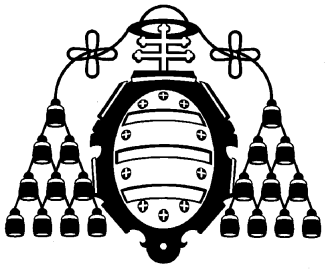
UNIVERSIDAD DE OVIEDO



Departamento de Ingeniería Informática
Centro de Inteligencia Artificial

Machine learning algorithms and experimentation methods applied to sample quantification

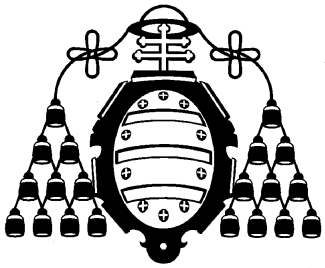
Ph.D. Thesis in Computer Science

José Barranquero Tolosa

Ph.D. Supervisors
Dr. Juan José del Coz Velasco
Dr. Jorge Díez Peláez

2013

UNIVERSIDAD DE OVIEDO



Departamento de Ingeniería Informática
Centro de Inteligencia Artificial

Machine learning algorithms and experimentation methods applied to sample quantification

Ph.D. Thesis in Computer Science

José Barranquero Tolosa

Ph.D. Supervisors
Dr. Juan José del Coz Velasco
Dr. Jorge Díez Peláez

2013

Agradecimientos

La investigación que se recoge en esta tesis ha sido subvencionada en parte por el Ministerio de Economía y Competitividad (MINECO) a través del proyecto de investigación TIN2011-23558, y por el Ministerio de Ciencia e Innovación (MICINN) a través del proyecto de investigación TIN2008-06247 y la beca FPI BES-2009-027102.

Me gustaría mostrar mi más sincero y cariñoso agradecimiento a mis directores, Juan José y Jorge, y al resto de compañeros del Centro de Inteligencia Artificial de la Universidad de Oviedo, por su inestimable apoyo, tanto a nivel técnico como humano, máxime durante estos duros y tristes años que está sufriendo la investigación en España.

A pesar de que lamento profundamente no poder seguir desarrollando mi investigación entre las mismas paredes que han dado vida a esta tesis, me gustaría agradecer también la desafiante oportunidad de que me han brindado mis compañeros del European Centre for Soft Computing; en especial a Sergio y Manuel, quienes también me han apoyado, casi diría que empujado, para presentar esta tesis definitivamente.

Dedicada mi hija, Ainoa, por ser el Sol que ilumina de sonrisas todos y cada uno de mis días; y a mi mujer, Sheila, por ser la Luna que arroja claridad y guarda mi cordura incluso en las noches más oscuras

Resumen

Existe una creciente demanda de métodos eficientemente rentables para la estimación de la distribución de las clases en una muestra de individuos. Una tarea de aprendizaje automático recientemente formalizada como cuantificación. Su principal objetivo es la estimación precisa del número de casos positivos (o distribución de clases) en un conjunto de evaluación, empleando un conjunto de entrenamiento que puede tener una distribución sustancialmente distinta.

Tras analizar la superficie del problema, la conclusión directa podría ser que cuantificar la proporciones de las clases en una muestra es tan simple como contar las predicciones de un clasificador estándar. Sin embargo, debido a los cambios de distribución que ocurren de forma natural en los problemas del mundo real esta solución suele ser poco efectiva, dado que las diferencias de distribución entre los conjuntos de entrenamiento y evaluación pueden tener un gran impacto negativo en la efectividad de clasificadores de vanguardia.

La suposición general que suelen establecer los métodos de clasificación es que las muestras son representativas, lo cual implica que las densidades intra-clases, $Pr(\boldsymbol{x}|y)$, y la distribución a priori de las clases, $Pr(y)$, son invariables. Obviamente, la segunda parte no se cumple para cuantificación, dado que por definición su objetivo es estimar $Pr(y)$. En esta tesis se estudia este problemática y el trabajo relacionado con cuantificación bajo condiciones de prior-shift, en donde sólo las densidades intra-clases se consideran constates.

Esta tesis propone tres contribuciones principales: (1) se presenta el primer estudio de investigación que formaliza una metodología específica para realizar comparativas estadísticas de varios cuantificadores evaluados sobre múltiples prevalencias; (2) se validan dos estrategias sencillas y computacionalmente rentables de ponderación por pesos aplicadas a algoritmos del vecino más cercano, las cuales resultan competitivas empíricamente; y (3) se implementa el primer método de aprendizaje que optimiza una métrica de cuantificación, proponiendo una nueva familia de funciones de pérdida parametrizables, capaces de balancear medidas de cuantificación y clasificación simultáneamente.

La contribución principal de la metodología propuesta es que nos permite analizar propiedades relevantes de estas comparativas desde un punto de vista estadístico, a la vez que ofrece indicadores sobre qué algoritmos son significativamente mejores. En comparación con la validación-cruzada estándar y los tests estadísticos relacionados, la validación de cuantificadores requiere medir su eficacia sobre un

gran abanico de conjuntos de evaluación con diferentes distribuciones de clases. Este es el motivo por el que las comparativas estadísticas de clasificadores no son aplicables directamente.

La segunda contribución ofrece un nuevo método de base para resolver problemas de cuantificación binaria, basado en algoritmos del vecino más próximo (NN). Se presentan dos estrategias sencillas y computacionalmente rentables de ponderación por pesos, que destacan entre modelos de cuantificación recientes. Las conclusiones extraídas de los test estadísticos de Nemenyi muestran que nuestras propuestas son las únicas de entre las estudiadas que ofrecen diferencias significativas con respecto a modelos menos robustos, como son CC, AC o T50; los cuales están considerados como enfoques vanguardistas por la mayoría de autores.

Nuestra última contribución está relacionada con el hecho de que los modelos actuales de cuantificación se basan en clasificadores, presentando la problemática de que son entrenados con una función de pérdida enfocada a clasificación, en lugar de a cuantificación. Otros intentos recientes de abordar este problema sufren ciertas limitaciones en cuanto a fiabilidad. Por lo que presentamos un método de aprendizaje que optimiza un métrica basada en estimación sobre muestras completas, combinando medidas de cuantificación y clasificación simultáneamente. Nuestra propuesta ofrece un nuevo enfoque que permite construir cuantificadores binarios que son capaces de estimar la proporción de positivos de forma precisa, basados a su vez en modelos con habilidades de clasificación fiables.

En el último capítulo se analizan las conclusiones principales, presentando directrices para futuras investigaciones; incluyendo la optimización de modelos de base que minimice la varianza durante la calibración de sus umbrales de decisión y el análisis empírico de la potencia y estabilidad de test estadísticos aplicados en experimentos de cuantificación.

"En tiempos de cambio, quienes estén abiertos al aprendizaje heredarán el futuro, mientras que aquellos que crean saberlo todo se encontrarán excelentemente equipados para vivir en un mundo que ya no existe" — Eric Hoffer

Conclusiones

Las tres contribuciones principales de esta tesis son las siguientes:

- Presentación del primer estudio de investigación que formaliza una metodología especializada en comparaciones estadísticas de varios cuantificadores sobre múltiples distribuciones de evaluación [Barranquero et al., 2013].
- Diseño e implementación de dos estrategias de ponderación por pesos, sencillas y eficientes, para algoritmos del vecino más próximo; las cuales ofrecen un redimiendo competitivo en tares de cuantificación [Barranquero et al., 2013].
- Diseño e implementación del primer método de aprendizaje que optimiza una métrica de cuantificación, proponiendo además una familia de funciones de pérdida parametrizable, las cuales son capaces de balancear criterios de cuantificación y clasificación [Barranquero et al., under review].

Metodología para comparación estadística de cuantificadores

Dado que la metodología de experimentación que requiere la tarea de cuantificación es relativamente infrecuente y todavía necesita estandarizarse y validarse por la comunidad de aprendizaje automático, en el Capítulo 4 proponemos una nueva metodología para comparaciones estadísticas de varios cuantificadores sobre múltiples distribuciones de evaluación a través de re-muestreo estratificado.

La contribución principal de esta nueva metodología es que nos permite analizar propiedades relevantes de estas comparativas desde un punto de vista estadístico. Además, también ofrece indicios sobre qué algoritmos son significativamente mejores, con un cierto grado de confianza, gracias a la adaptación de los dos test estadísticos post-hoc basados en el test de Friedman propuestos por Demšar [2006], y el re-diseño del procedimiento de generación de conjuntos de evaluación de la validación-cruzada estratificada de k-particiones.

La principal diferencias con respecto a la validación-cruzada estándar y los test estadísticos relacionados es que es necesario evaluar el rendimiento sobre conjuntos completos, en lugar de sobre resultados individuales de clasificación. Más aún, dicha valoración requiere evaluar el rendimiento sobre un amplio espectro de distribuciones de test, en contrapartida a usar un único conjunto de test. Es

por esto que las técnicas tradicionales para realizar comparativas de modelos de clasificación no son directamente aplicables y necesitan ser adaptadas.

Por todo ello realizamos una revisión exhaustiva de estos procedimientos estadísticos relacionados, discutiendo sus pros y contras. Tras este estudio describimos nuestra propuesta en detalle, adaptando los procedimientos existentes a los requerimientos específicos de las comparativas de cuantificación. En este sentido, consideramos que unos de los puntos fuertes de nuestra propuesta es que hemos prevalecido la robustez, en términos de reducir los errores Tipo I, frente a reducir los errores de Tipo II (ver Sección 4.2.4).

Cuantificación por vecinos más próximos

En el Capítulo 5, presentamos varios algoritmos basados en reglas tipo vecino más próximo, incluido el ampliamente conocido KNN y dos estrategias de ponderación identificadas como PWK y PWK$^{\alpha}$. Partiendo del objetivo principal de estudiar el comportamiento de métodos NN en el contexto de la cuantificación, proponemos un enfoque capaz de proveer rendimiento competitivo con un balance entre simplicidad y efectividad. Este estudio establece un nuevo enfoque baseline para afrontar el problema de la estimación de prevalencia en problemas binarios.

Encontramos que, en general, las versiones ponderadas basadas en NN ofrecen un rendimiento eficiente y de bajo coste. Las conclusiones extraídas de los test Nemenyi analizados en la Sección 5.3 sugieren que PWK y PWK$^{\alpha}$ destacan como los mejores sistemas, sin diferencias estadísticas entre ambos, pero sí ofrecen diferencias estadísticas con respecto a modelos menos robustos como CC, AC o T50.

Nuestros experimentos no ofrecen ningún indicador que ayude a discriminar cuál de estas dos estrategias de ponderación es más recomendable en aplicaciones del mundo real. La decisión final debería basarse en los requerimientos específicos del problema, las restricción del entorno, o la complejidad de los datos, entre otros.

Sin embargo, teniendo en cuenta las observaciones que se discuten en la Sección 5.3.5, parece que PWK puede ser más apropiado cuando la clase minoritaria es más relevante, mientras que PWK$^{\alpha}$ aparenta ser mucho más conservativo respecto a la clase mayoritaria. Teniendo en cuenta otros factores, PWK es más sencillo, sus pesos son más fácilmente interpretables y sólo requiere calibrar el número de vecinos.

Cuantificación mediante funciones de pérdida robustas

Finalmente, en el Capítulo 6 estudiamos el problema desde una perspectiva totalmente diferente. Tal y como Esuli y Sebastiani afirman en [Esuli and

Sebastiani, 2010], los algoritmos de cuantificación actuales no optimizan la función de pérdida que se emplea en la validación o comparación. Continuando su línea de investigación, consideramos que optimizar sólo una métrica de cuantificación durante el aprendizaje no aborda el problema de forma suficiente. Podríamos obtener cuantificadores con un comportamiento pobre, debido a que el modelo subyacente es incoherente en términos de habilidades de clasificación (ver Sección 6.1). En este sentido, la pregunta más importante que trata de responder este estudio es si es realmente recomendable confiar en modelos de cuantificación que que no distingan entre positivos y negativos a nivel individual.

Formalmente, la forma de resolver problemas de aprendizaje automático implica dos pasos: definir una métrica apropiada y un algoritmo que la optimice. Por tanto, la combinación de *Q-measure*, definida en la Sección 6.2, y el algoritmo multivariate de Joachims [2005], presentado in la Sección 6.3, ofrecen una solución formal para el aprendizaje de cuantifiacadores.

Las contribuciones principales son el estudio del primer procedimiento de aprendizaje orientado a cuantificación, es decir, el primer algoritmo que optimiza una métrica de cuantificación; y la definición de una función de pérdida parametrizable. Esta propuesta no sólo está teóricamente bien fundamentada, sino que además ofrece rendimiento competitivo en comparación con los algoritmos de cuantificación actuales.

Overall discussion of contributions

A pesar de que las propuestas basadas en NN pueden parecer técnicamente simples, es importante valorar el esfuerzo invertido en analizar el problema de cara a poder adaptar estos algoritmos a una tarea de optimización moderna. Este estudio también nos ha ayudado a entender el problema más profundamente.

Adicionalmente, el valor de soluciones simples ha sido recalcada mucha veces en la literatura. Uno no podría decir nunca a priori, cuánto de la estructura de un dominio puede ser captura por una regla de decisión sencilla. Puede ser que además sea ventajosa por cuestiones teóricas o practicas.

De hecho, lo modelos simples (iniciales) usualmente ofrecen las mayores mejoras, que pueden superar el 90% del poder predictivo que puede conseguirse, mientras que son menos propensos a sobreajustarse [Hand, 2006; Holte, 2006]. Obviamente, esto no significa que haya que menospreciar las reglas de decisión más complejas, sino que las sencillas no deberían descartarse. Este podría ser el caso de nuestras soluciones NN, que se basan la distancia euclídea y estrategias de ponderación simples. Sin embargo, tampoco significa que los enfoques NN no puedan proveer reglas de decisión más complejas a su vez.

Por el contrario, el enfoque multivariate para optimizar *Q-measure* podría sufrir de demasiada complejidad, medida en términos de coste computacional. Esta es una

de las razones por las que no hemos aplicado la corrección de Forman durante la validación de experimentos, resultando que estos experimentos no son comparables directamente con el estudio previo sobre NN. En cualquier caso consideramos que ambos enfoques son complementarios.

Curiosamente, Demšar [2006] dirige la atención a una opinión alternativa entre los expertos en estadística acerca de que los test de significancia no deberían realizares en ningún caso porque normalmente se emplean mal, bien por mala interpretación o por dar demasiado peso a los resultados. De todas formas, considera que los test estadísticos proveen de cierta validez y no-aleatoriedad de resultados públicos, aunque, deberían realizarse con cuidado.

El punto más relevante remarcado por Demšar es que los test estadísticos no deberían ser el factor decisivo a favor o en cuentea de un trabajo. Deberían valorarse otros méritos del algoritmo propuesto más allá de los resultados de estos tests.

En este sentido, nuestro enfoque *Q-measure* ofrece resultados de cuantificación competitivos con respecto al resto de modelos. Sin embargo, como ya hemos discutido, está formalmente bien definido y no se basa en ninguna regla heurística. Desde ese punto de vista creemos firmemente que pueda ser considerado en futuros estudios de cuantificación. Por lo menos ofrecen un sesgo de aprendizaje diferente a los actuales, pudiendo producir mejores resultados en algunos dominios.

Finalmente, es importante remarcar que los resultados experimentales de *Q-measure* no están ajustados por medio de la Equación (3.1). Por lo que estos métodos pueden considerarse variantes del CC, pudiendo mejorarse con estrategias similares a las aplicadas en AC, Max, X, MS or T50.

Abstract

There is an increasing demand in real-world applications for cost-effective methods to estimate the distribution of classes from a sample. A machine learning task that has been recently formalized as quantification. Its key objective is to accurately estimate the number of positive cases (or class distribution) in a test set, using a training set that may have a substantially different distribution.

After a first look at the problem, the straightforward conclusion could be that quantifying the proportions of classes in a sample is as simple as counting the predictions of a standard classifier. However, due to natural distribution changes occurring in real-world problems this solution is unsatisfactory, because different distributions of train and test data may have a huge impact on the performance of state-of-the-art classifiers.

The general assumption made by classification methods is that the samples are representative, which implies that the within-class probability densities, $Pr(\boldsymbol{x}|y)$, and the a priori class distribution, $Pr(y)$, do not vary. Obviously, the second part does not hold for quantification, given that by definition it is aimed at estimating $Pr(y)$. We study this problem and the related work in quantification under prior-shift conditions, which assume that only within-class probability densities are constant.

