

JOZEFOWICZ, R., ZAREMBA, W. y SUTSKEVER, I. An empirical exploration of recurrent network architectures. En *Proceedings of The 32nd International Conference on Machine Learning*, páginas 2342–2350. 2015. https://dl.acm.org/doi/10.5555/3045118.3045367.

UNIVERSIDAD DE MÁLAGA

Kang, G., Dong, X., Zheng, L. y Yang, Y. Patchshuffle regularization. *CoRR*, vol. abs/1707.07103, 2017. https://doi.org/10.48550/arXiv.1707.07103.

Kaplan, D. H., Shankaran, V., Dighe, A. S., Stockert, E., Aguet, M., Old, L. J. y Schreiber, R. D. Demonstration of an interferon γ-dependent tumor surveillance system in immunocompetent mice. *Proceedings of the National Academy of Sciences*, vol. 95(13), páginas 7556–7561, 1998. https://doi.org/10.1073/pnas.95.13.7556.

Karimi, M., Wu, D., Wang, Z. y Shen, Y. Deepaffinity: interpretable deep learning of compound–protein affinity through unified recurrent and convolutional neural networks. *Bioinformatics*, vol. 35(18), páginas 3329–3338, 2019. https://doi.org/10.1093/bioinformatics/btz111.

Karras, T., Aila, T., Laine, S. y Lehtinen, J. Progressive Growing of GANs for Improved Quality, Stability, and Variation. En *International Conference on Learning Representations*. 2018. https://doi.org/10.48550/arXiv.1710.10196.

Kearns, M. y Valiant, L. Cryptographic limitations on learning boolean formulae and finite automata. *Journal of the ACM (JACM)*, vol. 41(1), páginas 67–95, 1994. https://doi.org/10.1145/174644.174647.

Kim, S., Kim, K., Choe, J., Lee, I. y Kang, J. Improved survival analysis by learning shared genomic information from pan-cancer data. *Bioinformatics*, vol. 36(Supplement_1), páginas i389–i398, 2020. https://doi.org/10.1093/bioinformatics/btaa462.

Kingma, D. y Welling, M. Auto-encoding variational bayes. *CoRR*, 2013. https://doi.org/10.48550/arXiv.1312.6114.

Kingma, D. P. y Ba, J. Adam: A Method for Stochastic Optimization. *CoRR*, 2014. https://doi.org/10.48550/arXiv.1412.6980.

Kingma, D. P., Rezende, D. J., Mohamed, S. y Welling, M. Semi-Supervised Learning with Deep Generative Models. En *Advances in neural information processing systems*, páginas 3581–3589. 2014. https://dl.acm.org/doi/10.5555/2969033.2969226.

Klahr, D. y Siegler, R. S. The representation of children's knowledge. En *Advances in child development and behavior*, vol. 12, páginas 61–116. Elsevier, 1978. https://doi.org/10.1016/S0065-2407(08)60036-1.

Klein, J. Origin of major histocompatibility complex polymorphism: the trans-species hypothesis. *Human immunology*, vol. 19(3), páginas 155–162, 1987. https://doi.org/10.1016/0198-8859(87)90066-8.

UNIVERSIDAD DE MÁLAGA

KOHAVI, R. y PROVOST, F. On applied research in machine learning. *Machine Learning*, vol. 10, páginas 1–6, 1998. https://doi.org/10.1023/a:1007442505281.

KOHONEN, T. Correlation matrix memories. *IEEE transactions on computers*, vol. 100(4), páginas 353–359, 1972. https://doi.org/10.1109/tc.1972.5008975.

KOUROU, K., EXARCHOS, T. P., EXARCHOS, K. P., KARAMOUZIS, M. V. y FOTIADIS, D. I. Machine learning applications in cancer prognosis and prediction. *Computational and structural biotechnology journal*, vol. 13, páginas 8–17, 2015. https://doi.org/10.1016/j.csbj.2014.11.005.

KRIZHEVSKY, A., SUTSKEVER, I. y HINTON, G. E. ImageNet Classification with Deep Convolutional Neural Networks. En *Advances in Neural Information Processing Systems*, vol. 25, páginas 1097–1105. 2012. https://doi.org/10.1145/3065386.

KUBAT, M., HOLTE, R. C. y MATWIN, S. Machine learning for the detection of oil spills in satellite radar images. *Machine learning*, vol. 30(2), páginas 195–215, 1998. https://doi.org/10.1023/a:1007452223027.

KUBAT, M., MATWIN, S. ET AL. Addressing the curse of imbalanced training sets: one-sided selection. En *International Conference on Machine Learning*, vol. 97, páginas 179–186. Citeseer, 1997.

KULLBACK, S. y LEIBLER, R. A. On information and sufficiency. *The annals of mathematical statistics*, vol. 22(1), páginas 79–86, 1951. https://doi.org/10.1214/aoms/1177729694.

KUSNER, M. J., PAIGE, B. y HERNÁNDEZ-LOBATO, J. M. Grammar variational autoencoder. En *International Conference on Machine Learning*, páginas 1945–1954. PMLR, 2017. https://dl.acm.org/doi/10.5555/3305381.3305582.

KUTIKOV, A., WEINBERG, D. S., EDELMAN, M. J., HORWITZ, E. M., UZZO, R. G. y FISHER, R. I. A war on two fronts: cancer care in the time of covid-19. 2020. https://doi.org/10.7326/m20-1133.

LECUN, Y. *Modèles connexionnistes de l'apprentissage*. Tesis Doctoral, Université de Paris, 1987.

LECUN, Y., BENGIO, Y. y HINTON, G. E. Deep learning. *Nature*, vol. 521(7553), páginas 436–444, 2015. https://doi.org/10.1038/nature14539.

LECUN, Y., BOSER, B., DENKER, J. S., HENDERSON, D., HOWARD, R. E., HUBBARD, W. y JACKEL, L. D. Backpropagation applied to handwritten

UNIVERSIDAD DE MÁLAGA

zip code recognition. *Neural computation*, vol. 1(4), páginas 541–551, 1989. https://doi.org/10.1162/neco.1989.1.4.541.

LeCun, Y., Bottou, L., Bengio, Y. y Haffner, P. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, vol. 86(11), páginas 2278–2324, 1998. https://doi.org/10.1109/5.726791.

Ledig, C., Theis, L., Huszár, F., Caballero, J., Cunningham, A., Acosta, A., Aitken, A., Tejani, A., Totz, J., Wang, Z. et al. Photo-realistic single image super-resolution using a generative adversarial network. En *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, páginas 4681–4690. 2017. https://doi.org/10.1109/cvpr.2017.19.