This dissertation proposes three main contributions: (1) we present the first research study that formalizes an specialized methodology for statistical comparisons of several quantifiers over multiple test prevalences; (2) we validate two simple and cost-effective weighting strategies for nearest neighbour algorithms, offering competitive quantification performance in practice; and (3) we implement the first learning method that optimizes a quantification metric, proposing a new family of parametric loss functions that are able to balance quantification and classification measurements simultaneously.

The key contribution of our proposed methodology is that it allows us to analyze relevant properties of these comparatives from a statistical point of view, while providing meaningful insights about which algorithms are significantly better. In contrast with standard cross-validation procedures and related statistical tests, quantification assessment requires evaluating performance over a broad spectrum of test sets with different class distributions. That is why statistical comparisons of classification models are not directly applicable.

The second contribution offers a new baseline approach for solving binary quantification problems based on nearest neighbor (NN) algorithms. We present two simple and cost-effective weighting strategies, which stand out from state-of-the-art quantifiers. The conclusions drawn from Nemenyi post-hoc statistical tests show that our proposals are the only ones among studied methods that offer significant differences with respect to less robust algorithms, like CC, AC or T50; which are considered as state-of-the-art approaches by most authors.

Our last contribution is related with the fact that current quantification models are based on classifiers, presenting the weakness of being trained with a loss function aimed at classification, rather than quantification. Other recent attempts to address this issue suffer some limitations regarding reliability. Thus, we present a learning method that optimizes a multivariate metric that combines quantification and classification performance simultaneously. Our proposal offers a new framework that allows constructing binary quantifiers that are able to accurately estimate the proportion of positives, based on models with reliable classification abilities (high sensitivity).

In last chapter we discuss our main conclusions, presenting directions for future research; including optimization of root models for minimizing variance in threshold calibration and the empirical analysis of power and stability of statistical tests for quantification experiments.

"In times of change, learners inherit the future, while the learned find themselves beautifully equipped to deal with a world that no longer exists" — Eric Hoffer

Contents

Agradecimientos **iii**

Resumen **v**

Conclusiones **vii**

Abstract **xi**

1 Introduction **1**

1.1 Motivation 1

1.2 Problem description 2

1.3 Research goals and contributions 4

1.3.1 Methodology for statistical comparisons of quantifiers 4

1.3.2 Aggregative nearest neighbour quantification 5

1.3.3 Robust quantification via reliable loss minimization 5

1.4 Document outline 7

2 Quantification: notation, definitions and loss functions **9**

2.1 Notation 9

2.2 Binary quantification loss functions 10

2.2.1 Estimation bias 10

2.2.2 Absolute and squared errors 11

2.2.3 Kullback-Leibler Divergence 11

2.3 Quantification as a dataset-shift problem 12

2.3.1 Dataset-shift in machine learning 13

2.3.2 Learning assumptions under prior-shift 14

2.3.3 Invariant model characteristics under prior-shift 15

2.3.4 Other types of dataset-shift 17

2.3.5 Common causes of dataset-shift 19

3 Related work on quantification **21**

3.1 Naïve quantification: classify and count 22

3.2 Quantification via adjusted classification 23

3.2.1 Quantification via threshold selection policies 24

3.2.2 Quantification via probability estimators 26

3.3 Quantification-oriented learning 27

3.4 Alternative approaches 28

4 Methodology for statistical comparison of multiple quantifiers **31**

4.1 Adaptation of stratified k-fold cross-validation 32

4.1.1 Error estimation procedure 32

4.2 Adaptation of Friedman post-hoc tests 33

4.2.1 Discussion on statistical comparisons of classifiers 34

4.2.2 State-of-the-art tests for comparing classifiers 34

4.2.3 Standard formulation of Friedman post-hoc tests 36

4.2.4 Statistical comparisons over multiple test prevalences 38

4.3 Additional notes on evaluation methodology 39

5 Study and adaptation of NN algorithms for quantification **41**

5.1 Nearest neighbour quantification 42

5.1.1 K-nearest neighbor (KNN) 42

5.1.2 Proportion-weighted KNN (PWK$^\alpha$) 43

5.1.3 Naïve proportion-weighted KNN (PWK) 44

5.1.4 Additional notes on NN algorithms 45

5.2 Experimental setup 46

5.2.1 Datasets 46

5.2.2 Error estimation 46

5.2.3 Algorithms 47

5.2.4 Parameter tuning via grid-search 47

5.2.5 Estimation of *tpr* and *fpr* characteristics 49

5.2.6 Additional notes on learning methodology 49

5.3 Empirical analysis 49

5.3.1 Overview analysis of results 50

5.3.2 Friedman-Nemenyi statistical test 51
5.3.3 Complementary analysis of *KLD* measurements 51
5.3.4 Pair-wise comparisons with PWK$^{\alpha}$ 54
5.3.5 Analysis of results by test prevalence 56

6 Design and optimization of quantification loss functions 61
6.1 Balancing quantification and classification 62
6.2 *Q-measure*: a new parametric loss function 64
6.2.1 Classification performance 65
6.2.2 Quantification performance 66
6.2.3 Graphical analysis of *Q-measure* 67
6.3 Multivariate learning hypothesis 74
6.4 Experimental setup 77
6.4.1 Datasets 78
6.4.2 Algorithms 78
6.4.3 Estimation of *tpr* and *fpr* characteristics 79
6.5 Empirical analysis 79
6.5.1 Analysis of *AE* measurements 79
6.5.2 Analysis of *KLD* measurements 82
6.5.3 Discussion of results 83

7 Conclusions and Future Work 85
7.1 Conclusions 85
7.1.1 Methodology for statistical comparisons of quantifiers 85
7.1.2 Cost-effective nearest neighbour quantification 86
7.1.3 Quantification optimization via robust loss functions 87
7.1.4 Overall discussion of contributions 87
7.2 Future work 89
7.2.1 Optimization of root models for threshold calibration 89
7.2.2 Analysis of power and stability of statistical tests 89

Bibliography 91

List of Figures

2.1 Schematics of $X \to Y$ and $Y \to X$ domains 13
2.2 Graphical interpretation of prior-shift for binary classification . . . 16
2.3 Graphical interpretation of covariate-shift for regression 17

3.1 Classification vs quantification under prior-shift 22
3.2 Graphical interpretation of threshold selection policies 25
3.3 Graphical interpretation of SPA method 27

5.1 Statistical comparisons for NN experiments in terms of AE 52
5.2 Statistical comparisons for NN experiments in terms of KLD . . . 53
5.3 Pair-wise comparisons with respect to PWK$^\alpha$ (aggregated results) 55
5.4 Pair-wise comparisons with respect to PWK$^\alpha$ (all test prevalences) 57
5.5 Bar plots of AE results for AC, Max, KNN and PWK$^\alpha$ 60

6.1 Graphical representation of two conflicting perfect quantifiers. . . . 62
6.2 Graphical representation of classification loss functions ($p = 0.5$) . 68
6.3 Graphical representation of classification loss functions ($p = 0.09$) . 69
6.4 Graphical representation of quantification loss functions ($p = 0.5$) . 70
6.5 Graphical representation of quantification loss functions ($p = 0.09$) 71
6.6 Graphical representation with different β values for Q_β ($p = 0.5$) . 72
6.7 Graphical representation with different β values for Q_β ($p = 0.09$) 73
6.8 Statistical comparisons for Q_β experiments in terms of AE 80
6.9 Statistical comparisons for Q_β experiments in terms of KLD 81

List of Tables

2.1 Contingency table for binary problems 10

4.1 Type I and Type II errors 35
4.2 Critical values for Friedman post-hoc tests 37

5.1 PWK^{α} weights w.r.t. different training prevalences 44
5.2 Summary of datasets used for the experiments 48

6.1 Perfect quantifier with worst possible *recall* 63

Chapter 1
Introduction

This chapter reviews the main concepts that will be developed along this dissertation, trying to set out the problem and to motivate the research work behind it. We also establish our three primary research goals, including a new methodology for statistical comparisons of quantification models and two novel approaches for solving this relatively new task grounded in the field of machine learning.

1.1 Motivation

Any data scientist that had dealt with real-world problems knows that there exist classification domains that are inherently complex, being very difficult to obtain accurate predictions when focusing on each specific example; that is, to achieve high classification accuracy. However, it is not so strange to require estimations about the characteristics of the overall sample instead, mainly regarding data distribution.

For instance, in order to measure the success of a new product, there is an increasing demand of methods for tracking the overall consumer opinion, superseding classical approaches aimed at individual perceptions. To answer questions like *how many clients are satisfied with our new product?*, we need effective algorithms focused on estimating the distribution of classes from a sample. This has emerging relevance when dealing with tracking of trends over time [Rakthanmanon et al., 2012], as early detection of epidemics and endangered species, risk prevalence, market and ecosystem evolution, or any other kind of distribution change in general.

In many business, scientific and medical applications, it is sufficient, and sometimes even more relevant, to obtain estimations at an aggregated level in order to plan proper strategies. Companies could obtain greater return on investment if they were able to accurately estimate the proportion of *events* that will involve higher costs or benefits. This will avoid wasting resources in guessing the class of each specific event; a task that usually reveals itself as complex, expensive and error-prone. For example, the estimation of the proportion of policy holders that will

be involved in accidents during the next year, or the estimation of the overall consumer satisfaction with respect to any specific product, service or brand.

Tentative application scopes include opinion mining [Esuli and Sebastiani, 2010], network-behavior analysis [Tang et al., 2010], remote sensing [Guerrero-Curieses et al., 2009], quality control [Sánchez et al., 2008], word-sense disambiguation [Chan and Ng, 2006], monitoring of support-call logs [Forman et al., 2006], credit scoring [Hand, 2006], species distribution modeling [Dudık et al., 2007; Phillips et al., 2004], seminal quality control [González-Castro, 2011], or adaptive fraud-detection [Provost and Fawcett, 2001], among others.

In our case, the work described in [González et al., 2013] is the seminal work that gives origin to this thesis, in which we present a cost-sensitive learning approach to solve a real world problem: the biomass estimation of several plankton taxonomic groups. The method consist, basically, in attach to each instance its corresponding misclassification cost and then minimize a cost-sensitive loss function. Related con this dissertation, the most important achievement is the appearance of a real world dataset where prevalence of the different classes change with the time: the presence of any type of plankton can be increased or decreased by many factors. However, we later concluded that the problem under study was not directly addressable through quantification based on prior-shift restrictions, given that this dataset does not fulfill the learning assumptions formalized in Chapter 2.

1.2 Problem description

In machine learning, the task of quantification is *to accurately estimate the number of positive cases (or class distribution) in a test set, using a training set that may have a substantially different distribution* [Forman, 2008]. Despite having many potential applications, this problem has been barely tackled within the community, and needs yet to be properly standardized in terms of error measurement, experimental setup and methodology in general. Unfortunately, quantification has attracted little attention due to the mistaken belief of being somehow trivial. The key problem is that it is not as simple as classifying and counting the examples of each class, because different distributions of train and test data may have a huge impact on the performance of state-of-the-art classifiers. The general assumption made by classification methods is that the samples are representative [Duda et al., 2001], which implies that the within-class probability densities, $Pr(\boldsymbol{x}|y)$, and the a priori class distribution, $Pr(y)$, do not vary. Obviously, the second part does not hold for quantification, given that by definition it is aimed at estimating $Pr(y)$.

The influence of different changing environments on classification and knowledge-based systems performance have been analyzed in several studies (see, for instance, [Kelly et al., 1999; Groot et al., 2005; Hand, 2006; Alaiz-Rodríguez and Japkowicz,

2008; Cieslak and Chawla, 2009]). This suggests that addressing distribution drifts is a complex and critical problem. Moreover, many published works are focused on addressing distribution changes for classification, offering different views of what is subject to change and what is supposed to be constant. As in recent quantification-related works, we are only focused on studying changes on the a priori class distribution, but maintaining within-class probability densities constant. That is, we are focused on *prior-shift* problems. This kind of domains are identified as $Y \rightarrow X$ problems by Fawcett and Flach [2005]. Provided that we use stratified sampling [Fleiss et al., 2003], an example of situations where $Pr(\boldsymbol{x}|y)$ does not change is when the number of examples of one or both classes is conditioned by the costs associated with obtaining and labeling them [Weiss and Provost, 2003]. The explicit study of other types of distribution shifts, as well as $X \rightarrow Y$ domains, are out of the scope of this dissertation [for further reading refer to Webb and Ting, 2005; Holte, 2006; Quionero-Candela et al., 2009; Moreno-Torres et al., 2012a].

ROC-based methods [Provost and Fawcett, 2001; Fawcett, 2004] and cost curves [Drummond et al., 2006] have been successfully applied to adjust the classification threshold, given that new class priors are known in advance. However, as already stated by Forman [2008], these approaches are not useful for estimating class distributions from test sets. Similarly, if these new priors are unknown, two main approaches have been followed in the literature. On the one hand, most of the published works are focused on adapting deployed models to the new conditions [Latinne et al., 2001; Vucetic and Obradovic, 2001; Saerens et al., 2002; Xue and Weiss, 2009; Alaiz-Rodríguez et al., 2011]. On the other hand, the alternative view is mainly concerned with enhancing *robustness* in order to learn models that are more resilient to changes in class distribution [Alaiz-Rodríguez et al., 2007]. Anyhow, the aim of these methods, although related, is quite different from that of quantification because adapting a classifier for improving individual classification performance does not imply obtaining better quantification predictions, as we shall discuss in Chapter 3. Moreover, there exists a natural connection with imbalance-tolerant methods [Weiss, 2004; Vucetic and Obradovic, 2001], mainly those based on preprocessing of data [Ramentol et al., 2012; López et al., 2011]. Actually, quantification was originally designed to deal with highly imbalanced datasets; however, those preprocessing techniques are not directly applicable in changing environments [Forman, 2008].

The main approach that has been studied in the literature for learning an explicit binary-quantification model is based on standard classifiers, following a two-step training procedure. The first step is to train a classifier optimizing a classification metric, usually accuracy. Afterwards, the next step is to study some relevant properties of this classifier. The aim of this second step is to correct the quantification prediction obtained from aggregating classifier estimates [Forman, 2008; Bella et al., 2010].

1.3 Research goals and contributions

This thesis pursues three main objectives. In order to validate the results from our experiments, we establish a new experiment methodology for the task of quantification, based on the widespread cross-validation procedure and the two step Friedman-Nemenyi statistical test. The development of this experimental procedure is conducted in parallel with the design and validation of two complementary novel approaches for addressing quantification.

For our first milestone we evaluate how simple weighting strategies can improve the performance of NN-based classification algorithms over quantification problems. Then, we analyze the viability of a pure quantification-based learning approach (in contrast with a classification one), along with the designing of a new metric from scratch.

1.3.1 Methodology for statistical comparisons of quantifiers

Since the beginning of our research, we observed the lack of a standardized experimental design for comparing quantification algorithms, mainly in terms of statistical procedures. That is why we started by studying the most recent approaches, trying to redefine a new methodology that could cover the specific requirements of statistical comparisons of several quantifiers over multiple test prevalences, while preserving the core concepts already validated for standard machine learning methodologies.

The key contribution of our proposal is that this new methodology for quantification allows us to analyze relevant properties of these comparatives from a statistical point of view. Furthermore, it also provides meaningful insights about which algorithms are significantly better, with a certain confidence degree, thanks to the adaptation of two Friedman post-hoc statistical tests [Demšar, 2006], and the redesign of test set generation in stratified k-fold cross-validation [Refaeilzadeh et al., 2009].

Analyzing the inherent requirements of quantification, it is straightforward to conclude that it demands evaluating performance over whole sets, rather than by means of individual classification outputs. Moreover, quantification assessment also requires evaluating performance over a broad spectrum of test distributions in order for it to be representative. Thus, traditional machine learning techniques for comparing classification models [Garcia and Herrera, 2008; Demšar, 2006] are not directly applicable and need to be adapted.

The main difference with previous experimental setups followed for quantification is that our methodology is not focused on a particular domain, nor a specific range of train or test prevalences. We aim to cover a broader or more general scope.

Our proposed methodology is described in more detail in Chapter 4, and applied in the experiments discussed in Chapter 5 and Chapter 6, which cover the two remaining research objectives that we introduce in next sections. This experimental setup has been also successfully published on *Pattern Recognition* [Barranquero et al., 2013].

1.3.2 Aggregative nearest neighbour quantification

The first experimental study of this dissertation is aimed at analyzing the behavior of nearest neighbor (NN) algorithms for prevalence estimation in binary problems. It is well-known that each learning paradigm presents a specific learning bias, which is best suited for some particular domains. As it happens in other machine learning tasks, we expect that NN approaches should outperform other methods in some quantification domains.

The motivational intuition beyond this work is that the inherent behavior of NN algorithms should yield appropriate quantification results based on the assumption that they may be able to *remember* details of the topology of the data, independently of the presence of distribution changes between training and test. Moreover, bearing in mind that once the distance matrix has been constructed we are able to compute many different estimations in a straightforward way, we shall explain why we consider that these methods offer a cost-effective alternative for this problem. At the very least, they reveal themselves to be competitive baseline approaches, providing performance results that challenge more complex methods proposed in previous works.

In summary, we seek for a quantification approach with competitive performance that could offer simplicity and robustness. Earlier proposals are mostly based on SVM classifiers [Forman, 2008; Esuli and Sebastiani, 2010], which are one of the most effective state-of-the-art learners. These previous quantification methods showed promising empirical results due to theoretical developments aimed at correcting the aggregation of individual classifier outputs. Thus, our main hypothesis is whether we could apply the aforementioned theoretical foundations with simpler classifiers, such as NN-based algorithms, in order to stress the relevance of corrections of this kind over the use of any specific family of classifiers as base learners for quantification.

The results of this research have been recently published on *Pattern Recognition* [Barranquero et al., 2013], and are described in more detail in Chapter 5.

1.3.3 Robust quantification via reliable loss minimization

In parallel with our NN study, we realize that there exist an open issue regarding the fact that current learning processes are not taking into account the target

performance measure. That is, the algorithms are designed to optimize the results from a classification loss function, while they are then evaluated and compared with another loss function, which is usually aimed at quantification performance.

Therefore, our third objective is to evaluate whether it may be more effective to learn a classifier optimizing a quantification metric, in contrast with a classification-based one. Conceptually, this alternative strategy is more formal, because the learning process is taking into account the target performance measure. This issue will be explored in more detail in Chapter 6.

The idea of optimizing a pure quantification metric during learning was introduced by Esuli and Sebastiani [2010], although they neither implement nor evaluate it. Their proposal is based on learning a binary classifier with optimum quantification performance. We argue that this method has a pitfall. The key problem that arises when optimizing a pure quantification measure is that the resulting hypothesis space contains several global optimums (see Figures 6.4 and 6.5). However, in practice, those optimum hypotheses are not equally good because they differ in terms of the quality of their future quantification predictions.

This dissertation claims that the robustness of a quantifier based on an underlying classifier is directly related to the reliability of such classifier. For instance, given several models showing equivalent quantification performance during training, the learning method should prefer the best one in terms of its potential of generalization. As we shall explain in Chapter 6, this factor is closely related with their classification abilities. We believe that these ideas, primarily those regarding trustability, open an interesting research area within quantification.

This lead us to further explore the approach of Esuli and Sebastiani, trying to build a learning method able to induce more robust quantifiers based on classifiers that are as reliable as possible. In order to accomplish this goal, we introduce a new metric that combines both factors. That is, a metric that mix in classification performance to the training objective, resulting in better quantification models.

As it happens with any other quantification metric, our proposal measures the performance from an aggregated perspective, taking into account the results over the whole sample. The difficulty for optimizing such functions is that they are not decomposable as a linear combination of the individual errors. Hence, not all binary learners are capable of optimizing them directly, requiring a more advanced learning machine. Therefore, we adapt Joachim's multivariate SVMs [Joachims, 2005] to implement our proposal and the idea presented by Esuli and Sebastiani. In order to validate them, another key contribution is to perform an exhaustive study in which we compare both approaches, as well as several state-of-the-art quantifiers by means of benchmark datasets.

This study is currently under review on *Pattern Recognition* [Barranquero et al., under review]. The final results are discussed in Chapter 6.

1.4 Document outline

Chapter 2 introduces the core concepts related with quantification, along with a brief review of different learning assumptions under dataset-shift. Chapter 3 examines current state-of-the-art approaches for addressing quantification. Chapter 4 establishes the foundations of our proposed methodology in order to present the experimental results of NN algorithms in Chapter 5, and of multivariate optimization in Chapter 6. Finally, in Chapter 7 we discuss our main conclusions and directions for future research.

Chapter 2

Quantification: notation, definitions and loss functions

Before going into the details of other related works and our own proposals, this chapter presents a formal view of quantification, defined in terms of a traditional machine learning task. We start presenting a standard definition for the problem and its notation. Then we introduce the state-of-the-art loss functions that are already used on previous works. And finally we review some alternative views of the task, contextualizing it within the general dataset-shift framework.

2.1 Notation

From a statistical point of view, a binary quantification task is aimed at estimating the prevalence of an event or property within a sample. During the learning stage, we have a training set with examples labeled as positives or negatives; formally, $D = \{(\boldsymbol{x}_i, y_i) : i = 1 \dots S\}$, in which $\boldsymbol{x}_i$ is an object of the input space $\mathcal{X}$ and $y_i \in \mathcal{Y} = \{-1, +1\}$. This dataset shows a specific distribution that can be summarized with the actual proportion of positives or prevalence. The learning objective is to obtain a model being able to predict the prevalence of another sample (p), usually identified as the test set, that may show a remarkably different distribution of classes. Thus, the input data is equivalent to that of traditional classification problems, but the focus is stressed over the estimated prevalence of the sample (p'), rather than the labels assigned to each test individual. Notice that we use p and p' to identify actual and estimated prevalences of any sample, these variables are not tied to training or test sets in any way.

Table 2.1 summarizes the notation that we shall employ throughout this dissertation. First an algorithm is applied over the training set in order to learn a classifier. Then, we apply it over the test test, where P represents the count of actual positives and N the count of actual negatives. Once the classifier is applied over this second sample to predict its classes, we have that P' is the count of individuals predicted as positives, N' the count of predicted negatives, while

TP, FN, TN and FP represent the count of *true positives*, *false negatives*, *true negatives* and *false positives*.

Table 2.1: Contingency table for binary problems

	P	N
P'	TP	FP
N'	FN	TN
$(S = P + N = P' + N')$		

Once computed all the values presented in Table 2.1, we can then obtain actual prevalence

$$p = \frac{P}{S} = \frac{TP + FN}{S}, \tag{2.1}$$

and estimated prevalence

$$p' = \frac{P'}{S} = \frac{TP + FP}{S}. \tag{2.2}$$

Notice that they only differ with respect to one term, being FN and FP respectively. This suggests that both FN and FP values may play an important role during performance evaluation.

2.2 Binary quantification loss functions

This section presents a brief review of several quantification loss functions that have been proposed on relevant papers related with binary quantification. In Section 6.2.2 we discuss them in more detail, along with our own proposals, and including meaningful graphical interpretations for balanced (Figure 6.4) and unbalanced (Figure 6.5) scenarios.

2.2.1 Estimation bias

According to Forman [2008], the estimation bias is a natural error metric for quantification, which is computed as the estimated percent positives minus the actual percent positives

$$bias = p' - p = \frac{P' - P}{S} = \frac{FP - FN}{S}. \tag{2.3}$$

When a method outputs more FP than FN then it shows a positive bias, and vice-versa. Therefore, this metric measures whether the model tends to overestimate

or underestimate the proportion of the positive class. However, this metric is not useful to evaluate the overall performance in terms of average error (for a set of samples), because negative and positive biases are neutralized. That is, as Forman points out, *a method that guesses 5% too high or too low equally often will have zero bias* on average.

2.2.2 Absolute and squared errors

Forman proposed the *Absolute Error* (AE) between actual and predicted positive prevalence as standard loss function for quantification [Forman, 2005, 2006, 2008], which is simple, interpretable and directly applicable:

$$AE = |p' - p| = \frac{|P' - P|}{S} = \frac{|FP - FN|}{S}. \tag{2.4}$$

As an alternative to AE, the *Squared Error* (SE) is proposed by Bella et al. [2010]

$$SE = (p' - p)^2 = \left(\frac{P' - P}{S}\right)^2 = \left(\frac{FP - FN}{S}\right)^2. \tag{2.5}$$

Actually, *Mean Absolute Error* (MAE) and *Mean Squared Error* (MSE) are probably the most commonly used loss functions for regression problems. The concept of computing the absolute or squared error of real value estimations can be extended to any problem based on a continuos variable, like p. However, in the case of quantification we have a single prediction per sample, and averaging among samples with different actual prevalence or from different domains has some implications that should be carefully taken into account [Forman, 2008]. Note, for instance, that having a 5% AE for a test set with 45% of positive examples may not be equivalent to obtaining the same error over a test set with only 10%. Nevertheless, as Esuli and Sebastiani [2010] suggest, a function must deteriorate with $|FP - FN|$ in order to be considered an appropriate quantification metric, which is fulfilled by AE and SE. In fact, both loss functions are definitely suitable for evaluating quantification performance, provided that they are only averaged over specific testing prevalences, as in standard cross-validations.

2.2.3 Kullback-Leibler Divergence

Kullback-Leibler Divergence (KLD), also known as normalized cross-entropy [Forman, 2008; Esuli and Sebastiani, 2010, see] can be applied in the context of quantification. Assuming that we have only two classes, the final equation is:

$$KLD = \frac{P}{S} \cdot \log\left(\frac{P}{P'}\right) + \frac{N}{S} \cdot \log\left(\frac{N}{N'}\right). \tag{2.6}$$

This metric determines the error made in estimating the predicted distribution $(P'/S, N'/S)$, with respect to the true distribution $(P/S, N/S)$.

The main advantages of KLD are that it may be more appropriate to average over different test prevalences and more suitable for extending the quantification task for multiclass problems. However, a drawback of KLD is that it is less interpretable than other measures, like AE.

Furthermore, we also need to define its output for those cases where P, N, P' or N' are zero. This implies that KLD is not properly bounded, obtaining undesirable results, like infinity or indeterminate values, when the actual or estimated proportions are near 0% or 100%, needing further corrections to be applicable [Forman, 2008]. According to Forman's recommendations, we fix these issues as follows:

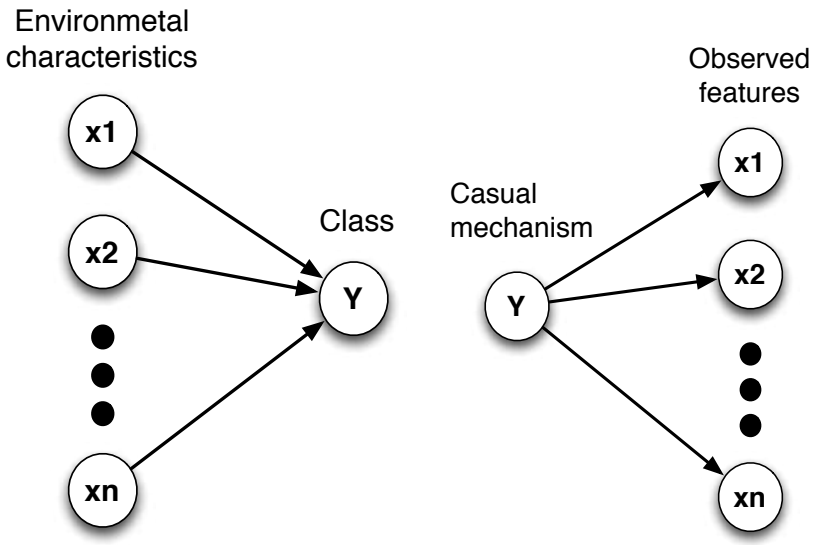
Algorithm 1 Correction of KLD under extreme conditions

if $(p' == 0 \;||\; p' == 1)$ **then**
  $p' \Leftarrow \mid p' - (0.5/S) \mid$
end if

2.3 Quantification as a dataset-shift problem

Although the term quantification has been recently formalized within the umbrella of machine learning [Forman, 2008], obviously it is not a new problem in the more global sense. In fact, after a deep review of possible complementary works, we found that there exists alternatives views on similar problems with different names and aims. It is common to find abstractions like data fracture [Moreno-Torres et al., 2010], learning transfer [Storkey, 2009], distributional divergence [Cieslak and Chawla, 2009], distribution matching [Gretton et al., 2009], changing environments [Alaiz-Rodríguez and Japkowicz, 2008], contrast mining [Yang et al., 2008], or changes of classification [Wang et al., 2003], among others. These related lines of research show different outlooks of what is subject to change and what is supposed to be constant. In this regard, the book by Quionero-Candela et al. [2009], and the paper by Moreno-Torres et al. [2012a] offer a unifying view on the field, contextualizing these problems under the general term *dataset-shift*.

The novelty about quantification is on the perspective adopted, because the problem is faced from a completely different approach. Most of similar problems are mainly focused on adapting learned models for improving classification accuracy on a test set that shows a different distribution with respect to the train set. However, with quantification, the aim is no longer on adapting the individual outputs, but on estimating the test class distribution directly. Although in many



(a) $X \rightarrow Y$ (b) $Y \rightarrow X$

Figure 2.1: Schematics of $X \rightarrow Y$ and $Y \rightarrow X$ domains [Fawcett and Flach, 2005].

cases this global outcome may also require to be calibrated, as we shall discuss in Chapter 3.

2.3.1 Dataset-shift in machine learning

Dataset-shift is defined as *a change between training and test joint distribution* or $Pr(\boldsymbol{x}, y)$ [Moreno-Torres et al., 2012a]. In this context, we can distinguish two main types of problems, already introduced as $X \rightarrow Y$ (predictive) and $Y \rightarrow X$ (causal); where the joint probability distribution can be rewritten as $Pr(y|\boldsymbol{x})Pr(x)$ and $Pr(\boldsymbol{x}|y)Pr(y)$ respectively.

The paper by Webb and Ting [2005] and the response to that paper by Fawcett and Flach [2005] set out a very interesting discussion of these two domains in the context of ROC analysis (see Figure 2.1, taken from [Fawcett and Flach, 2005]). On the one hand, Webb and Ting put ROC analysis under suspect for $X \rightarrow Y$ problems, given that *tpr* and *fpr* characteristics varies between training and test (see Section 2.3.3). On the other hand, Fawcett and Flach counter-argue in favor of ROC techniques, with examples of $Y \rightarrow X$ problems.

Interestingly enough, both papers propose a manufacturing fault detection problem as one of their application examples, but from two alternative views; which could help in highlighting the core differences between both domains. An

example of an $X \rightarrow Y$ scenario is when the fault, y, is conditioned by production line configuration, $\boldsymbol{x}$. In this case, it is unrealistic that a faulty product could cause the production line to be in a specific configuration, while a change in the frequency of faults will result from a change in the frequency of specific configurations (see Figure 2.1a). However, we could state this problem as a $Y \rightarrow X$ scenario if we consider that the feature vector $\boldsymbol{x}$ is built from the outcomes of different sensors for detecting defective properties of the products. As long as these sensors do not alter the nature of the products, it is also unrealistic that a specific lecture from these sensors could cause the product to be faulty, while a change in the frequency of faults will result in a change in the frequency of active sensors (see Figure 2.1b).

There are three main types of dataset-shift are prior-shift, covariate-shift and concept-shift. On next subsection we go into further details of the learning assumptions under prior-shift as a $Y \rightarrow X$ problem, which is the core assumption followed in the experiments presented in this dissertation. Then we briefly review the general assumptions under covariate-shift and concept-shift in order to present prior-shift in a more general context.

2.3.2 Learning assumptions under prior-shift

The learning assumptions established by Forman on his seminal works [Forman, 2008, 2006; Forman et al., 2006; Forman, 2005, 2002] are focused on prior-shift problems [Quionero-Candela et al., 2009]. Nevertheless, the concept of prior-shift is broader than that of quantification, existing many complementary works that are focused on different goals, where it is also referred as prior probability shift [Moreno-Torres et al., 2012a], class prior change [du Plessis and Sugiyama, 2012], shifting priors [Cieslak and Chawla, 2009], class distribution change [Alaiz-Rodríguez and Japkowicz, 2008], or even varying class distributions [Webb and Ting, 2005].

Recall that the general postulation made by traditional machine learning methods is that the samples are representative [Duda et al., 2001]. This premise implies that the within-class probability densities, $Pr(\boldsymbol{x}|y)$, and the a priori class distribution, $Pr(y)$, must be constant between training and test. Not surprisingly, the second assertion is discarded for quantification, given that this task is aimed at estimating test class distribution or $Pr(y)$.