Lemaître, G., Nogueira, F. y Aridas, C. K. Imbalanced-learn: A python toolbox to tackle the curse of imbalanced datasets in machine learning. *Journal of Machine Learning Research*, vol. 18(1), página 559–563, 2017. https://dl.acm.org/doi/10.5555/3122009.3122026.

Li, H., Tian, S., Li, Y., Fang, Q., Tan, R., Pan, Y., Huang, C., Xu, Y. y Gao, X. Modern deep learning in bioinformatics. *Journal of molecular cell biology*, vol. 12(11), páginas 823–827, 2020. https://doi.org/10.1093/jmcb/mjaa030.

Li, J., Monroe, W., Shi, T., Jean, S., Ritter, A. y Jurafsky, D. Adversarial Learning for Neural Dialogue Generation. *CoRR*, 2017. https://doi.org/10.48550/arXiv.1701.06547.

Lin, K., Li, D., He, X., Zhang, Z. y Sun, M.-T. Adversarial Ranking for Language Generation. *Advances in Neural Information Processing Systems*, páginas 3158–3168, 2017. https://dl.acm.org/doi/10.5555/3294996.3295075.

Lin, S.-C., Yuan-chin, I. C. y Yang, W.-N. Meta-learning for imbalanced data and classification ensemble in binary classification. *Neurocomputing*, vol. 73(1-3), páginas 484–494, 2009. https://doi.org/10.1016/j.neucom.2009.06.015.

Liu, X.-Y., Wu, J. y Zhou, Z.-H. Exploratory undersampling for class-imbalance learning. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 39(2), páginas 539–550, 2008. https://doi.org/10.1109/tsmcb.2008.2007853.

Liu, Y., Zhou, Y., Liu, X., Dong, F., Wang, C. y Wang, Z. Wasserstein GAN-Based Small-Sample Augmentation for New-Generation Artificial Intelligence: A Case Study of Cancer-Staging Data in Biology. *Engineering*, vol. 5(1), páginas 156–163, 2019. https://doi.org/10.1016/j.eng.2018.11.018.

UNIVERSIDAD DE MÁLAGA

LLOYD-EVANS, E., MORGAN, A. J., HE, X., SMITH, D. A., ELLIOT-SMITH, E., SILLENCE, D. J., CHURCHILL, G. C., SCHUCHMAN, E. H., GALIONE, A. y PLATT, F. M. Niemann-Pick disease type C1 is a sphingosine storage disease that causes deregulation of lysosomal calcium. *Nature medicine*, vol. 14(11), página 1247, 2008. https://doi.org/10.1038/nm.1876.

LÓPEZ, V., FERNÁNDEZ, A., GARCÍA, S., PALADE, V. y HERRERA, F. An insight into classification with imbalanced data: Empirical results and current trends on using data intrinsic characteristics. *Information sciences*, vol. 250, páginas 113–141, 2013. https://doi.org/10.1016/j.ins.2013.07.007.

MA, W., ZHAO, Y., JI, Y., GUO, X., JIAN, X., LIU, P. y WU, S. Breast cancer molecular subtype prediction by mammographic radiomic features. *Academic radiology*, vol. 26(2), páginas 196–201, 2019. https://doi.org/10.1016/j.acra.2018.01.023.

MAAS, A. L., HANNUN, A. Y. y NG, A. Y. Rectifier Nonlinearities Improve Neural Network Acoustic Models. En *International Conference on Machine Learning*, vol. 30(1), página 3. 2013.

VAN DER MAATEN, L. y HINTON, G. Visualizing data using t-sne. *Journal of machine learning research*, vol. 9(11), 2008.

VON DER MALSBURG, C. Self-organization of orientation sensitive cells in the striate cortex. *Biological Cybernetics*, vol. 14(2), páginas 85–100, 1973. https://doi.org/https://doi.org/10.1007/BF00288907.

MAMOSHINA, P., VIEIRA, A., PUTIN, E. y ZHAVORONKOV, A. Applications of deep learning in biomedicine. *Molecular pharmaceutics*, vol. 13(5), páginas 1445–1454, 2016. https://doi.org/10.1021/acs.molpharmaceut.5b00982.

MARCH, S. T. y SMITH, G. F. Design and natural science research on information technology. *Decision support systems*, vol. 15(4), páginas 251–266, 1995. https://doi.org/10.1016/0167-9236(94)00041-2.

MARINGE, C., SPICER, J., MORRIS, M., PURUSHOTHAM, A., NOLTE, E., SULLIVAN, R., RACHET, B. y AGGARWAL, A. The impact of the covid-19 pandemic on cancer deaths due to delays in diagnosis in england, uk: a national, population-based, modelling study. *The lancet oncology*, vol. 21(8), páginas 1023–1034, 2020. https://doi.org/10.1016/S1470-2045(20)30388-0.

MAROUF, M., MACHART, P., BANSAL, V., KILIAN, C., MAGRUDER, D. S., KREBS, C. F. y BONN, S. Realistic in silico generation and augmentation of single-cell RNA-seq data using generative adversarial networks. *Nature*

UNIVERSIDAD DE MÁLAGA

communications, vol. 11(1), páginas 1–12, 2020. https://doi.org/10.1038/s41467-019-14018-z.

Marr, D. y Hildreth, E. Theory of edge detection. *Proceedings of the Royal Society of London B: Biological Sciences*, vol. 207(1167), páginas 187–217, 1980. https://doi.org/10.1098/rspb.1980.0020.

Marshall, D. D. y Powers, R. Beyond the paradigm: Combining mass spectrometry and nuclear magnetic resonance for metabolomics. *Progress in nuclear magnetic resonance spectroscopy*, vol. 100, páginas 1–16, 2017. https://doi.org/10.1016/j.pnmrs.2017.01.001.

Martin, J. D., Cabral, H., Stylianopoulos, T. y Jain, R. K. Improving cancer immunotherapy using nanomedicines: progress, opportunities and challenges. *Nature Reviews Clinical Oncology*, vol. 17(4), páginas 251–266, 2020. https://doi.org/10.1038/s41571-019-0308-z.

Matsuoka, K. Noise injection into inputs in back-propagation learning. *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 22(3), páginas 436–440, 1992. https://doi.org/10.1109/21.155944.

Matthews, B. W. Comparison of the predicted and observed secondary structure of t4 phage lysozyme. *Biochimica et Biophysica Acta (BBA)-Protein Structure*, vol. 405(2), páginas 442–451, 1975. https://doi.org/10.1016/0005-2795(75)90109-9.

McCulloch, W. S. y Pitts, W. A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, vol. 5(4), páginas 115–133, 1943. https://doi.org/10.1007/bf02478259.

McMahan, H. B. Follow-the-regularized-leader and mirror descent: Equivalence theorems and L1 regularization. En *AISTATS*, páginas 525–533. 2011.

Meikle, P. J., Hopwood, J. J., Clague, A. E. y Carey, W. F. Prevalence of lysosomal storage disorders. *Journal of the American medical association*, vol. 281(3), páginas 249–254, 1999. https://doi.org/10.1001/jama.281.3.249.

Melamud, O., Levy, O. y Dagan, I. A simple word embedding model for lexical substitution. En *Proceedings of the 1st Workshop on Vector Space Modeling for Natural Language Processing*, páginas 1–7. 2015. https://doi.org/10.3115/v1/w15-1501.

Michaiski, I. y Chiiausky, R. Learning by being told and learning from examples: an experimental comparison. *International Journal of Policy Analysis Information Systems*, vol. 4(12), página 160, 1980.

UNIVERSIDAD DE MÁLAGA

MIRZA, M. y OSINDERO, S. Conditional Generative Adversarial Nets. *CoRR*, 2014. https://doi.org/10.48550/arXiv.1411.1784.

MIYATO, T. y KOYAMA, M. cgans with projection discriminator. En *International Conference on Learning Representations*. 2018. https://doi.org/10.48550/arXiv.1802.05637.

MONTESINOS-LÓPEZ, O. A., MARTÍN-VALLEJO, J., CROSSA, J., GIANOLA, D., HERNÁNDEZ-SUÁREZ, C. M., MONTESINOS-LÓPEZ, A., JULIANA, P. y SINGH, R. New deep learning genomic-based prediction model for multiple traits with binary, ordinal, and continuous phenotypes. *G3: Genes, genomes, genetics*, vol. 9(5), páginas 1545–1556, 2019. https://doi.org/10.1534/g3.119.300585.

MORENO-BAREA, F. J., FRANCO, L., ELIZONDO, D. y GROOTVELD, M. Application of data augmentation techniques towards metabolomics. *Computers in Biology and Medicine*, vol. 148, página 105916, 2022a. https://doi.org/10.1016/j.compbiomed.2022.105916.

MORENO-BAREA, F. J., FRANCO, L., ELIZONDO, D. y GROOTVELD, M. Data augmentation techniques to improve metabolomic analysis in niemann-pick type c disease. En *International Conference on Computational Science*, páginas 78–91. Springer, 2022b. https://doi.org/10.1007/978-3-031-08757-8_8.

MORENO-BAREA, F. J., JEREZ, J. M. y FRANCO, L. Improving classification accuracy using data augmentation on small data sets. *Expert Systems with Applications*, vol. 161, página 113696, 2020. https://doi.org/10.1016/j.eswa.2020.113696.

MORENO-BAREA, F. J., JEREZ, J. M. y FRANCO, L. Gan-based data augmentation for prediction improvement using gene expression data in cancer. En *International Conference on Computational Science*, páginas 28–42. Springer, 2022c. https://doi.org/10.1007/978-3-031-08757-8_3.

MORENO-BAREA, F. J., JEREZ, J. M. y FRANCO, L. Meta Classifier Scheme with Data Augmentation in imbalanced datasets. En *IEEE Symposium Series on Computational Intelligence (SSCI 2022)*. IEEE, Aceptado para publicación, 2022d.

MORENO-BAREA, F. J., STRAZZERA, F., JEREZ, J. M., URDA, D. y FRANCO, L. Forward Noise Adjustment Scheme for Data Augmentation. En *IEEE Symposium Series on Computational Intelligence (SSCI 2018)*. IEEE, 2018. https://doi.org/10.1109/ssci.2018.8628917.

UNIVERSIDAD DE MÁLAGA

Mujtaba, G., Shuib, L., Idris, N., Hoo, W. L., Raj, R. G., Khowaja, K., Shaikh, K. y Nweke, H. F. Clinical text classification research trends: Systematic literature review and open issues. *Expert systems with applications*, vol. 116, páginas 494–520, 2019. https://doi.org/10.1016/j.eswa.2018.09.034.

Mukkamala, M. C. y Hein, M. Variants of rmsprop and adagrad with logarithmic regret bounds. En *International conference on machine learning*, páginas 2545–2553. PMLR, 2017. https://dl.acm.org/doi/10.5555/3305890.3305944.

Nair, V. y Hinton, G. E. Rectified linear units improve restricted boltzmann machines. En *Proceedings of the 27th International Conference on International Conference on Machine Learning*, páginas 807–814. 2010. https://dl.acm.org/doi/10.5555/3104322.3104425.

Nakai, K. y Kanehisa, M. Expert system for predicting protein localization sites in gram-negative bacteria. *Proteins: Structure, Function, and Bioinformatics*, vol. 11(2), páginas 95–110, 1991. https://doi.org/.

Nakai, K. y Kanehisa, M. A knowledge base for predicting protein localization sites in eukaryotic cells. *Genomics*, vol. 14(4), páginas 897–911, 1992. https://doi.org/10.1016/s0888-7543(05)80111-9.

Narendra, K. S. y Thathachar, M. A. Learning automata-a survey. *IEEE Transactions on systems, man, and cybernetics*, vol. 4, páginas 323–334, 1974. https://doi.org/10.1109/tsmc.1974.5408453.

Nesterov, Y. A method for unconstrained convex minimization problem with the rate of convergence o (1/k2). En *Doklady an SSSR*, número 3 en 269, páginas 543–547. 1983.

Nicholson, R. I., Gee, J. M. W. y Harper, M. E. Egfr and cancer prognosis. *European journal of cancer*, vol. 37, páginas 9–15, 2001. https://doi.org/10.1016/s0959-8049(01)00231-3.

Nielsen, M. A. *Neural Networks and Deep Learning*. Determination Press, 2015.

Nocedal, J. y Wright, S. *Numerical optimization*. Springer Science & Business Media, 2006.

Noordewier, M. O., Towell, G. G. y Shavlik, J. W. Training knowledge-based neural networks to recognize genes in dna sequences. En *Advances in neural information processing systems*, páginas 530–536. 1991. https://dl.acm.org/doi/10.5555/2986766.2986838.

UNIVERSIDAD DE MÁLAGA

ODENA, A., OLAH, C. y SHLENS, J. Conditional image synthesis with auxiliary classifier gans. En *International conference on machine learning*, páginas 2642–2651. PMLR, 2017. https://dl.acm.org/doi/10.5555/3305890.3305954.

PAN, S. J. y YANG, Q. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, vol. 22(10), páginas 1345–1359, 2010. https://doi.org/10.1109/tkde.2009.191.

PARDOLL, D. M. The blockade of immune checkpoints in cancer immunotherapy. *Nature Reviews Cancer*, vol. 12(4), páginas 252–264, 2012. https://doi.org/10.1038/nrc3239.