Therefore, prior-shift is defined under the assumption that the a priori class distribution, $Pr(y)$, changes, but the within-class probability densities, $Pr(\boldsymbol{x}|y)$ still remains constant between training and test. These conditions are fulfilled, for instance, when the changes in class priors are obtained by means of stratified sampling [Webb and Ting, 2005] or when the number of examples of one or both classes is conditioned by the costs associated with obtaining and labeling them [Weiss and Provost, 2003]. This kind of domains are identified as $Y \rightarrow X$ problems by Fawcett and Flach [2005], as already discussed in Section 2.3.1.

In general, prior-shift is also assumed to imply a change in joint probability, $Pr(\boldsymbol{x}, y)$, a posteriori probability, $Pr(y|\boldsymbol{x})$, and covariate distribution, $Pr(\boldsymbol{x})$ [González-Castro et al., 2013; Cieslak and Chawla, 2009]. Therefore, only $Pr(\boldsymbol{x}|y)$ should be considered constant.

Figure 2.2 shows a graphical interpretation for a binary problem, with positives depicted as red crosses ($y = +1$), and negatives as blue circles ($y = -1$). On top we have a training set with a specific class distribution (Figure 2.2a), while on the bottom-left we can observe that the test set shows a higher proportion for positive class (Figure 2.2b). Given that the set of examples from the negative class is not modified, then it is self-proven that $Pr(\boldsymbol{x}|y = -1)$ is constant. Additionally, for the positive class this equally holds, $Pr(\boldsymbol{x}|y = +1)$ is also constant, because its instances follows the same distribution between both figures. That is, the shape and position of the cloud drawn by positive instances is invariant, only changes its density. For the bottom-right hand side (Figure 2.2c), we have that both $Pr(\boldsymbol{x}|y)$ and $P(y)$ change, showing a completely different problem that is not covered by prior-shift framework.

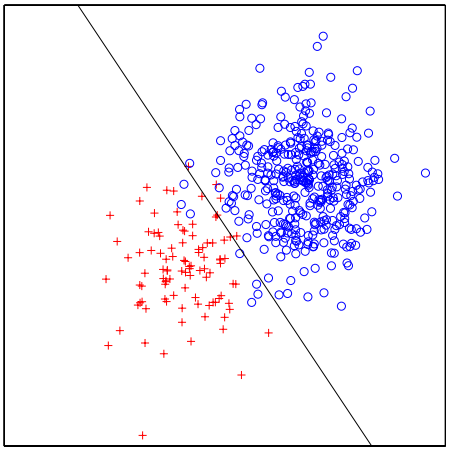
It is also worth mentioning that Storkey [2009] alternatively formalizes the problem as a causal model $Pr(\boldsymbol{x}|y)Pr(y)$, which can be used for inferentially obtain $Pr(y|\boldsymbol{x})$. In this probabilistic classification context, he sets out two main possible situations: knowing $Pr(y)$ for test set or not. For the first case, we could have a potential application for quantification, given that it may be used as a tool for estimating test $Pr(y)$. For the latter case he proposes a more complicated solution in which the main idea is to estimate test $Pr(y|\boldsymbol{x})$ given that $Pr(\boldsymbol{x}|y)$ is known, and therefore certain distributions over class variable are more likely.

2.3.3 Invariant model characteristics under prior-shift

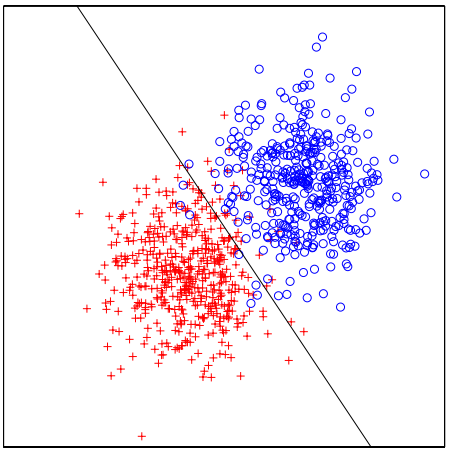
The key issue about preserving within-class probabilities is that this guarantees that some of the intrinsic characteristics of the model are independent of prior-shift changes in class distribution. The two more important in the context of quantification are *tpr* (true positive rate) and *fpr* (false positive rate) characteristics [Forman, 2008; Webb and Ting, 2005], defined as

$$tpr = \frac{TP}{P} \qquad \text{and} \qquad fpr = \frac{FP}{N}. \tag{2.7}$$

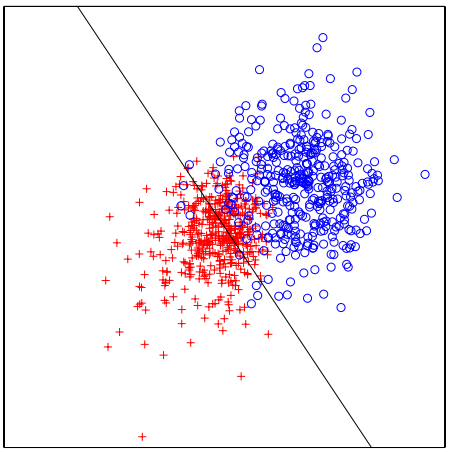
These two rates are crucial for understanding state-of-the-art algorithms, as we discuss in Chapter 3. Observing Figure 2.2, it easy to see that for Figure 2.2b these rates are constant: the proportion of instances that falls on the wrong side of the hyperplane is equivalent with respect to Figure 2.2a. This does not hold for Figure 2.2c, where the count of TP weighted in terms of P is not comparable.



(a) Binary problem

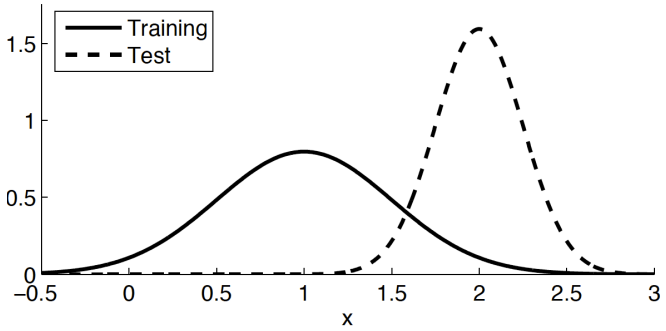


(b) $Pr(\boldsymbol{x}|y)$ constant

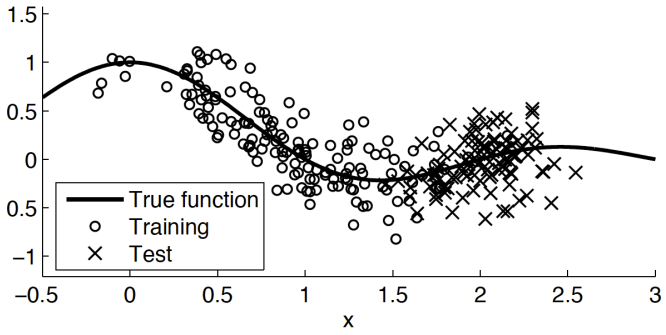


(c) $Pr(\boldsymbol{x}|y)$ changes

Figure 2.2: Graphical interpretation of prior-shift for binary classification. The figure on top (a) shows a dataset with a specific class distribution. The figure on bottom-left (b) shows a prior-shift change over this dataset with a higher proportion for positive class (i.e, $Pr(y)$ changes), while maintaining within-class probabilities constant. The figure on bottom-right (c) shows another change in class distribution, which is not considered a prior-shift because it does not preserve within-class probabilities.



(a) Input data densities



(b) Objective function

Figure 2.3: Graphical interpretation of covariate-shift for regression [Yamazaki et al., 2007]. Top figure (a) represents the data distribution for input variable x, in both training and test datasets. Bottom figure (b) represents the objective function and the position of (x, y) points, observing a possible problem of overfitting due to covariate-shift.

2.3.4 Other types of dataset-shift

Although the explicit study of other types of dataset-shift are out of the scope of this dissertation, we present here a brief review of the most important in order to contextualize prior-shift on a more general framework.

Covariate-shift (population-drift)

Covariate-shift problem refers to changes in the input variables, $\boldsymbol{x}$, also known as covariates or features. It is probably one of the most studied problems related with dataset-shift, which was defined several years ago by Shimodaira [2000]. Although,

there exist equivalent terms, like population drift [Alaiz-Rodríguez and Japkowicz, 2008; Hand, 2006; Kelly et al., 1999], they are less widely used.

The formal definition proposed by Moreno-Torres et al. [2012a] is that covariate-shift is *a change in* $Pr(\boldsymbol{x})$ *where* $Pr(y|\boldsymbol{x})$ *remains constant.* They also argue that this problem only happens on $X \rightarrow Y$ domains. This definition is based on [Storkey, 2009], where covariate-shift is defined as a change that occurs *when the data is generated according to a model* $Pr(y|\boldsymbol{x})Pr(x)$ *and where the distribution* $Pr(x)$ *changes between training and test scenarios.*

Figure 2.3, taken from [Yamazaki et al., 2007], shows an example of covariate-shift for a non-linear regression task, which highlights the overfitting problem usually associated with this type of dataset-shift. The top graph represents two data distributions for variable x, one for training and another for test, showing a strong shift in the distribution of this covariate (moved to the right). The lower graph represents both x and y values, as well as the actual objective function, given by $Pr(y|x)$, which is common for both training and test. Any learner that only takes into account the training distribution will result in a model with very poor generalization abilities with respect to unseen test data.

There are many different approaches for tackling covariate-shift. One of the most widely studied is based on re-weighting the training data such that its distribution more closely matches that of the test data, commonly known as distribution matching [Gretton et al., 2009; Sugiyama et al., 2009, 2007b; Yamazaki et al., 2007]. Similar works are focused on unbiased error estimation under covariate-shift, like importance weighted cross validation [Sugiyama et al., 2007a].

Other appealing alternatives include discriminative learning [Bickel and Scheffer, 2009], where neither training nor test distribution are modeled explicitly, rewriting the problem as an integrated optimization problem solved through kernel logistic regression. Finally, Globerson et al. [2009] also propose an adversarial view through a minimax approach, in which the problem is addressed as a two-player game where the model is trained against a feature removal algorithm. For further reading, a recent book by Sugiyama and Kawanabe [2012] offer a deeper review on covariate-shift.

Concept-shift (concept-drift)

The problem of concept-shift is formally defined by Moreno-Torres et al. [2012a] as *a change in the relationship between input and class variables*, with two main forms depending on the domain. For $X \rightarrow Y$, it is defined as a change in $Pr(y|\boldsymbol{x})$ (usually known as functional relational change [Yamazaki et al., 2007]), where $Pr(\boldsymbol{x})$ remains constant. Similarly for $Y \rightarrow X$, it is defined a change in $Pr(\boldsymbol{x}|y)$ (or class definition change [Alaiz-Rodríguez and Japkowicz, 2008; Hand, 2006]), where $Pr(y)$ remains constant. Although it is more commonly referred as concept-

drift, Moreno et al. proposed the term *shift* for unifying the names of all types of dataset-shift.

It is identified as one of the hardest of the three types of changes that we have reviewed until now, and possibly the oldest and most widespread. In fact, there exists many works that have address this problem with very different approaches; including incremental learning in imbalance scenarios [Hoens et al., 2012] and in dynamic contexts [Syed et al., 1999; Schlimmer and Granger, 1986], adaptation of classes and subclasses [Alaiz-Rodríguez et al., 2011], conceptual equivalence [Yang et al., 2008], minimizing disagreements [Helmbold and Long, 1994], etc.

Other related works are in turn focused on tracking this type of changes [Klinkenberg and Joachims, 2000; Widmer and Kubat, 1996, 1993], with more recent ones using techniques like boosting [Grbovic and Vucetic, 2011]. For further reading refer to the technical reports by Žliobaitė [2010] and Tsymbal [2004].

2.3.5 Common causes of dataset-shift

The two main causes of dataset-shift are sample selection bias [Zadrozny, 2004] and non-stationary environments [Sugiyama and Kawanabe, 2012]. There are some confusion around these terms, being usually confused as dataset-shifts on their own [Cieslak and Chawla, 2009]. However they are better categorized as causes for some of them, as proposed by [Moreno-Torres et al., 2012a].

Sample selection bias is the most common cause of dataset-shift [Phillips et al., 2009; Huang et al., 2007; Dudık et al., 2005], with three variants: *missing completely at random* (MCAR), *missing at random* (MAR), *missing at random-class* (MARC), and *missing not at random* (MNAR). All of them are based on the assumption that the training examples have been obtained through a biased method, and thus do not represent reliably the operating environment where the classifier is to be deployed. That is, the training joint probability is defined as $Pr(s = 1|x, y)$, where s represents the event of choosing an individual for training or not (the bias).

The main difference between the four alternatives is based on the independence properties of this bias. For MCAR, s is not related with $\boldsymbol{x}$ nor y: $Pr(s = 1|x, y) = Pr(s = 1)$, resulting in no dataset-shift at all. For MAR, s is independent of y given $\boldsymbol{x}$: $Pr(s = 1|x, y) = Pr(s = 1|\boldsymbol{x})$, which is a common cause of covariate-shift. For MARC, s is independent of $\boldsymbol{x}$ given y: $Pr(s = 1|x, y) = Pr(s = 1|y)$, which usually involves a prior-shift (also known as stratified sampling [Fleiss et al., 2003]). And finally, for MNAR, the is no independence assumption, which can potentially produce any type of dataset-shift.

In order to be able to address sample selection bias Quionero-Candela et al. [2009] give a set of sufficient and necessary conditions. The *support condition* simply states that any covariate vector $\boldsymbol{x}$ that can be found in training set must also

can be drawn from test distribution. The *selection condition* goes a step further, applying the same restriction to any pair $(\boldsymbol{x}, y)$. In other words, the training joint distribution must be contained within test joint distribution.

Non-stationary environments are the second most common cause of dataset-shift, occurring when the difference between training and test data is a consequence of a temporal or spatial change. Usually, this happens in adversarial scenarios like spam filtering, network intrusion detection, and security in general [Barreno et al., 2010; Biggio et al., 2010; Laskov and Lippmann, 2010]. Other tentative application scopes are also remote sensing and seasonal-based changes [Alaiz-Rodríguez et al., 2009].

This type of problem is receiving an increasing amount of attention in the machine learning field; and usually copes with non-stationary environments due to the existence of an adversary that tries to work around the existing classifier's learned concepts. In terms of the machine learning task, this adversary warps the test set so that it becomes different from the training set, thus introducing any possible kind of dataset-shift [Moreno-Torres et al., 2012a].

Chapter 3

Related work on quantification

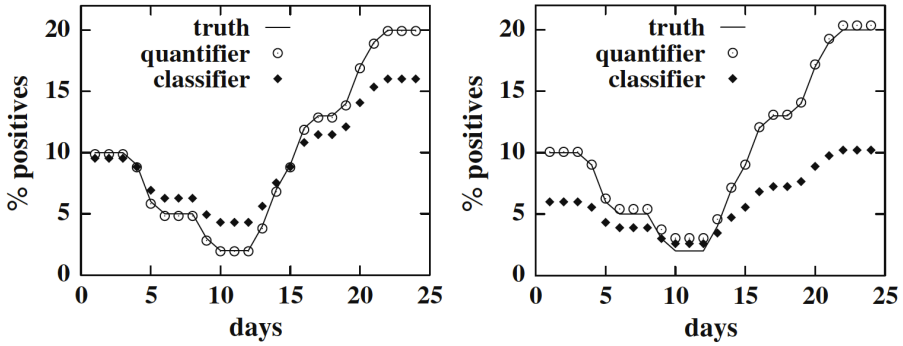
Quantification problem has been addressed formally in a limited number of papers during the last years, where several complementary approaches have been proposed. This chapter presents a review of the most relevant quantification methods, focusing on two-class problems. For completeness, on last section we also describe other related quantification methods, which are less related with the core of this dissertation, like quantification for cost-based learning and regression.

The fact is that for many classification problems it is very costly or rather impossible to induce an accurate classifier; given that real-world concepts can be very difficult to learn, even for state-of-the-art classification algorithms. Moreover, the dataset-shifts that we have reviewed on previous chapter are fundamental to many application domains, specifically for tracking of trends. However, it is also common that machine learning methods assume that training set is a random sample drawn from test distribution, as we have already discussed in previous chapter.

For some application domains it may be enough to detect the change or simply knowing the direction of the trend. However, many business problems require more precise estimates of the proportion of classes. In this regard, it is sufficient, but not necessary, to induce a *perfect* classifier. Actually, if the number of FN compensates the count of FP errors, the estimated prevalence is correct. Therefore, most of the state-of-the-art quantifiers no longer focus on estimating accurate individual outputs: shifting the nature of uncertainty from individuals to aggregate count of cases [Bella et al., 2010; Forman, 2008, 2006, 2005].

Furthermore, these new approaches offers other benefits given that they also perform well with limited number of samples and under high class imbalance, as depicted in Figure 3.1.

Forman [2008] states several reasons why quantification may have been unrecognized as a formal machine learning task. First of all, it might seem trivial, although empirical experiments give reasons to think otherwise. Moreover, it does not fit well with traditional machine learning techniques for empirical



(a) Training with $P = 100$ and $N = 900$

(b) Training with $P = 10$ and $N = 90$

Figure 3.1: Classification vs quantification under prior-shift [Forman, 2008]. These figures shows the comparative behavior of a classifier vs a quantifier for a fictitious trend, training both models on day 0. On left figure (a) the training dataset size is 1000, showing that classifier systematically overestimates test class distribution when the proportion of positives decreases, and vice-versa. On right figure (b) the training size is reduced to 100, observing a more robust performance for the quantifier with respect to sample size.

error estimation. Examples are standard cross-validation and statistical tests for comparing models, which usually assume that both training and test distributions are equivalent. In fact, it requires a different, and more complex, methodology for experimentation (see Chapter 4).

3.1 Naïve quantification: classify and count

It is worth noting that quantification is traditionally linked with classification algorithms. The most simple method for building a quantifier is to learn a classifier, use the resulting model to label the instances of the sample and count the proportions of each class. This method is taken as baseline by Forman [2008], identifying it as *Classify & Count* (CC). Actually, it is straightforward to conclude that a perfect classifier would lead to a perfect quantifier. The key problem is that developing a perfect classifier is unrealistic, getting instead imperfect classifiers in real-world environments. This also implies that the quantifier will inherit the bias of the underlying classifier.

For instance, given a binary classification problem where the learned classifier tends to misclassify some positive examples, then the derived quantifier will underestimate the proportion of the positive class. This effect becomes even more

problematic in a changing environment, in which the test distribution usually is substantially different from that of the training set. Following the previous example, when the proportion of the positive class goes up uniformly in the test set, then the number of misclassified positive instances increases and the quantifier will underestimate the positive class even more (see Figure 3.1). Forman pointed out and studied this behavior for binary quantification, proposing several methods to undertake the classification bias.

3.2 Quantification via adjusted classification

Aimed at correcting such bias, Forman [2005] proposed a new method termed *Adjusted Count* (AC), where the process is to train a classifier and estimate its *tpr* and *fpr* characteristics, defined in Equation (2.7), through cross-validation over the training set. That is, for each fold we compute TP, FP, P and N to average *tpr* and *fpr* among all folds. Then, the next step is to count the positive predictions of the classifier over the test examples (i.e., just like the CC method) and adjust this value with the following formula

$$p'' = \frac{p' - fpr}{tpr - fpr}, \tag{3.1}$$

where p'' denotes the adjusted proportion of positive test examples and p' is the estimated proportion via counting the classifier outputs over the test set. In some cases, this leads to infeasible estimates of p, requiring a final step in order to clip the estimation into the range $[0, 1]$.

Taking into account that the values of *tpr* and *fpr* are also estimates, we obtain an approximation p'' of the actual proportion p. These two rates are crucial in understanding quantification methods as proposed by Forman because they are designed under the assumption that the a priori class distribution, $Pr(y)$, changes, but the within-class probability densities, $Pr(\boldsymbol{x}|y)$, do not (see Section 2.3.2).

Note that due to (2.7), only the *tpr* fraction of any shift in P will be perceived by the already-trained classifier ($TP = tpr \cdot P$). Observe also that when $Pr(\boldsymbol{x}|y)$ is preserved the proportion of TP versus FN is also constant. That is, the estimation of *tpr* in Figure 2.2a is still representative of the data distribution in Figure 2.2b, but not in Figure 2.2c.

In the same way, the *fpr* fraction of N is misclassified by any CC-based method as false positives ($FP = fpr \cdot N$). According to all these observations, Forman [2008] states the following theorem and its corresponding proof:

Theorem 3.1 (Forman's Theorem)**.** *For an imperfect classifier, the CC method will underestimate the true proportion of positives p in a test set for $p > p^*$, and overestimate for $p < p^*$, where p^* is the particular proportion at which the CC*

method estimates correctly; i.e., the CC method estimates exactly p^* for a test set having p^* positives.

Proof. The expected prevalence p' of classifier outputs over the test set, written as a function of the actual positive prevalence p, is

$$p'(p) \;=\; tpr \cdot p \;+\; fpr \cdot (1-p) \tag{3.2}$$

Given that $p'(p^*) = p^*$, then for a strictly different prevalence $p^* + \Delta$, where $\Delta \neq 0$, CC does not produce the correct prevalence

$$\begin{aligned} p'(p^* + \Delta) &= tpr \cdot (p^* + \Delta) \;+\; fpr \cdot (1 - (p^* + \Delta)) \\ &= tpr \cdot p^* \;+\; fpr \cdot (1 - p^*) + (tpr - fpr) \cdot \Delta \\ &= p'(p^*) + (tpr - fpr) \cdot \Delta \\ &= p^* + (tpr - fpr) \cdot \Delta. \end{aligned}$$

Moreover, since Forman's theorem assumes an imperfect classifier, then we have that $(tpr - fpr) < 1$, and thus

$$p'(p^* + \Delta) \begin{cases} < p^* + \Delta & \text{if } \Delta > 0 \\ > p^* + \Delta & \text{if } \Delta < 0. \end{cases}$$ □

The overall conclusion is that a non-adjusted classifier, like CC, tends to underestimate the prevalence of the positive class when it rises, and vice-versa (see Figure 3.1).

3.2.1 Quantification via threshold selection policies

Provided that the AC method allows using any base classifier for building a quantifier, the underlying learning process has attracted little attention. Much of the efforts are, again, due to Forman, who proposed a collection of methods that are based on training a linear SVM classifier, with a posterior calibration of its threshold. The main difference among these methods is the threshold selection policy employed, trying to alleviate some drawbacks of AC correcting formula from alternative perspectives.

A key problem related with the AC method is that its performance depends mostly on the degree of imbalance in the training set, degrading when the positive class is scarce [Forman, 2006]. In this case the underlying classifier tends to minimize the false positive errors, which usually implies a low *tpr* [see Fawcett, 2004] and a small denominator in Equation (3.1). This fact implies a high sensitivity to fluctuations in the estimation of *tpr* or *fpr*.

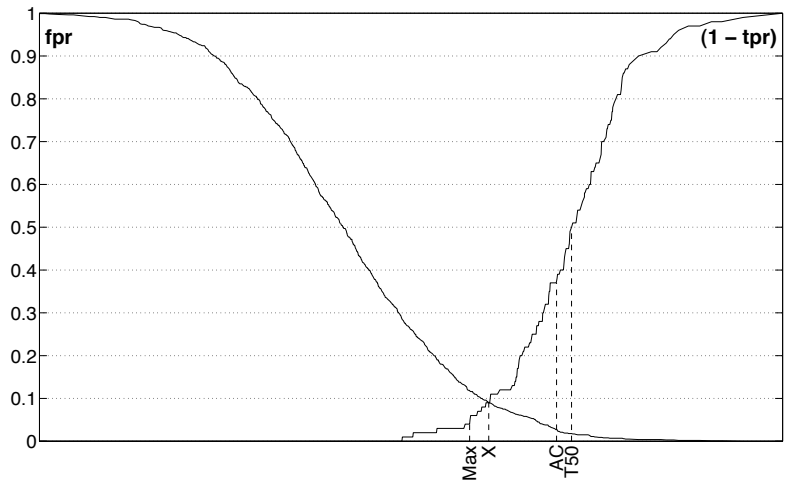


Figure 3.2: Graphical interpretation of threshold selection policies.

For highly imbalanced situations, the main intuition is that selecting a threshold that allows more true positives, even at the cost of many more false positives, could deserve better quantification performance. The objective is to choose those thresholds where the estimates of *tpr* and *fpr* have less variance or where the denominator in Equation (3.1) is big enough to be more resistant to estimation errors.

The three main policies proposed by Forman are:

- **X**: selects a threshold where $fpr = 1 - tpr$, avoiding the tails of both curves.
- **T50**: selects a threshold with $tpr = 50\%$, avoiding only the tails of *tpr* curve.
- **Max**: selects a threshold that maximizes the difference between *tpr* and *fpr*

Figure 3.2 shows an example of *fpr* and $1-tpr$ curves, obtained from a dataset with 1000 negative and 100 positive instances. Since there are many more negatives, the *fpr* curve is smoother than that of $1-tpr$. Note also that, as expected, the default threshold used in AC is clearly biased towards the negative class, with a very low *fpr* value and an intermediate value for *tpr*. T50 fixes a specific performance over the positive class, which in this example degrades the *tpr* from the AC threshold, although this depends on the complexity of each problem.

In the example presented in Figure 3.2, the point where the two curves intersect is clearly identified by means of the X policy. However, this point is not always unique, because although one curve is monotonically decreasing and the other

is monotonically increasing, there could be parts of both curves where they are constant. The same occurs with Max and T50; there could be several points where $(tpr - fpr)$ are maximized or where $tpr = 50\%$. Therefore, several strategies could be applied to resolve these ties. For the experiments presented in this dissertation, we always select the lowest threshold fulfilling the specific conditions, i.e., the first occurrence starting from the left, which provides the highest *tpr* value.

Median Sweep (MS)

Notwithstanding, there is another problem related with all these methods, due to the fact that the estimation of *tpr* and *fpr* may differ significantly from the real values. Thus, Forman proposed a more advanced method, *Median Sweep* (MS), based on estimating the prevalence for all thresholds during testing, in order to compute their median. He also points out that this method may show an odd behavior when the denominator in Equation (3.1) is too small, recommending to discard any threshold with $tpr - fpr < 1/4$. However, he does not make any recommendation in case there is no threshold that avoids that restriction. Therefore, we decide to fix these missing values with the Max method, which provides the threshold with the greatest value for that difference. The empirical results provided in [Forman, 2008] suggests that the MS method is comparatively consistent, claiming that it may smooth over estimation errors like in bootstrap-based algorithms.

3.2.2 Quantification via probability estimators

Bella et al. [2010] have recently developed a family of methods identified by them as *probability estimation & average*. Their core proposal is to develop a probabilistic version of AC. First they introduce a simple method called *Probability Average* (PA), which is clearly aligned with CC. The key difference is that in this case the classifier learned is probabilistic (see also [Bella et al., 2013a]). Once the probability predictions are obtained from test dataset, the average of these probabilities is computed for the positive class as follows

$$p' = \hat{\pi}^{PA}_{Test}(\oplus) = \frac{1}{S}\sum_{i=1}^{S} Pr(y_i = 1|\boldsymbol{x}_i). \tag{3.3}$$

As might be expected, when the proportion of positives changes between training and test, then PA will underestimate or overestimate as with happens with CC. Therefore, they propose an enhanced version of this method, termed *Scaled Probability Average* (SPA). In resemblance with CC and AC, the estimation p' obtained from Equation (3.3) is corrected according to a simple scaling formula:

$$p'' = \hat{\pi}^{SPA}_{Test}(\oplus) = \frac{p' - FP_{pa}}{TP_{pa} - FP_{pa}}, \tag{3.4}$$

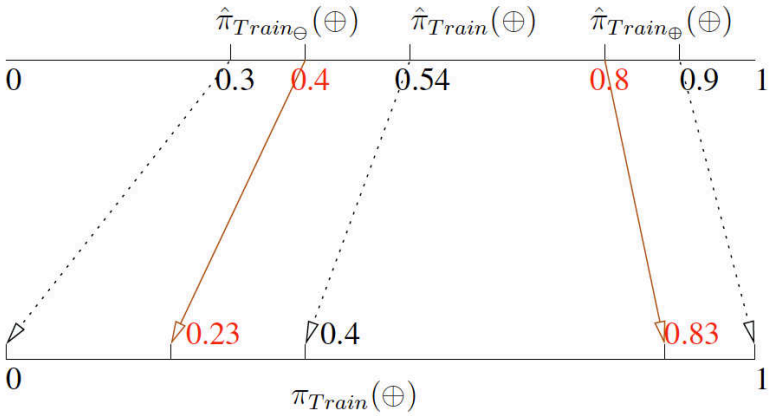


Figure 3.3: Graphical interpretation of SPA method [Bella et al., 2010]. The limits in the training set are placed at 0.3 and 0.9. The estimated value for the training set is 0.54 whereas the actual proportion in the training set is 0.4. The scaling would move a case at 0.4 to 0.23 and a case at 0.8 to 0.83.

where TP_{pa} and FP_{pa} are values estimated from the training set, defined respectively as *TP probability average* or *positive probability average of the positives*

$$TP_{pa} = \hat{\pi}_{Train_\oplus}(\oplus) = \frac{\sum_{\{i|y_i=1\}} Pr(y_i = 1|\boldsymbol{x}_i)}{\#\{y_i = 1\}}, \tag{3.5}$$

and *FP probability average* or *positive probability average of the negatives*

$$FP_{pa} = \hat{\pi}_{Train_\ominus}(\oplus) = \frac{\sum_{\{i|y_i=-1\}} Pr(y_i = 1|\boldsymbol{x}_i)}{\#\{y_i = -1\}}. \tag{3.6}$$

The expression defined in Equation(3.4) yields a probabilistic version of Forman's adjustment defined with Equation (3.1). The graphical interpretation of this scaling procedure is presented in Figure 3.3. In their experiments, SPA method outperforms CC, AC and T50; although they do not compare their proposal with other methods based on threshold selection policies like Max, X or MS.

3.3 Quantification-oriented learning

Esuli and Sebastiani [2010] suggest the first training approach explicitly designed to learn a binary quantifier, in the context of a sentiment quantification task. However, a key limitation is that they neither implement nor validate it in

practice. For this dissertation, we present the first experiment results based on this approach, compared with other state-of-the-art algorithms and our own quantification-oriented learning proposal (see Chapter 6).

They justify the motivation for applying quantification on opinion mining problems arguing that it has been traditionally neglected whether the analysis of large quantities of text should be carried out at the individual or aggregate level. Actually, this is an important issue because some applications like open-answer classification for customer satisfaction analysis demand attention at the individual level, while others such as open-answer classification for market research or review classification for product or brand positioning are best analyzed at the aggregate level.

Moreover, they raise and interesting point regarding the fact that it would seem obvious that the more we improve a classifier's accuracy at the individual level, the higher its accuracy at the aggregate level will become, leading to a misleading conclusion about that the only way to improve a classifier's ability to correctly estimate the test class distribution is to improve its ability to classify each individual. However, this mostly depends on what we mean by *accuracy at the individual level*.

For instance, most standard classification loss functions (see Section 6.2.1) may reject a classifier h_1 with $FP = 50$ and $FN = 50$ when comparing it with a classifier h_2 with $FP = 10$ and $FN = 0$. Nevertheless, they state that h_1 is better than h_2 according to to any reasonable measure for evaluating quantification accuracy. Indeed, h_1 is a perfect quantifier since FP and FN are equal and thus compensate each other, so that the distribution of the test items is estimated perfectly.

Although the training method that they describe is also based on building a classifier, in this case the learning process optimizes the quantification error, without taking into consideration the classification performance of the model. Essentially, as their focus is on binary quantification problems, they argue that compensating the errors between both classes provides the means for obtaining better quantifiers. Therefore, the key idea is to optimize a metric derived from the expression $|FP - FN|$. That is, a perfect quantifier should simply counterbalance all false positives with the same amount of false negative errors. In fact, all loss functions reviewed in Section 2.2 reach their optimum when this difference is 0.

3.4 Alternative approaches

The quantification methods described so far are those that are directly related with the core research objectives of this dissertation. In this last section we briefly introduce other related approaches for the sake of completeness.

A recent work by Bella et al. [2013b] establishes a more general framework for quantification, proposing a new taxonomy of quantification tasks and presenting the first experimental study focused on quantification for regression. One of they contributions is the distinction between the estimation of the expected value versus the whole distribution, for both classification and regression. After discussing that current quantification methods are not suitable for aggregating regression outcomes they propose several alternatives based on discretization that show good performance in practice.

This last work is closely related with the discussion presented by Esuli and Sebastiani [2010] about the application of quantification for ordinal regression, suggesting the use of *Earth Mover's Distance* as quality measure. Moreover, another recent work by Esuli and Sebastiani [under review] introduce a tentative application domain for quantification of radiology reports under the ACR Index.