PASCUAL, S., BONAFONTE, A. y SERRÀ, J. SEGAN: Speech Enhancement Generative Adversarial Network. En *18th Annual Conference of the International Speech Communication Association (INTERSPEECH 2017)*, páginas 3642–3646. 2017. https://doi.org/10.21437/Interspeech.2017-1428.

PEARSON, K. On lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin philosophical magazine and journal of science*, vol. 2(11), páginas 559–572, 1901. https://doi.org/10.1080/14786440109462720.

PEDREGOSA, F., VAROQUAUX, G., GRAMFORT, A., MICHEL, V., THIRION, B., GRISEL, O., BLONDEL, M., PRETTENHOFER, P., WEISS, R., DUBOURG, V. ET AL. Scikit-learn: Machine learning in python. *the Journal of machine Learning research*, vol. 12, páginas 2825–2830, 2011. https://dl.acm.org/doi/10.5555/1953048.2078195.

PENG, Y., WU, Z. y JIANG, J. A novel feature selection approach for biomedical data classification. *Journal of Biomedical Informatics*, vol. 43(1), páginas 15–23, 2010. https://doi.org/10.1016/j.jbi.2009.07.008.

PERCIVAL, B. C., GIBSON, M., WILSON, P. B., PLATT, F. M. y GROOTVELD, M. Metabolomic Studies of Lipid Storage Disorders, with Special Reference to Niemann-Pick Type C Disease: A Critical Review with Future Perspectives. *International journal of molecular sciences*, vol. 21(7), página 2533, 2020. https://doi.org/10.3390/ijms21072533.

PERCIVAL, B. C., LATOUR, Y. L., TIFFT, C. J. y GROOTVELD, M. Rapid Identification of New Biomarkers for the Classification of GM1 Type 2 Gangliosidosis Using an Unbiased 1H NMR-Linked Metabolomics Strategy. *Cells*, vol. 10(3), página 572, 2021. https://doi.org/10.3390/cells10030572.

UNIVERSIDAD DE MÁLAGA

Perou, C. M., Sørlie, T., Eisen, M. B., van de Rijn, M., Jeffrey, S. S., Rees, C. A., Pollack, J. R., Ross, D. T., Johnsen, H., Akslen, L. A. et al. Molecular portraits of human breast tumours. *Nature*, vol. 406(6797), páginas 747–752, 2000. https://doi.org/10.1038/35021093.

Piotrowski, A. P. y Napiorkowski, J. J. A comparison of methods to avoid overfitting in neural networks training in the case of catchment runoff modelling. *Journal of Hydrology*, vol. 476, páginas 97–111, 2013. ISSN 0022-1694. 10.1016/j.jhydrol.2012.10.019.

Platt, F. M., d'Azzo, A., Davidson, B. L., Neufeld, E. F. y Tifft, C. J. Lysosomal storage diseases. *Nature reviews Disease primers*, vol. 4(1), páginas 1–25, 2018. https://doi.org/10.1038/s41572-018-0025-4.

Polyak, B. T. Some methods of speeding up the convergence of iteration methods. *USSR Computational Mathematics and Mathematical Physics*, vol. 4(5), páginas 1–17, 1964. https://doi.org/10.1016/0041-5553(64)90137-5.

Pomyen, Y., Wanichthanarak, K., Poungsombat, P., Fahrmann, J., Grapov, D. y Khoomrung, S. Deep metabolome: Applications of deep learning in metabolomics. *Computational and Structural Biotechnology Journal*, vol. 18, páginas 2818–2825, 2020. https://doi.org/10.1016/j.csbj.2020.09.033.

Prakash, O., Hossain, F., Danos, D., Lassak, A., Scribner, R. y Miele, L. Racial disparities in triple negative breast cancer: a review of the role of biologic and non-biologic factors. *Frontiers in Public Health*, vol. 8, página 576964, 2020. https://doi.org/10.3389/fpubh.2020.576964.

Prechelt, L. Proben1 - A Set of Neural Network Benchmark Problems and Benchmarking Rules. *Technical Report*, vol. 21/94, 1994.

Probert, F., Ruiz-Rodado, V., Te Vruchte, D., Nicoli, E.-R., Claridge, T. D., Wassif, C. A., Farhat, N., Porter, F. D., Platt, F. M. y Grootveld, M. NMR analysis reveals significant differences in the plasma metabolic profiles of Niemann Pick C1 patients, heterozygous carriers, and healthy controls. *Scientific reports*, vol. 7(1), páginas 1–12, 2017. https://doi.org/10.1038/s41598-017-06264-2.

Provost, F. y Fawcett, T. Robust classification for imprecise environments. *Machine learning*, vol. 42(3), páginas 203–231, 2001. https://doi.org/10.1023/a:1007601015854.

UNIVERSIDAD DE MÁLAGA

PU, Y., GAN, Z., HENAO, R., YUAN, X., LI, C., STEVENS, A. y CARIN, L. Variational Autoencoder for Deep Learning of Images, Labels and Captions. En *Advances in neural information processing systems*, páginas 2360–2368. 2016. https://dl.acm.org/doi/10.5555/3157096.3157360.

QUENOUILLE, M. H. Notes on bias in estimation. *Biometrika*, vol. 43(3/4), páginas 353–360, 1956. https://doi.org/10.2307/2332914.

QUINLAN, J. R., COMPTON, P. J., HORN, K. y LAZARUS, L. Inductive knowledge acquisition: a case study. En *Proceedings of the Second Australian Conference on Applications of expert systems*, páginas 137–156. 1987. https://dl.acm.org/doi/abs/10.5555/33429.33438.

RADFORD, A., METZ, L. y CHINTALA, S. Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks. *CoRR*, 2015. https://doi.org/10.48550/arXiv.1511.06434.

RAJESWAR, S., SUBRAMANIAN, S., DUTIL, F., PAL, C. y COURVILLE, A. Adversarial Generation of Natural Language. *CoRR*, páginas 241–251, 2017. https://doi.org/10.48550/arXiv.1705.10929.

RANZATO, M., POULTNEY, C., CHOPRA, S. y LECUN, Y. Efficient learning of sparse representations with an energy-based model. En *Proceedings of the 19th International Conference on Neural Information Processing Systems*, páginas 1137–1144. MIT Press, 2006. https://dl.acm.org/doi/10.5555/2976456.2976599.

REED, R. D. y MARKS, R. J. *Neural Smithing: Supervised Learning in Feedforward Artificial Neural Networks*. Cambridge, MA, USA: MIT press, 1998.

REN, S., HE, K., GIRSHICK, R. y SUN, J. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, vol. 28, páginas 91–99, 2015. https://dl.acm.org/doi/10.5555/2969239.2969250.