Forman [2008] introduced in turn the problem of quantification for cost sensitive learning (see also [Hernández-Orallo et al., 2013]), while setting the foundations for extending binary quantification for the general multi-class scenario. There exist also other approaches regarding class distribution estimation under prior-shift, which mainly cover distribution matching through semi-supervised leaning [du Plessis and Sugiyama, 2012] and Hellinger distance [González-Castro et al., 2013, 2010], among others.

Chapter 4

Methodology for statistical comparisons of quantifiers

This chapter describes the methodology followed on the experiments discussed in this dissertation. We have also applied this experimental setup in practice, publishing our results in two articles: [Barranquero et al., 2013] (see Chapter 5) and [Barranquero et al., under review] (see Chapter 6).

Since the beginning of our research, we observed the lack of a standardized experimental design for comparing quantification algorithms, mainly in terms of statistical procedures. That is why we started by studying the most recent approaches, trying to redefine a new methodology that could cover the specific requirements of statistical comparisons of multiple quantifiers, while preserving the core concepts already validated for standard machine learning methodologies.

The key contribution of our proposal is that this new methodology for quantification allows us to analyze relevant properties of these comparatives from a statistical point of view. Furthermore, it also provides meaningful insights about which algorithms are significantly better, with a certain confidence degree, thanks to the adaptation of the two-step Friedman-Nemenyi statistical test [Demšar, 2006; Nemenyi, 1963; Friedman, 1940, 1937], and the redesign of test set generation in stratified k-fold cross-validation (see, e.g., [Refaeilzadeh et al., 2009]).

The fact is that the required experiment methodology for quantification is relatively uncommon and has yet to be properly standardized and validated by machine learning community. Its main difference with respect to traditional classification methodology is that we need to evaluate performance over whole sets, rather than by means of individual classification outputs. Moreover, quantification assessment requires evaluating performance over a broad spectrum of test sets with different class distributions, instead of using a single test set.

Therefore, traditional machine learning techniques for comparing classification models are not directly applicable and needs to be adapted. On next sections we introduce our proposed methodology, which is based in turn on the global guidelines already established by Forman [2008] for evaluating quantifiers.

This new methodology is mostly focused on providing statistical relevance to quantification experiments, adapting techniques borrowed from statistical classification comparatives [García et al., 2010; Garcia and Herrera, 2008; Demšar, 2006], proposing several variations and improvements in order to enhance replicability for comparisons among quantification algorithms.

4.1 Adaptation of stratified k-fold cross-validation

Classical paradigm for error estimation in machine learning assumes that the distributions do not change over time [Hand, 2006; Holte, 2006]. However, this is not the case in real-world domains. For example, in commercial applications customers will change their behavior with changes in prices, products, competition, economic conditions or marketing efforts.

Although stratified cross validation is a mature and widespread procedure [Stone, 1974a,b], with many asymptotic studies [Kohavi, 1995; Stone, 1977], the problem is that it does not take into account any kind of dataset-shift, assuming invariant sample distribution between training and test. There exist other alternatives, like .632+ bootstrap [Efron and Tibshirani, 1997] or jackknife [Efron and Gong, 1983], which try to compensate for possible estimation bias. However they also lack of mechanisms to tackle dataset-shift.

Recent research works have tried to address these drawbacks, but mainly for the case of covariate-shift. Some examples are density-preserving sampling [Budka and Gabrys, 2013], distribution optimally-balanced stratified cross-validation [Moreno-Torres et al., 2012b] (which is an improvement on distribution-balanced stratified cross-validation [Raeder et al., 2010]) and importance-weighted cross-validation [Sugiyama et al., 2007a] (which is focused on problems that already suffer from covariate-shift).

However, we have not found any work that addresses the problem of prior-shift by adapting cross-validation procedure, and that is why we design a new extension for quantification problems under prior-shift. Next section presents our proposed setup in detail.

4.1.1 Error estimation procedure

Assuming that we have a dataset with known positive prevalence, the error estimation procedure proposed consists in a modified version of standard stratified k-fold cross-validation, taking into account specific requirements for quantification, while preserving the original prevalence in all training iterations.

In summary, once a model is trained with k−1 folds, the remaining fold is used to generate several different random test sets with specific positive proportions, by

means of stratified under-sampling [Fleiss et al., 2003]. This setup guarantees that the within-class distributions $P(\boldsymbol{x}|y)$ are maintained between training and test, as stated in Section 2.3.2, due to the fact that resampling processes are uniformly randomized through stratified sampling [Webb and Ting, 2005; Fawcett and Flach, 2005]. Furthermore, given that sampling is controlled from the experiment setup, it allows using any dataset for validation, independently of being from $X \rightarrow Y$ or $Y \rightarrow X$ domains. Notwithstanding, for deployable models the estimated error is only representative of the problem fulfills the learning assumptions of prior-shift (see Section 2.3).

This variation in testing conditions may seem rather unnatural, requiring more appropriate collections of data. Changes in training and test conditions should be extracted directly from different snapshots of the same population, showing natural shifts in their distribution. However, for the time being we have not been able to find publicly-available collections of datasets offering these features, while maintaining at the same time the aforementioned within-class distributions.

The specific parameters of the setup are not fixed, offering the possibility of adapting this procedure to the requirement of a particular experiment. For this dissertation, we use 10-fold cross-validation for training, generating 11 test sets from test fold, with positive prevalence ranging from 0% to 100% in steps of 10%. Provided that we test for 0% and 100% positive proportions, this approach guarantees that all the examples are tested at least once: we are using all the negative and positive examples of test fold, respectively. For further reading, the book chapter by Refaeilzadeh et al. [2009] offers a comprehensive summary with specific statistical properties of different k-fold configurations, in addition to hold-out and leave-one-out.

4.2 Adaptation of Friedman post-hoc tests

As already introduced, the key contribution of our proposed methodology is that it offers the possibility of evaluating which quantifiers are statistically better than others with a certain significance level.

There exist other approaches that propose statistical tests for detecting dataset-shifts Cieslak and Chawla [2009], or for comparing multiple classifiers over multiple datasets [García et al., 2010; Garcia and Herrera, 2008; Demšar, 2006].

However, these are not directly applicable for the case of comparing multiple quantifiers over multiple test prevalences of multiple problems (datasets). Obviously this a more complex scenario, which requires further adaptation.

In next section, we briefly review the available alternatives for comparing classifiers, laying the foundations to present our non-parametric proposal for comparing quantifiers in Section 4.2.4.

4.2.1 Discussion on statistical comparisons of classifiers

The de-facto standard for comparisons of classifiers over multiple datasets was published by [Demšar, 2006] in *Journal of Machine Learning Research* (JMLR). He focus his work in the analysis of new machine learning algorithms, proposing a set of simple, yet safe and robust non-parametric tests for statistical comparisons of classifiers: the Wilcoxon signed ranks test [Wilcoxon, 1945] for comparison of two classifiers and the Friedman test [Friedman, 1940, 1937] for multiple classifiers. He also introduces the corresponding post-hoc tests for comparison of several classifiers over multiple data sets: the Nemenyi test for making all pairwise comparisons [Nemenyi, 1963], and the Bonferroni-Dunn [Dunn, 1961] test for comparing with a control classifier (usually the new proposed method).

Two years later, in the same journal, Garcia and Herrera [2008] propose several alternative tests, which they claim that offer higher power, i.e., lower Type II error or false negative rate (see Table 4.1). Actually, they argue against Demšar's proposal of using Nemenyi post-hoc test because it is very conservative and it may not find any significant difference in most of the experimentations [Yang et al., 2007b; Núñez et al., 2007], requiring to employ many datasets [Yang et al., 2007a].

However, we strongly believe that not only this is not a weakness, but a strength of this method because as Demšar notes: *the resulting conclusions from statistical tests should be drawn cautiously.* Furthermore, statistical tests should not be the deciding factor for or against publishing a well-founded research work. Other merits of the proposed algorithm that are beyond the grasp of statistical testing should also be considered and possibly even favored over pure improvements in predictive power. As a matter of fact, we have prevailed robustness, in terms of lower Type I errors, against statistical power in the design of our proposed methodology (see Section 4.2.4).

4.2.2 State-of-the-art tests for comparing classifiers

As we have already introduced, the analysis of results can be done through two alternatives: single-problem analysis and multiple-problem analysis. The first one corresponds to the study of the performance of several algorithms over a unique dataset. The second one study several algorithms over more than one dataset simultaneously, assuming the fact that each problem has a specific degree of difficulty and that the results obtained among different problems are not comparable.

The single-problem analysis is well-known and is usually found in specialized literature, like grid-search through 5×2 cross-validation [Alpaydm, 1999; Dietterich, 1998]. Although the required conditions for using parametric statistics are not usually checked, a parametric statistical study could obtain similar

Table 4.1: Type I and Type II errors

	True H_0	False H_0
Accept H_0	Success (TP)	Type II (FN)
Reject H_0	Type I (FP)	Success (TN)

conclusions to a nonparametric one. However, in a multiple-problem analysis, a parametric test may reach erroneous conclusions [Demšar, 2006].

On the other hand, a distinction between pairwise and multiple comparison tests is necessary. The former are valid procedures to compare two algorithms and the latter should be used when comparing more than two methods. The main reason that distinguishes both kinds of test is related to the control of the *Family Wise Error Rate* (FWER) [Sheskin, 2003], which is the probability of making false positive discoveries or Type I errors (see Table 4.1) . Intended pairwise tests, such as the Wilcoxon test [Demšar, 2006; Wilcoxon, 1945], do not control the error propagation of making more than one comparison and they should not be used in multiple comparisons.

If we want to make multiple comparisons using several statistical inferences simultaneously, then we have to take into account this multiplicative effect in order to control the family wise error. Demšar [2006] described a set of nonparametric tests for performing multiple comparisons and he analyzed them in contrast to well-known parametric tests in terms of power, obtaining that the nonparametric tests are more suitable for comparisons of machine learning algorithms. He presents the Friedman test [Friedman, 1940, 1937] as base test for multiple comparisons, and some post-hoc procedures, such as Nemenyi [Nemenyi, 1963] for all pair-wise comparisons. For comparisons based on a control method he reviews Bonferroni–Dunn [Dunn, 1961], Holm [Holm, 1979], Hochberg [Hochberg, 1988] and Hommel [Hommel, 1988].

The paper by Garcia and Herrera [2008] is an extension of [Demšar, 2006]. Authors deal in depth some topics related to multiple comparisons involving all the algorithms, as well as computations of adjusted p-values. They also describe additional testing procedures for conducting all pairwise comparisons in a multiple-dataset comparison analysis [Shaffer, 1995, 1986; Rom, 1990; Bergmann and Hommel, 1988].

García et al. [2010] extend the set of non-parametric procedures for performing multiple statistical comparisons between more than two algorithms focusing on the case in which a control treatment is compared against other treatments, presenting an experimental analysis of power and stability of these statistical tests.

4.2.3 Standard formulation of Friedman post-hoc tests

The Friedman test [Friedman, 1940, 1937] is a non-parametric equivalent of the *repeated-measures* or *within-subjects* ANOVA [Fisher, 1959]. It ranks the algorithms for each data set separately, assigning average ranks in case of ties.

Let r_i^j be the rank of the j-th of k algorithms on the i-th of N data sets. The Friedman test compares the average ranks of algorithms:

$$R_j = \frac{1}{N} \sum_i r_i^j. \tag{4.1}$$

Under the null-hypothesis, which states that all the algorithms are equivalent and so their ranks R_j should be equal, the Friedman statistic

$$\chi_F^2 = \frac{12N}{k(k+1)} \left(\sum_j R_j^2 - \frac{k(k+1)^2}{4} \right) \tag{4.2}$$

is distributed according to χ_F^2 with $k-1$ degrees of freedom, when N and k are big enough ($N > 10$ and $k > 5$). For a smaller number of algorithms and datasets, exact critical values have been computed [Demšar, 2006].

After showing that Friedman statistic is undesirable conservative, Iman and Davenport [1980] derive an adjusted statistic

$$F_F = \frac{(N-1)\chi_F^2}{N(k-1) - \chi_F^2}, \tag{4.3}$$

which is distributed according to the F-distribution with $k-1$ and $(k-1)(N-1)$ degrees of freedom. The table of critical values can be found in any statistical book or computed with any standard statistical software.

Nemenyi all-vs-all post-hoc test

When Friedman's null-hypothesis is rejected, we can then proceed with a post-hoc test. The Nemenyi test (Nemenyi, 1963) is similar to the Tukey test [Tukey, 1949] for ANOVA and is used when all pairwise comparisons are tested. The performance of two classifiers is significantly different if the corresponding average ranks differ by at least the critical difference

$$CD = q_\alpha \sqrt{\frac{k(k+1)}{6N}} \tag{4.4}$$

where critical values q_α are based on the Studentized range statistic divided by $\sqrt{2}$ (see Table 4.2a).

k	2	3	4	5	6	7	8	9	10
$q_{0.05}$	1.960	2.343	2.569	2.728	2.850	2.949	3.031	3.102	3.164
$q_{0.10}$	1.645	2.052	2.291	2.459	2.589	2.693	2.780	2.855	2.920

(a) Critical values for two-tailed Nemenyi post-hoc test

k	2	3	4	5	6	7	8	9	10
$q_{0.05}$	1.960	2.241	2.394	2.498	2.576	2.638	2.690	2.724	2.773
$q_{0.10}$	1.645	1.960	2.128	2.241	2.326	2.394	2.450	2.498	2.539

(b) Critical values for two-tailed Bonferroni-Dunn post-hoc test

Table 4.2: Critical values for Friedman post-hoc tests [Demšar, 2006]

Bonferroni-Dunn one-vs-all post-hoc test

If we want to compare all classifiers with a control method, the Bonferroni correction or similar procedures can be used to control the family-wise error in multiple hypothesis testing. Although these methods usually have little power, they are more powerful than the Nemenyi test for one-vs-all comparatives, since the latter adjusts the critical value for making $k(k-1)/2$ comparisons while when comparing with a control we only make $k-1$ comparisons.

The test statistics for comparing the i-th and j-th classifier with these methods is

$$z = (R_i - R_j) \Big/ \sqrt{\frac{k(k+1)}{6N}} \,. \tag{4.5}$$

The z-value is used to find the corresponding probability from the table of normal distribution, which is then compared with an appropriate α. The tests differ in the way they adjust the value of α to compensate for multiple comparisons.

The proposal suggested by Demšar [2006] is Bonferroni-Dunn test [Dunn, 1961], which controls the family-wise error rate by dividing a by the number of comparisons $(k-1)$. An alternative way to compute the same test is to calculate the CD using the same equation as for the Nemenyi test, but using the critical values for $\alpha/(k-1)$ (see Table 4.2b). If we compare these critical values with those presented for Nemenyi in Table 4.2a, it is straightforward to conclude that the power of the post-hoc test is much greater when all classifiers are compared only to a control classifier and not between themselves. Therefore, all pairwise comparisons should only be done when we want to perform a global comparative, but we do not want to test whether a newly proposed method is better than the existing ones.

4.2.4 Statistical comparisons over multiple test prevalences

As in [Demšar, 2006], we propose two adaptations for post-hoc statistical tests. In both cases, the first step consists of a Friedman test [Friedman, 1940, 1937] of the null hypothesis that all approaches perform equally. When this hypothesis is rejected, a Nemenyi post-hoc test [Nemenyi, 1963] (for all-vs-all comparatives) or a Bonferroni-Dunn test [Dunn, 1961] (for one-vs-all comparatives) is then conducted to compare the methods in a pairwise way. All tests are based on the average of the ranks.

The comparisons are performed over several models that are trained with a collection datasets or domains, following a stratified cross-validation procedure, and then evaluated over different test prevalences for each of these datasets. These test sets are generated from each test fold by means of stratified under-sampling (see Section 4.1.1).

As Demšar notes, there are variations of the ANOVA and Friedman tests which can consider multiple repetitions per problem, provided that the observations are independent [Zar, 2009]. However, since each collection of test sets is sampled from the same fold, we cannot guarantee the assumption of independence among them. For the best of our knowledge there is no statistical test that could take this into account for the time being.

Moreover, as proposed by Demšar, multiple resampling from each data set is used only to assess the performance score and not its variance. The sources of the variance are the differences in performance over independent datasets and not on (usually dependent) samples, so the uncertainty about elevated Type I error is not an issue. In addition, given that multiple resampling does not bias the score estimation, various types of cross-validation or leave-one-out procedures can be used without any risk.

The problems with the multiple dataset tests are quite different, even in a sense complementary: the measurements from different datasets are usually incommensurate, and the normality of their distributions and the homogeneity of variance is questionable at best [Demšar, 2006]. Hence, running the algorithms on multiple datasets naturally gives a sample of independent measurements, where comparisons are even simpler than comparisons on a single dataset.

In order to take into account the differences between algorithms over several test prevalences from the same dataset, we first obtain their ranks for each test prevalence and then compute an average rank per dataset, which is used to rank algorithms on that problem. As an alternative, averaging results over all the prevalences that are tested for each dataset suffers the problem of how to handle large outliers and the inconsistency of averaging along different test prevalences, so we do not average results in any case.

We also discard the option of performing the tests for each specific test prevalence

because it would imply analyzing too much results, without offering the possibility of obtaining a global conclusion and with the additional problem of reducing confidence due to the multiplicative effect of the family wise error [Sheskin, 2003] associated to multiple dependent statistical tests.

Therefore, we only consider the original number of datasets to calculate the critical difference (CD) defined in Equation (4.4), rather than using all test cases, resulting in a more conservative value. The reason for this is not only that the assumption of independence is not fulfilled, but also that the number of test cases is not bound. Otherwise, simply taking a wider range of prevalences to test would imply a lower CD value, which appears to be unjustified from a statistical point of view and can be prone to distorted conclusions.

4.3 Additional notes on evaluation methodology

For our experiments we use AE as default error measure, complementing this view with KLD where required (see Section 2.2). The main advantages of KLD are that it may be more appropriate to average over different test prevalences and more suitable for extending the quantification task for multiclass problems. However, a drawback of KLD is that it is less interpretable than other measures, like AE.

Furthermore, we also need to define its output for those cases where P, N, P' or N' are cancelled. This implies that KLD is not properly bounded, obtaining undesirable results, like infinity or indeterminate values, when the actual or estimated proportions tend to 0% or 100%, needing further corrections to be applicable (see Section 2.2.3).

As might be expected, the experimental setting proposed in this chapter have been redesigned several times before reaching the final form followed in this dissertation and its associated publications. In this regard, we would like to point out that having more test prevalences may reduce variance, but may also imply more dependence among test results. This issue should deserve more research for future work, although finally, we have decided to fix the test procedure to 11 prevalences, offering an appropriate tradeoff between range of tests and independence of results.

Other important source of debate is where to stress the relevance on the range from 0% to 100%. For our experiments we distribute the relevance over the whole range, though some reviewers have stated that it may be more appropriate to give higher weight to lower prevalences ($< 25\%$), in order to focus the conclusions on unbalanced scenarios. Nevertheless, our experiments are designed to validate our proposals over a broader range of contexts, and that is why we do not restrict the range of prevalences with any such assumption. However, the methodology that we propose is open to other interpretations, in which the criterium followed to distribute the test prevalences could be neither linear nor uniform. It will depend mostly on the final aim of the experiment.

Chapter 5

Study and adaptation of Nearest Neigbor (NN) algorithms for quantification

This chapter presents a new approach for solving binary quantification problems based on nearest neighbor (NN) algorithms. Our main objective is to study the behavior of these methods in the context of prevalence estimation. We seek for NN-based quantifiers able to provide competitive performance while balancing simplicity and effectiveness. The results of this research have been recently published on *Pattern Recognition* [Barranquero et al., 2013].

Our first intuition was that the inherent behavior of NN algorithms should yield appropriate quantification results based on the assumption that they may be able to *remember* details of the topology of the data. However, the main motivation behind this work is that similar NN approaches have been successfully applied in a wide range of learning tasks, providing simple and competitive algorithms for classification [Cover and Hart, 1967], regression [Hardle, 1992], ordinal regression [Hechenbichler and Schliep, 2004], clustering [Wong and Lane, 1983], preference learning [Broos and Branting, 1994] and multi-label [Zhang and Zhou, 2007] problems, among others.

The key contribution is the adaptation of k-nearest neighbor (KNN) algorithm, by means of the correction defined with Equation (3.1), as well as the proposal of two effective weighting strategies, PWK and PWK$^{\alpha}$, which stand out among state-of-the-art quantifiers. These weight-based proposals are the only ones that offer statistical differences with respect to less robust algorithms, like CC or AC. Our final aim is to explore the applicability of NN algorithms for binary quantification, using standard benchmark datasets from different domains.

5.1 Nearest neighbour quantification

NN approaches present significative advantages in order to build an AC-based quantifier. In fact, they allow to implement more efficient methods for estimating *tpr* and *fpr*, which are required to compute the quantification correction defined in Equation (3.1). The standard procedure for the computations of these rates is cross-validation [Forman, 2008]. When working with SVM as base-learner for AC, we have to re-train a model for each partition, while NN approaches allow us to compute the distance matrix once and use it for all partitions. Thus, we can estimate *tpr* and *fpr* at a small computational cost, even applying a *leave-one-out* (LOO) procedure, which may provide a better estimation for some domains.

5.1.1 K-nearest neighbor (KNN)

One of the best known NN-based methods is the k-nearest neighbor (KNN) algorithm. Despite its simplicity, it has been demonstrated to yield very competitive results in many real world situations. In fact, Cover and Hart [1967] pointed out that the probability of error of the NN rule is upper bounded by twice the Bayes probability of error.

Given a binary problem $D = \{(\boldsymbol{x}_i, y_i) : i = 1 \dots S\}$, consisting of a collection of labels $\bar{y} = (y_1, ..., y_n)$ and their corresponding predictor features $\bar{\boldsymbol{x}} = (\boldsymbol{x_1}, ..., \boldsymbol{x_n})$, in which $\boldsymbol{x}_i$ represents an object of the input space $\mathcal{X}$ and $y_i \in \mathcal{Y} = \{+1, -1\}$. Thus, for a test example $\boldsymbol{x}_j$, the resulting output for KNN is computed as

$$h(\boldsymbol{x}_j) = \text{sign}\left(\sum_{i \sim j}^{k} y_i \right); \tag{5.1}$$

where $i \sim j$ denotes the k-nearest neighbors of the test example $\boldsymbol{x}_j$.

Regarding the selection of k, Hand and Vinciotti [2003] pointed out that, as the number of neighbors determines the bias versus variance tradeoff of the model, the value assigned to k should be smaller than the smallest class. This is especially relevant with unbalanced datasets, which is the common case in many domains. Another widely cited study, by Enas and Choi [1986], proposes $n^{2/8}$ or $n^{3/8}$ as heuristic values, arguing that the optimal k is a function of the dimension of the sample space, the size of the space, the covariance structure and the sample proportions. In practice, however, this optimal value is usually determined empirically through a standard cross-validation procedure. Moreover, the selection of an appropriate metric or distance is also decisive and complex, in which the Euclidean norm is usually the default option (known as *vanilla* KNN). For our study we decided to simplify all these decisions where possible, limiting our search to selecting the k value that leads to better empirical performance through a grid-search procedure (see Section 5.2.2), and using the Euclidean distance.

5.1.2 Proportion-weighted KNN (PWK$^\alpha$)

Although KNN has provided competitive quantification results in our experiments, Forman states that quantification models should be ready to learn from highly imbalanced datasets, like in one-vs-all multiclass scenarios or in narrowly defined categories. This gave us the idea of complementing it with weighting policies, mainly those depending on class proportions, in order to counteract the bias towards the majority class.

The main drawback when addressing the definition of a suitable strategy for any weight-based method is the broad range of weighting alternatives depending on the focus of each problem or application. Two major directions for assigning weights in NN-based approaches are identified by Kang and Cho [2008]. On the one hand, we can assign weights to features or attributes before distance calculation, usually through specific kernel functions or flexible metrics [Domeniconi et al., 2002]. On the other hand, we can assign weights to each neighbor after distance calculation. We have focused our efforts on the latter approach.

This problem has already been studied by Tan [2005], as the core of neighbor-weighted k-nearest neighbor (NWKNN) algorithm, mostly aimed at unbalanced text problems. Tan's method is based on assigning two complementary weights for each test document: one based on neighbour distributions and another based on similarities between documents. The former assigns higher relevance to smaller classes and the latter adjusts the contribution of each neighbor by means of its relative distance to the test document. Similarly as in (5.1), for a binary problem and given a test example $\boldsymbol{x}_j$, the estimated output can be obtained as

$$h(\boldsymbol{x}_j) = \text{sign}\left(\sum_{i \sim j}^{k} \text{sim}(\boldsymbol{x}_i, \boldsymbol{x}_j)\ y_i\ w_{y_i}\right). \tag{5.2}$$

We discarded similarity score for our study,

$$h(\boldsymbol{x}_j) = \text{sign}\left(\sum_{i \sim j}^{k} y_i\ w_{y_i}\right), \tag{5.3}$$

simplifying the notation and the guidelines for computing the class weights described by Tan. In summary, he proposes class weights that balance the relevance between classes, compensating the natural influence bias of bigger classes in multi-class scenarios. He also includes an additional parameter, which can be interpreted as a shrink factor: when this parameter grows, the penalization of bigger classes is softened progressively. In this paper, we use α to identify this parameter. We compute each class weight during training as the adjusted quotient

between the cardinalities of that class (N_c) and the minority class (M)

$$w_c^{(\alpha)} = \left(\frac{N_c}{M}\right)^{-1/\alpha}, \text{with } \alpha \geq 1 \tag{5.4}$$

Therefore, the bigger the class size observed during training, the smaller its weight. To illustrate this fact, Table 5.1 shows the weights assigned to one of the classes, varying its prevalence from 1% to 99% for different values of α. Note that when we compute the weight of the minority class, or when the problem is balanced (50%), we always get a weight of 1; i.e., there is no penalization. However, when we compute the weight for the majority class, we get a penalizing weight ranging from 0 to less than 1. The simplified algorithm defined by (5.3) and (5.4) is renamed as the proportion-weighted k-nearest neighbor (PWK$^\alpha$) algorithm.

Table 5.1: PWK$^\alpha$ weights w.r.t. different training prevalences (binary problem)

α	1%	$\cdots$	50%	60%	70%	80%	90%	99%
1	1	$\cdots$	1	0.67	0.43	0.25	0.11	0.01
2	1	$\cdots$	1	0.82	0.65	0.50	0.33	0.10
3	1	$\cdots$	1	0.87	0.75	0.63	0.48	0.22
4	1	$\cdots$	1	0.90	0.81	0.71	0.58	0.32
5	1	$\cdots$	1	0.92	0.84	0.76	0.64	0.40

5.1.3 Naïve proportion-weighted KNN (PWK)

As an alternative to Equation (5.4), we propose the following class weight

$$w_c = 1 - \frac{N_c}{S}, \tag{5.5}$$

which produces equivalent weights for $\alpha = 1$. This expression makes it easier to see that each weight w_c is inversely proportional to the size of the class c, with respect to the total size of the sample, denoted by S.

Theorem 5.1. *For any binary problem, the prediction rule in Equation* (5.3) *produces the same results regardless of whether class weights are calculated using Equation* (5.4) *or Equation* (5.5)*, fixing* $\alpha = 1$.

Proof. Let c_1 be the minority class and c_2 the majority class, then the idea is to prove that weights $w_{c_1}^{(1)}$ and $w_{c_2}^{(1)}$, computed by means of Equation (5.4), are equal to their respective w_{c_1} and w_{c_2}, computed by means of Equation (5.5), when

they are divided by a unique constant, which happens to be equal to w_{c_1}. For the majority class:

$$w_{c_2}^{(1)} = \frac{N_{c_1}}{N_{c_2}} = \frac{N_{c_1}/S}{N_{c_2}/S} = \frac{1 - N_{c_2}/S}{1 - N_{c_1}/S} = \frac{w_{c_2}}{w_{c_1}}.$$

Given that by definition $w_{c_1}^{(1)} = 1$, we can rewrite it as $w_{c_1}^{(1)} = w_{c_1} \,/\, w_{c_1}$. Thus, if we fix $\alpha = 1$ in (5.4) and divide all the weights obtained from (5.5) by the minority class weight, w_{c_1}, the weights obtained from both equations are equivalent and prediction results are found to be equal. □

The combination of (5.3) and (5.5) is identified as PWK in our experiments. We initially considered this simplified PWK method as a naïve baseline for weighted NN approaches. However, despite their simplicity, the resulting models have shown competitive results in our experiments. Moreover, PWK fits both binary and multiclass problems. Within a binary context, Equation (5.5) weights each class in terms of the training proportion of the opposite class; while in multiclass scenarios, each class is weighted with respect to the training proportion of all other classes.

5.1.4 Additional notes on NN algorithms

The key benefit of PWK$^\alpha$ over PWK is that the former provides additional flexibility to further adapt the model to each dataset through its α parameter, usually increasing precision when α grows, but decreasing recall. Conversely, PWK$^\alpha$ requires a more expensive training procedure due to the calibration of this free parameter. Our experiments in Section 5.3 suggest no statistical difference between both, so the final decision for a real-world application should be taken in terms of the specific needs of the problem, the constraints of the environment, or the complexity of the data, among others.

It is worth noting that for binary problems when α tends to infinity Equation (5.4) produces a weight of 1 for both classes, and given that PWK$^\alpha$ is equivalent to PWK when $\alpha = 1$, then KNN and PWK can be interpreted as particular cases of PWK$^\alpha$. The parameter α can be thus reinterpreted as a tradeoff between traditional KNN and PWK.

The exhaustive analysis of alternative weighting approaches for KNN is beyond the scope of this dissertation. A succinct review of weight-based KNN proposals is given in [Kang and Cho, 2008], including attractive approaches for quantification like weighting examples in terms of their classification history [Cost and Salzberg, 1993], or accumulating the distances to k neighbors from each of the classes in order to assign the class with the smallest sum of distances [Hattori and Takahashi, 1999]. Tan has also proposed further evolutions of his NWKNN, such as the *DragPushing* strategy [Tan, 2006], in which the weights are iteratively refined taking into account the classification accuracy of previous iterations.

5.2 Experimental setup

The specific settings described in this section follow the general principles introduced in Chapter 4. That is, we use standard datasets with known positive prevalence, along with the adaptations of stratified cross-validation and Friedman-Nemenyi statistical test. The main objective is to evaluate state-of-the-art quantifiers, comparing them with simpler NN-based models.

5.2.1 Datasets

In order to enable fair comparisons among our proposals and those presented in the literature, we have selected a collection of datasets from the UCI Machine Learning Repository [Frank and Asuncion, 2010], taking problems with ordinal or continuous features with at the most three classes, and ranges from 100 to 2,500 examples. The summary of the 24 datasets meeting these criteria is presented in Table 5.2.

Notice that the percentage of positive examples goes from 8% to 78%. This fact offers the possibility of evaluating the methods over a wide spectrum of different training conditions. For datasets that originally have more than two classes, we followed a one-vs-all decomposition approach. We also extracted two different datasets from *acute*, which provides two alternative binary labels.

For datasets with positive class over 50%, *ctg.1* in our experiments, an alternative approach when using T50 method is to reverse the labels between both classes. We have tried both setups, but we have found no significant differences. Therefore, we decided to preserve the actual labeling, because we consider that it is more relevant to perform the comparisons between systems under the same conditions.

Moreover, as given that algorithms use Euclidean distance and linear kernels, which are not scale invariant, we have applied a normalization for *transfusion* and *wine* datasets. After this normalization all feature columns in these datasets have mean 0 and standard deviation 0.5. The transformation enables that both SVM and NN algorithms achieve more consistent results.

5.2.2 Error estimation

We collected results from all datasets, applying a stratified 10-fold cross-validation for each of them, preserving their original class distribution. After each training, we always assess the performance of the resulting model with 11 test sets generated from the remaining fold, varying the positive prevalence from 0% to 100% in steps of 10% (see Section 4.1.1).

We therefore performed 240 training processes and 2,640 tests for every system we evaluated. This setup generates 264 cross-validated results for each algorithm, that is, 24 datasets × 11 test distributions.

5.2.3 Algorithms

As one of the experiment baselines, we selected a dummy method that always predicts the distribution observed in training data, irrespective of the test distribution, which is denoted by BL. This allows us to verify the degree of improvement provided by other methods, that is, the point upon which they learn something significant. Although this baseline can be considered a *non-method*, it is able to highlight deficiencies in some algorithms. Actually, as we shall discuss later in Section 5.3, there are methods that do not show significant differences with respect to BL.

We chose CC, AC, Max, X, T50 and MS as state-of-the-art quantifiers from Forman's proposals, considering CC as primary baseline. The underlying classifier for all these algorithms is a linear SVM from the *LibSVM* library [Chang and Lin, 2011]. The process of learning and threshold characterization, discussed in Section 5.2.6, is common to all these models, reducing the total experiment time and guaranteeing an equivalent base SVM for them all.