REZENDE, D. J., MOHAMED, S. y WIERSTRA, D. Stochastic backpropagation and approximate inference in deep generative models. En *International conference on machine learning*, páginas 1278–1286. PMLR, 2014. https://dl.acm.org/doi/10.5555/3044805.3045035.

RIBELLES, N., JEREZ, J. M., URDA, D., SUBIRATS, J. L., MÁRQUEZ, A., QUERO, C. y FRANCO, L. Galén: Sistema de información para la gestión y coordinación de procesos en un servicio de oncología. *RevistaeSalud*, vol. 6(21), páginas 1–12, 2010.

UNIVERSIDAD DE MÁLAGA

Rosenblatt, F. The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological review*, vol. 65(6), página 386, 1958. https://doi.org/10.1037/h0042519.

Rosenblatt, F. Principles of neurodynamics: perceptrons and the theory of brain mechanisms. Informe técnico, DTIC Document, 1961.

Ruiz-Rodado, V., Marcos Luque-Baena, R., te Vruchte, D., Probert, F., H Lachmann, R., J Hendriksz, C., E Wraith, J., Imrie, J., Elizondo, D., Sillence, D. et al. 1H NMR-linked urinary metabolic profiling of Niemann-Pick Class C1 (NPC1) disease: Identification of potential new biomarkers using correlated component regression (CCR) and genetic algorithm (GA) analysis strategies. *Current Metabolomics*, vol. 2(2), páginas 88–121, 2014. https://doi.org/10.2174/2213235x02666141112215616.

Ruiz-Rodado, V., Nicoli, E.-R., Probert, F., Smith, D. A., Morris, L., Wassif, C. A., Platt, F. M. y Grootveld, M. 1H NMR-linked metabolomics analysis of liver from a mouse model of NP-C1 disease. *Journal of proteome research*, vol. 15(10), páginas 3511–3527, 2016. https://doi.org/10.1021/acs.jproteome.6b00238.

Rumelhart, D. E., Hinton, G. E. y Williams, R. J. Learning representations by back-propagating errors. *Nature*, vol. 323, páginas 533–536, 1986. https://doi.org/10.1038/323533a0.

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A. C. y Fei-Fei, L. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision*, vol. 115(3), páginas 211–252, 2015. https://doi.org/10.1007/s11263-015-0816-y.

Salakhutdinov, R. y Hinton, G. E. Deep boltzmann machines. En *Procedings of the International Conference on Artificial Intelligence and Statistics*, páginas 448–455. 2009.

Salimans, T., Goodfellow, I., Zaremba, W., Cheung, V., Radford, A. y Chen, X. Improved techniques for training gans. *Advances in neural information processing systems*, vol. 29, páginas 2234–2242, 2016. https://dl.acm.org/doi/10.5555/3157096.3157346.

Sandfort, V., Yan, K., Pickhardt, P. J. y Summers, R. M. Data augmentation using generative adversarial networks (CycleGAN) to improve generalizability in CT segmentation tasks. *Scientific reports*, vol. 9(1), páginas 1–9, 2019. https://doi.org/10.1038/s41598-019-52737-x.

UNIVERSIDAD DE MÁLAGA

DOS SANTOS TANAKA, F. H. K. y ARANHA, C. Data Augmentation Using GANs. *Proceedings of Machine Learning Research XXX*, vol. 1, página 16, 2019. https://doi.org/10.48550/arXiv.1904.09135.

SAPOUNTZOGLOU, N., LAGO, J. y RAISON, B. Fault diagnosis in low voltage smart distribution grids using gradient boosting trees. *Electric Power Systems Research*, vol. 182, página 106254, 2020. https://doi.org/10.1016/j.epsr.2020.106254.

SARATHY, R. y MURALIDHAR, K. Evaluating laplace noise addition to satisfy differential privacy for numeric data. *Transactions on Data Privacy*, vol. 4(1), páginas 1–17, 2011. https://dl.acm.org/doi/10.5555/2019312.2019313.

SATIJA, H. *Using deep reinforcement learning for online machine translation*. McGill University (Canada), 2018.

SCHAPIRE, R. E. The strength of weak learnability. *Machine learning*, vol. 5(2), páginas 197–227, 1990. https://doi.org/10.1007/bf00116037.

SCHERER, D., MÜLLER, A. y BEHNKE, S. Evaluation of pooling operations in convolutional architectures for object recognition. En *International Conference on Artificial Neural Networks*, páginas 92–101. Springer, 2010. https://doi.org/10.1007/978-3-642-15825-4_10.

SCHMIDHUBER, J. Deep learning in neural networks: An overview. *Neural networks*, vol. 61, páginas 85–117, 2015. https://doi.org/10.1016/j.neunet.2014.09.003.

SCHMIDT, M., LE ROUX, N. y BACH, F. Minimizing finite sums with the stochastic average gradient. *Mathematical Programming*, vol. 162(1-2), páginas 83–112, 2017. https://doi.org/10.1007/s10107-016-1030-6.

SCHUSTER, M. y PALIWAL, K. K. Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing*, vol. 45(11), páginas 2673–2681, 1997. https://doi.org/10.1109/78.650093.

SENNRICH, R., HADDOW, B. y BIRCH, A. Improving neural machine translation models with monolingual data. En *54th Annual Meeting of the Association for Computational Linguistics*, páginas 86–96. Association for Computational Linguistics (ACL), 2016. https://doi.org/10.18653/v1/p16-1009.

SHAO, S., WANG, P. y YAN, R. Generative adversarial networks for data augmentation in machine fault diagnosis. *Computers in Industry*, vol. 106, páginas 85–93, 2019. https://doi.org/10.1016/j.compindJ.2019.01.001.

UNIVERSIDAD DE MÁLAGA

Sharma, P., Hu-Lieskovan, S., Wargo, J. A. y Ribas, A. Primary, adaptive, and acquired resistance to cancer immunotherapy. *Cell*, vol. 168(4), páginas 707–723, 2017. https://doi.org/10.1016/j.cell.2017.01.017.

Sheng, P., Yang, Z., Hu, H., Tan, T. y Qian, Y. Data augmentation using conditional generative adversarial networks for robust speech recognition. En *2018 11th International Symposium on Chinese Spoken Language Processing (ISCSLP)*, páginas 121–125. IEEE, 2018. https://doi.org/10.1109/iscslp.2018.8706651.

Shi, S., Wang, Q., Xu, P. y Chu, X. Benchmarking state-of-the-art deep learning software tools. En *International Conference on Cloud Computing and Big Data*, vol. 7, páginas 99–104. 2016. https://doi.org/10.1109/ccbd.2016.029.

Shijie, J., Ping, W., Peiyi, J. y Siping, H. Research on data augmentation for image classification based on convolution neural networks. En *Chinese Automation Congress*, páginas 4165–4170. IEEE, 2017. https://doi.org/10.1109/cac.2017.8243510.

Shin, H., Oh, S., Hong, S., Kang, M., Kang, D., Ji, Y.-g., Choi, B. H., Kang, K.-W., Jeong, H., Park, Y. et al. Early-stage lung cancer diagnosis by deep learning-based spectroscopic analysis of circulating exosomes. *ACS nano*, vol. 14(5), páginas 5435–5444, 2020. https://doi.org/10.1021/acsnano.9b09119.

Sigillito, V. G., Wing, S. P., Hutton, L. V. y Baker, K. B. Classification of radar returns from the ionosphere using neural networks. *Johns Hopkins APL Technical Digest*, vol. 10(3), páginas 262–266, 1989.

Silver, D., Huang, A., Maddison, C. J., Guez, A., Sifre, L., Van Den Driessche, G., Schrittwieser, J., Antonoglou, I., Panneershelvam, V., Lanctot, M. et al. Mastering the game of go with deep neural networks and tree search. *Nature*, 2016. https://doi.org/10.1038/nature16961.

Simard, P. Y., Steinkraus, D., Platt, J. C. et al. Best practices for convolutional neural networks applied to visual document analysis. En *International Conference on Document Analysis and Recognition*, vol. 3, páginas 958–962. 2003. https://doi.org/10.1109/icdar.2003.1227801.

Simonyan, K. y Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognition. *CoRR*, 2014. https://doi.org/10.48550/arXiv.1409.1556.

UNIVERSIDAD DE MÁLAGA

SMITH, J. W., EVERHART, J. E., DICKSON, W., KNOWLER, W. C. y JOHANNES, R. S. Using the adap learning algorithm to forecast the onset of diabetes mellitus. En *Proceedings of the annual symposium on computer application in medical care*, página 261. American Medical Informatics Association, 1988.

SMOLENSKY, P. Information processing in dynamical systems: foundations of harmony theory. En *Parallel distributed processing: explorations in the microstructure of cognition, vol. 1*, páginas 194–281. MIT Press, 1986.

SOHN, K., LEE, H. y YAN, X. Learning Structured Output Representation using Deep Conditional Generative Models. En *Advances in neural information processing systems*, páginas 3483–3491. 2015. https://dl.acm.org/doi/10.5555/2969442.2969628.

SØRLIE, T., PEROU, C. M., TIBSHIRANI, R., AAS, T., GEISLER, S., JOHNSEN, H., HASTIE, T., EISEN, M. B., VAN DE RIJN, M., JEFFREY, S. S. ET AL. Gene expression patterns of breast carcinomas distinguish tumor subclasses with clinical implications. *Proceedings of the National Academy of Sciences*, vol. 98(19), páginas 10869–10874, 2001. https://doi.org/10.1073/pnas.191367098.

SRIVASTAVA, N., HINTON, G., KRIZHEVSKY, A., SUTSKEVER, I. y SALAKHUTDINOV, R. Dropout: a simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, vol. 15(1), páginas 1929–1958, 2014. https://dl.acm.org/doi/10.5555/2627435.2670313.

STEIHAUG, T. The conjugate gradient method and trust regions in large scale optimization. *SIAM Journal on Numerical Analysis*, vol. 20(3), páginas 626–637, 1983. https://doi.org/10.1137/0720042.

SU, P.-H., MRKŠIĆ, N., CASANUEVA, I. y VULIĆ, I. Deep learning for conversational ai. En *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Tutorial Abstracts*, páginas 27–32. 2018. https://doi.org/10.18653/v1/n18-6006.

SUNG, H., FERLAY, J., SIEGEL, R. L., LAVERSANNE, M., SOERJOMATARAM, I., JEMAL, A. y BRAY, F. Global cancer statistics 2020: Globocan estimates of incidence and mortality worldwide for 36 cancers in 185 countries. *CA: a cancer journal for clinicians*, vol. 71(3), páginas 209–249, 2021. https://doi.org/10.3322/caac.21660.

SUTSKEVER, I., MARTENS, J., DAHL, G. E. y HINTON, G. E. On the importance of initialization and momentum in deep learning. *International Conference on Machine Learning*, vol. 28, páginas 1139–1147, 2013. https://dl.acm.org/doi/10.5555/3042817.3043064.

UNIVERSIDAD DE MÁLAGA

Sutskever, I., Vinyals, O. y Le, Q. V. Sequence to sequence learning with neural networks. En *Advances in neural information processing systems*, páginas 3104–3112. 2014. https://dl.acm.org/doi/10.5555/2969033.2969173.

Svensson, V., Gayoso, A., Yosef, N. y Pachter, L. Interpretable factor models of single-cell rna-seq via variational autoencoders. *Bioinformatics*, vol. 36(11), páginas 3418–3421, 2020. https://doi.org/10.1093/bioinformatics/btaa169.

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V. y Rabinovich, A. Going deeper with convolutions. En *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, páginas 1–9. 2015. https://doi.org/10.1109/cvpr.2015.7298594.

Szymańska, E., Saccenti, E., Smilde, A. K. y Westerhuis, J. A. Double-check: validation of diagnostic statistics for PLS-DA models in metabolomics studies. *Metabolomics*, vol. 8(1), páginas 3–16, 2012. https://doi.org/10.1007/s11306-011-0330-3.

Talukder, A., Barham, C., Li, X. y Hu, H. Interpretation of deep learning in genomics and epigenomics. *Briefings in Bioinformatics*, vol. 22(3), página bbaa177, 2021. https://doi.org/10.1093/bib/bbaa177.

Tang, B., Tu, Y., Zhang, Z. y Lin, Y. Digital Signal Modulation Classification With Data Augmentation Using Generative Adversarial Nets in Cognitive Radio Networks. *IEEE Access*, vol. 6, páginas 15713–15722, 2018a. https://doi.org/10.1109/access.2018.2815741.

Tang, G., Ni, Y., Wang, K. y Yong, Q. Fine-tuning neural patient question retrieval model with generative adversarial networks. En *Studies in Health Technology and Informatics*, vol. 247, páginas 720–724. 2018b. https://doi.org/10.3233/978-1-61499-852-5-720.

Thomas, L. On immunosurveillance in human cancer. *The Yale journal of biology and medicine*, vol. 55(3-4), página 329, 1982.

Tibshirani, R. Regression shrinkage and selection via the Lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 58(1), páginas 267–288, 1996. https://doi.org/10.1111/j.2517-6161.1996.tb02080.x.

Tissera, M. D. y McDonnell, M. D. Deep extreme learning machines: supervised autoencoding architecture for classification. *Neurocomputing*, vol. 174, páginas 42–49, 2016. https://doi.org/10.1016/j.neucom.2015.03.110.

UNIVERSIDAD DE MÁLAGA

TOMCZAK, K., CZERWIŃSKA, P. y WIZNEROWICZ, M. The cancer genome atlas (tcga): an immeasurable source of knowledge. *Contemporary oncology*, vol. 19(1A), página A68, 2015. https://doi.org/10.5114/wo.2014.47136.

TOMEK, I. An experiment with the edited nearest-neighbor rule. En *IEEE Transactions on Systems, Man, and Cybernetics*, vol. SMC-6, páginas 448–452. 1976. https://doi.org/10.1109/tsmc.1976.4309523.

TSOUKALAS, L. H. y UHRIG, R. E. *Fuzzy and neural approaches in engineering*. John Wiley & Sons, Inc., 1996.

TUKEY, J. Bias and confidence in not quite large samples. *Annals of Mathematical Statistics*, vol. 29, 1958.

UMAROV, R., KUWAHARA, H., LI, Y., GAO, X. y SOLOVYEV, V. Promoter analysis and prediction in the human genome using sequence-based deep learning models. *Bioinformatics*, vol. 35(16), páginas 2730–2737, 2019. https://doi.org/10.1093/bioinformatics/bty1068.

URDA, D., MONTES-TORRES, J., MORENO, F., FRANCO, L. y JEREZ, J. M. Deep learning to analyze rna-seq gene expression data. En *International work-conference on artificial neural networks*, páginas 50–59. Springer, 2017. https://doi.org/10.1007/978-3-319-59147-6_5.

URDA, D., RIBELLES, N., SUBIRATS, J. L., FRANCO, L., ALBA, E. y JEREZ, J. M. Addressing critical issues in the development of an oncology information system. *International journal of medical informatics*, vol. 82(5), páginas 398–407, 2013. https://doi.org/10.1016/j.ijmedinf.2012.08.001.

VALDES-MORA, F., HANDLER, K., LAW, A. M., SALOMON, R., OAKES, S. R., ORMANDY, C. J. y GALLEGO-ORTEGA, D. Single-cell transcriptomics in cancer immunobiology: the future of precision oncology. *Frontiers in immunology*, vol. 9, página 2582, 2018. https://doi.org/10.3389/fimmu.2018.02582.

VAN DEN BOSCH, A., WEIJTERS, A., VAN DEN HERIK, H. J. y DAELEMANS, W. When small disjuncts abound, try lazy learning: A case study. En *Proceedings of the Seventh Belgian-Dutch Conference on Machine Learning*, páginas 109–118. Citeseer, 1997.

VANIER, M. T. Niemann-Pick disease type C. *Orphanet journal of rare diseases*, vol. 5(1), páginas 1–18, 2010. https://doi.org/10.1186/1750-1172-5-16.

UNIVERSIDAD DE MÁLAGA

Verhulst, P. F. Recherches mathématiques sur la loi dáccroissement de la population. *Nouveaux mémoires de lácadémie royale des sciences et belles-lettres de Bruxelles*, vol. 18, páginas 14–54, 1845.

Villani, C. *Optimal transport: old and new*, vol. 338. Springer Science & Business Media, 2008. https://doi.org/10.1007/978-3-540-71050-9.

Vincent, P., Larochelle, H., Bengio, Y. y Manzagol, P.-A. Extracting and composing robust features with denoising autoencoders. En *Proceedings of the 25th international conference on Machine learning*, páginas 1096–1103. 2008. https://doi.org/10.1145/1390156.1390294.

Vincent, P., Larochelle, H., Lajoie, I., Bengio, Y. y Manzagol, P.-A. Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. *Journal of Machine Learning Research*, vol. 11(Dec), páginas 3371–3408, 2010. https://dl.acm.org/doi/10.5555/1756006.1953039.

Waheed, A., Goyal, M., Gupta, D., Khanna, A., Al-Turjman, F. y Pinheiro, P. R. Covidgan: data augmentation using auxiliary classifier gan for improved covid-19 detection. *Ieee Access*, vol. 8, páginas 91916–91923, 2020. https://doi.org/10.1109/access.2020.2994762.

Wallach, I., Dzamba, M. y Heifets, A. Atomnet: a deep convolutional neural network for bioactivity prediction in structure-based drug discovery. *CoRR*, 2015. https://doi.org/10.48550/arXiv.1510.02855.

Wang, S. y Manning, C. Fast dropout training. En *Proceedings of the 30th International Conference on Machine Learning (ICML-13)*, páginas 118–126. 2013. https://dl.acm.org/doi/10.5555/3042817.3042907.

Wang, S., Weng, S., Ma, J. y Tang, Q. Deep cnf-d: Predicting protein order/disorder regions by weighted deep convolutional neural fields. *International Journal of Molecular Sciences*, vol. 16(8), páginas 17315–17330, 2015. https://doi.org/10.3390/ijms160817315.

Wang, Z., Gerstein, M. y Snyder, M. Rna-seq: a revolutionary tool for transcriptomics. *Nature reviews genetics*, vol. 10(1), páginas 57–63, 2009. https://doi.org/10.1038/nrg2484.

Warde-Farley, D., Goodfellow, I. J., Courville, A. y Bengio, Y. An empirical analysis of dropout in piecewise linear networks. En *Proceedings of the 2nd International Conference on Learning Representations (ICLR)*. 2014. https://doi.org/10.48550/arXiv.1312.6197.

Wei, J. y Zou, K. Eda: Easy data augmentation techniques for boosting performance on text classification tasks. En *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th*

UNIVERSIDAD DE MÁLAGA

International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), páginas 6382–6388. 2019. https://doi.org/10.48550/arXiv.1901.11196.

WERBOS, P. J. *Beyond Regression: New Tools for Prediction and Analysis in the Behavioral Sciences*. Tesis Doctoral, Harvard University, 1974.

WILLSHAW, D. J. y VON DER MALSBURG, C. How patterned neural connections can be set up by self-organization. *Proceedings of the Royal Society of London B: Biological Sciences*, vol. 194(1117), páginas 431–445, 1976. https://doi.org/10.1098/rspb.1976.0087.

WINKLER, M. B., KIDMOSE, R. T., SZOMEK, M., THAYSEN, K., RAWSON, S., MUENCH, S. P., WÜSTNER, D. y PEDERSEN, B. P. Structural insight into eukaryotic sterol transport through Niemann-Pick type C proteins. *Cell*, vol. 179(2), páginas 485–497, 2019. https://doi.org/10.1016/j.cell.2019.08.038.

WISHART, D. S., FEUNANG, Y. D., MARCU, A., GUO, A. C., LIANG, K., VÁZQUEZ-FRESNO, R., SAJED, T., JOHNSON, D., LI, C., KARU, N. ET AL. Hmdb 4.0: the human metabolome database for 2018. *Nucleic acids research*, vol. 46(D1), páginas D608–D617, 2018. https://doi.org/10.1093/nar/gkx1089.

WOLBERG, W. H. y MANGASARIAN, O. L. Multisurface method of pattern separation for medical diagnosis applied to breast cytology. *Proceedings of the national academy of sciences*, vol. 87(23), páginas 9193–9196, 1990. https://doi.org/10.1073/pnas.87.23.9193.

WU, E., WU, K., COX, D. y LOTTER, W. Conditional Infilling GANs for Data Augmentation in Mammogram Classification. En *Image Analysis for Moving Organ, Breast, and Thoracic Images*, páginas 98–106. Springer, 2018. https://doi.org/10.1007/978-3-030-00946-5_11.

XIAO, Y., WU, J. y LIN, Z. Cancer diagnosis using generative adversarial networks based on deep learning from imbalanced data. *Computers in Biology and Medicine*, página 104540, 2021. https://doi.org/10.1016/j.compbiomed.2021.104540.

XIE, Q., DAI, Z., HOVY, E., LUONG, T. y LE, Q. Unsupervised data augmentation for consistency training. *Advances in Neural Information Processing Systems*, páginas 6256–6268, 2020. https://dl.acm.org/doi/abs/10.5555/3495724.3496249.

XU, B., WANG, N., CHEN, T. y LI, M. Empirical Evaluation of Rectified Activations in Convolutional Network. *CoRR*, 2015. https://doi.org/10.48550/arXiv.1505.00853.

UNIVERSIDAD DE MÁLAGA

Xu, W., Sun, H., Deng, C. y Tan, Y. Variational autoencoder for semi-supervised text classification. En *31st AAAI Conference on Artificial Intelligence*, páginas 3358—3364. 2017. https://doi.org/10.1609/aaai.v31i1.10966.

Yao, L., Mao, C. y Luo, Y. Graph convolutional networks for text classification. En *Proceedings of the AAAI conference on artificial intelligence*, vol. 33-01, páginas 7370–7377. 2019. https://doi.org/10.1609/aaai.v33i01.33017370.

Yu, J., Li, Y., Zheng, X., Zhong, Y. y He, P. An effective cloud detection method for gaofen-5 images via deep learning. *Remote Sensing*, vol. 12(13), página 2106, 2020. https://doi.org/10.3390/rs12132106.

Yu, L., Zhang, W., Wang, J. y Yu, Y. SeqGAN: Sequence Generative Adversarial Nets with Policy Gradient. En *31st AAAI Conference on Artificial Intelligence*, páginas 2852–2858. 2016. https://doi.org/10.1609/aaai.v31i1.10804.

Yue, L., Shen, H., Li, J., Yuan, Q., Zhang, H. y Zhang, L. Image super-resolution: The techniques, applications, and future. *Signal Processing*, vol. 128, páginas 389–408, 2016. https://doi.org/10.1016/j.sigpro.2016.05.002.

Zeiler, M. D. Adadelta: an adaptive learning rate method. *CoRR*, 2012. https://doi.org/10.48550/arXiv.1212.5701.

Zeng, H., Edwards, M., Liu, G. y Gifford, D. Convolutional neural network architectures for predicting dna–protein binding. *Bioinformatics*, vol. 32(12), páginas 121–127, 2016. https://doi.org/10.1093/bioinformatics/btw255.

Zhang, Y., Liu, G., Luan, W., Yan, C. y Jiang, C. An approach to class imbalance problem based on stacking and inverse random under sampling methods. En *IEEE 15th International Conference on Networking, Sensing and Control*, páginas 1–6. IEEE, 2018. https://doi.org/10.1109/icnsc.2018.8361344.

Zhao, D., Zhu, D., Lu, J., Luo, Y., Zhang, G., Zhao, D., Zhu, D., Lu, J., Luo, Y. y Zhang, G. Synthetic Medical Images Using F&BGAN for Improved Lung Nodules Classification by Multi-Scale VGG16. *Symmetry*, vol. 10(10), página 519, 2018. https://doi.org/10.3390/sym10100519.

Zheng, Z., Wu, X. y Srihari, R. Feature selection for text categorization on imbalanced data. *ACM Sigkdd Explorations Newsletter*, vol. 6(1), páginas 80–89, 2004. https://doi.org/10.1145/1007730.1007741.

UNIVERSIDAD DE MÁLAGA

ZHONG, Z., ZHENG, L., KANG, G., LI, S. y YANG, Y. Random erasing data augmentation. En *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34 (07), páginas 13001–13008. 2020. https://doi.org/10.1609/aaai.v34i07.7000.

ZHOU, Y. y CHELLAPPA, R. Computation of optical flow using a neural network. En *IEEE International Conference on Neural Networks*, páginas 71–78. 1988. https://doi.org/10.1109/icnn.1988.23914.

ZHU, J.-Y., PARK, T., ISOLA, P. y EFROS, A. A. Unpaired image-to-image translation using cycle-consistent adversarial networks. En *Proceedings of the IEEE international conference on computer vision*, páginas 2223–2232. 2017. https://doi.org/10.1109/iccv.2017.244.

ZHU, W., XIE, L., HAN, J. y GUO, X. The application of deep learning in cancer prognosis prediction. *Cancers*, vol. 12(3), página 603, 2020. https://doi.org/10.3390/cancers12030603.

ZHU, X., LIU, Y., LI, J., WAN, T. y QIN, Z. Emotion classification with data augmentation using generative adversarial networks. En *Pacific-Asia conference on knowledge discovery and data mining*, páginas 349–360. Springer, 2018. https://doi.org/10.1007/978-3-319-93040-4_28.

ZUR, R. M., JIANG, Y., PESCE, L. y DRUKKER, K. Noise injection for training artificial neural networks: A comparison with weight decay and early stopping. *Medical physics*, vol. 36(10), páginas 4810–4818, 2009. https://doi.org/10.1118/1.3213517.

UNIVERSIDAD DE MÁLAGA

Nada tiene tanto poder para ampliar
la mente que la capacidad de investigar
de manera sistemática y real todo lo que es
susceptible de observación en la vida.

Marco Aurelio

La utopía está en el horizonte.
Camino dos pasos, ella se aleja dos pasos
y el horizonte se corre diez pasos más allá.
¿Entonces para que sirve la utopía?
Para eso, sirve para caminar.

Eduardo Galeano

UNIVERSIDAD
DE MÁLAGA