The group of NN-based algorithms consists of KNN, PWK and PWK$^{\alpha}$. For the sake of simplicity, we always use the standard Euclidean distance and perform a grid-search procedure to select the best k value, as discussed in Section 5.1. It is worth noting that we apply Forman's correction defined in (3.1) for all these NN algorithms. The main objective is to verify whether we can obtain competitive results with instance-based methods, while taking into account the formalisms already introduced by Forman. In contrast with threshold quantifiers, those based on NN rules do not calibrate any threshold after learning the classification model.

5.2.4 Parameter tuning via grid-search

We use a grid-search procedure for parameter configuration, consisting of a 2×5 cross-validation [Alpaydm, 1999; Dietterich, 1998]. The loss function applied for discriminating the best values is the geometric mean (GM) of *tpr* and *tnr* (*true negative rate*, defined as TN/N), i.e., *sensitivity* and *specificity*. This measure is particularly useful when dealing with unbalanced problems in order to alleviate the bias towards the majority class during learning [Barandela et al., 2003]. For those algorithms that use SVM as base learner, the search space for the regularizer parameter C is $\{0.01, 0.1, 1, 10, 100\}$. For NN-based quantifiers, the range for k parameter is $\{1, 3, 5, 7, 11, 15, 25, 35, 45\}$. In the case of PWK$^{\alpha}$, we also adjust parameter α over the integer range from 1 to 5. The grid-search for NN models

Table 5.2: Summary of datasets used for the experiments

Dataset	Identifier	*Size*	*Attrs.*	*Pos.*	*Neg.*	*%pos.*
Acute Inflammations (urinary bladder)	acute.a	120	6	59	61	49%
Acute Inflammations (renal pelvis)	acute.b	120	6	50	70	42%
Balance Scale Weight & Distance (left)	balance.1	625	4	288	337	46%
Balance Scale Weight & Distance (balanced)	balance.2	625	4	49	576	8%
Balance Scale Weight & Distance (right)	balance.3	625	4	288	337	46%
Contraceptive Method Choice (no use)	cmc.1	1473	9	629	844	43%
Contraceptive Method Choice (long term)	cmc.2	1473	9	333	1140	23%
Contraceptive Method Choice (short term)	cmc.3	1473	9	511	962	35%
Cardiotocography Data Set (normal)	ctg.1	2126	22	1655	471	78%
Cardiotocography Data Set (suspect)	ctg.2	2126	22	295	1831	14%
Cardiotocography Data Set (pathologic)	ctg.3	2126	22	176	1950	8%
Haberman's Survival Data	haberman	306	3	81	225	26%
Johns Hopkins University Ionosphere	ionosphere	351	34	126	225	36%
Iris Plants Database (setosa)	iris.1	150	4	50	100	33%
Iris Plants Database (versicolour)	iris.2	150	4	50	100	33%
Iris Plants Database (virginica)	iris.3	150	4	50	100	33%
Sonar, Mines vs. Rocks	sonar	208	60	97	111	47%
SPECTF Heart Data	spectf	267	44	55	212	21%
Tic-Tac-Toe Endgame Database	tictactoe	958	9	332	626	35%
Blood Transfusion Service Center	transfusion	748	4	178	570	24%
Wisconsin Diagnostic Breast Cancer	wdbc	569	30	212	357	37%
Wine Recognition Data (1)	wine.1	178	13	59	119	33%
Wine Recognition Data (2)	wine.2	178	13	71	107	40%
Wine Recognition Data (3)	wine.3	178	13	48	130	27%

is easily optimizable, because once the distance matrix has been constructed and sorted, the computations with different values of k can be obtained almost straightforwardly.

5.2.5 Estimation of *tpr* and *fpr* characteristics

All quantification outputs are adjusted by means of Equation (3.1), except for BL and CC. The estimations of *tpr* and *fpr* for quantification corrections are obtained through a standard 10-fold cross-validation in all cases. Other alternatives like 50-fold CV or LOO are discarded because they are much more computationally expensive for SVM-based models. In the case of NN-based algorithms, the straightforward method for estimating these rates is by means of the distance matrix, applying a LOO procedure. However, we finally decided to use only one common estimation method for all competing algorithms for fairer comparisons.

5.2.6 Additional notes on learning methodology

The learning procedure established by Forman [2008] does not involve the calibration of the underlying SVM parameters. He states that the focus is no longer on the accuracy of individual outputs, but on the correctness of the aggregated estimations. Thus, in some sense, the *goodness* of the original classifier is not relevant, as long as its predictions are correctly adjusted. This approach is followed for comparing the proposed quantification-oriented learners in Chapter 6.

However, in this chapter we present a new quantification approach based on Nearest Neighbour rules, which require the calibration of parameter k. Realizing that it would be unfair to compare them with SVM models without previously adjusting the regularization parameter C, our proposed learning process for the experimental setup starts by selecting the best value for this regularization parameter through a grid-search procedure (see Section 5.2.2). Once this optimized model has been obtained, its default threshold is varied over the spectrum of raw training outputs, and the *tpr* and *fpr* values for each of these thresholds are estimated through cross-validation. After collecting all this information, several threshold selection policies can be applied in order to prepare the classifier for the following step, as already set out in Section 3.2.1. Each of these strategies provides a derived model which is ready to be used and compared.

5.3 Empirical analysis

Given that we obtain almost equivalent conclusions for AE and KLD results in Sections 5.3.1–5.3.3, for the sake of readability we focus our analyses on AE scores

for Sections 5.3.4 and 5.3.5. In fact, one of the key drawbacks that we encountered during the analysis of these experiments is the broad range of standpoints that can be adopted and the information overload with respect to classification problems. Therefore, we consider that coherent and meaningful summaries of this information are crucial to understand, analyze and discuss the results properly.

5.3.1 Overview analysis of results

The first approach that we followed is to represent the AE results for all 11 test conditions in all 24 datasets by means of a boxplot of each system under study. Thus, in Figure 5.1a we can observe the range of errors for every system. Each box represents the first and third quartile by means of the lower and upper side respectively and the median or second quartile by means of the inner red line. The whiskers extend to the most extreme results that are not considered outliers, while the outliers are plotted individually with crosses. In this case, we consider as outliers any point greater than the third quartile plus 1.5 times the inter-quartile range. Note that we are not discarding the outliers for any computation, we are simply plotting them individually.

We distinguish four main groups in Figure 5.1a according to the learning procedure followed. The first one comprises only BL, covering a wide range of the spectrum of possible errors. This is probably due to the varying training conditions of each dataset, given that this system always predicts the proportion observed during training. The second group, including CC and AC, shows strong discrepancies between actual and estimated prevalences of up to 100% in some outlier cases. These systems appear to be quite unstable under specific circumstances, which we shall analyze later. The third group includes T50, MS, X and Max, all of which are based on threshold selection policies (see Section 3.2.1). However, as we shall also discuss later, the T50 method stands out as the worst approach in this group due to the evident upward shift of its box. The final group comprises NN-based algorithms: KNN, PWK and PWK$^\alpha$. The weighted versions of this last group offer the most stable results, with the third quartile below 15% in all cases. The weight-based versions present maximum outlier values below 45%.

Figure 5.1a provides other helpful insights regarding the algorithms under study. Taking into account the main elements of each box, we can observe that PWK and PWK$^\alpha$ stand out as the most compact systems in terms of the inter-quartile range. Both of them have their third quartile, their median and their first quartile around 10%, 5% and 2.5%, respectively. Note also that most of the models have a median AE of around 5%, meaning that 50% of the tests over those systems appear to yield competitive quantification predictions. Once again, however, the major difference is highlighted by the upper tails of the boxes, including the third quartile, the upper whisker and the outliers. From the shape and position of the boxes, KNN, Max, X and MS also appear to be noteworthy.

5.3.2 Friedman-Nemenyi statistical test

According to the experimental methodology presented in Chapter 4, we have carried out an adapted version of Friedman-Nemenyi. The experiment includes a comparison of 10 algorithms over 24 datasets or domains, tested over 11 different prevalences, resulting in 264 test cases per algorithm. As already discussed on Section 4.2, since each collection of 11 test sets is sampled from the same fold, we cannot guarantee the assumption of independence among them. Hence, in order to take into account the differences between algorithms over several test prevalences from the same dataset, we first obtain their ranks for each test prevalence and then compute an average rank per dataset, which is used to rank algorithms on that domain.

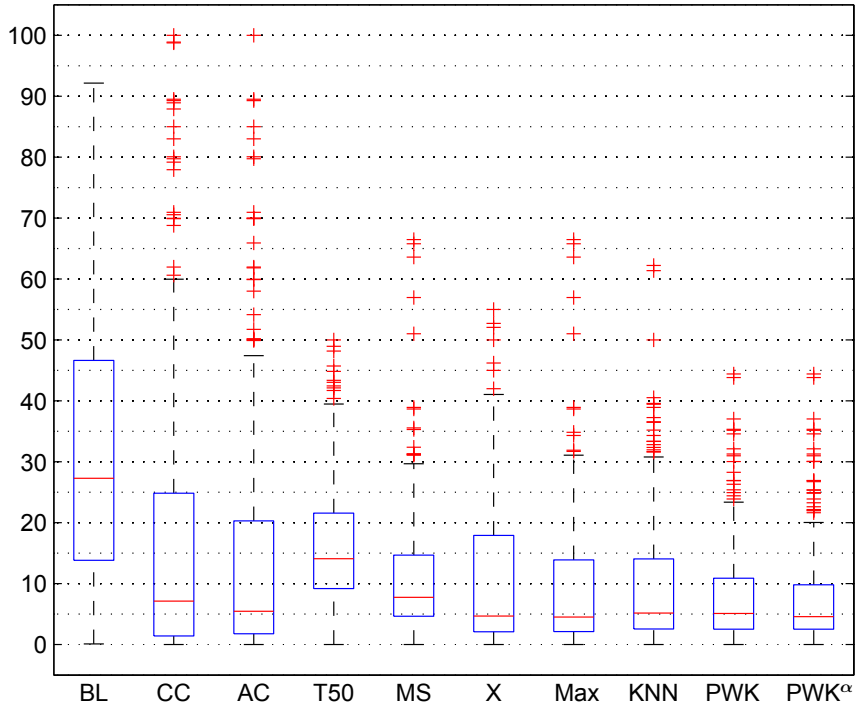
Friedman's null hypothesis is rejected at the 5% significance level and the CD for the Nemenyi test with 24 datasets and 10 algorithms is 2.7654. The overall results of the Nemenyi test are shown in Figure 5.1b, in which each system is represented by a thin line, linked to its name on one side and to its average rank on the other. The thick horizontal segments connect models that are not significantly different at a confidence level of 5%. Therefore, this plot suggests that PWK$^{\alpha}$ and PWK are the models that perform best in this experiment in terms of AE loss comparison for Nemenyi's test. In this setting, we have no statistical evidence of differences between the two approaches. Neither do they show clear differences with KNN, Max or X. We can only appreciate that PWK$^{\alpha}$ and PWK are significantly better than CC, AC, MS, T50 and BL; Max is still connected with CC and MS, while X and KNN are also connected with AC. It is worth noting that neither AC nor T50 show clear differences with respect to BL, suggesting a lack of consistency in the results provided by the former systems.

5.3.3 Complementary analysis of KLD measurements

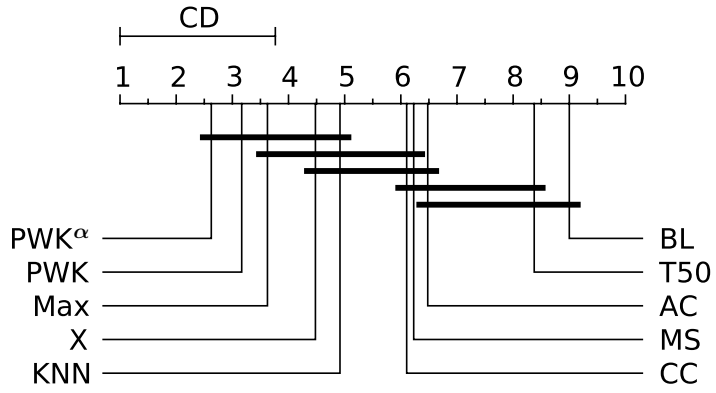
This section presents the results of the experiment in terms of KLD, which are omitted in previous discussion. In summary, we observe that AE results provide similar statistical evidences as with KLD. However, we consider that AE results are more interpretable because they have actual meaning.

KLD values presented in Figure 5.2 show a similar shape with respect to the AE boxplot presented in Figure 5.1. It is also worth noting that in Figure 5.2a we have to use a logarithmic scale, reducing its readability. The most relevant finding is that this metric is able to highlight extreme cases. However, as the comparison with Friedman-Nemenyi is performed in terms of wins and losses, this fact is not relevant from a statistical point of view. It may be only useful as an alternative visualization tool.

If we observe the Nemenyi output presented in Figure 5.2b, the conclusions drawn are almost the same. Only T50 and BL show a different behavior, which is

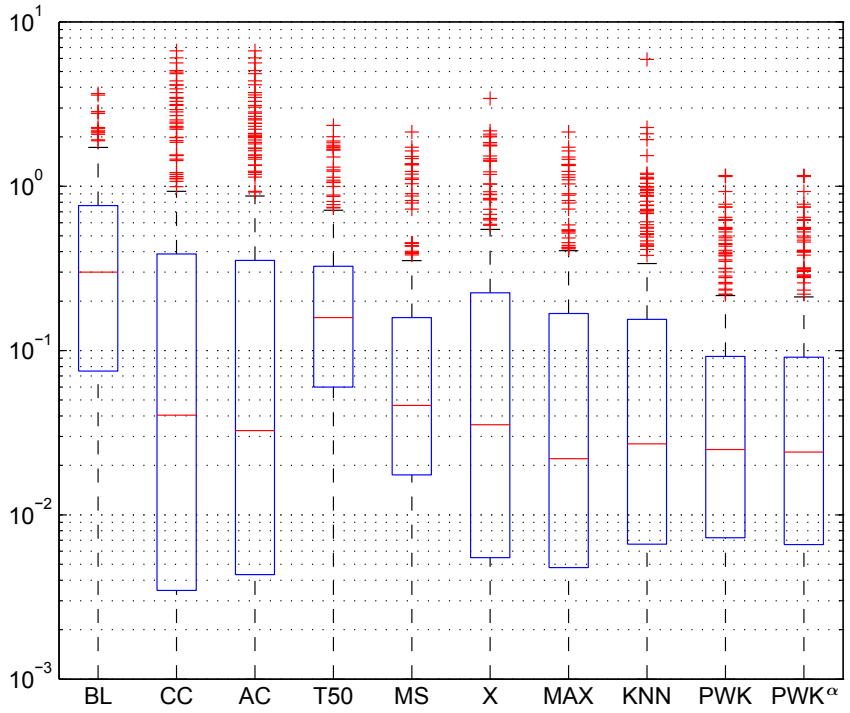


(a) Boxplots of all AE results

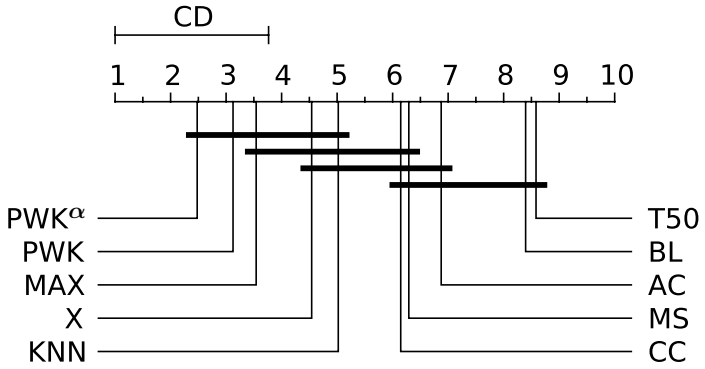


(b) Nemenyi at 5% (CD = 2.7654)

Figure 5.1: Statistical comparisons for NN experiments in terms of AE



(a) Boxplots of all *KLD* results



(b) Nemenyi at 5% (CD = 2.7654)

Figure 5.2: Statistical comparisons for NN experiments in terms of *KLD*

irrelevant from the perspective of the study. All other methods behave likewise, without benefiting any of the competing systems. This supports the analysis described in previous sections.

5.3.4 Pair-wise comparisons with PWK$^\alpha$

Since PWK$^\alpha$ appears to be the algorithm that yields the lowest values for AE in general, obtaining the best average rank in the Nemenyi test, from now on we shall use it as a *pivot* model so as to compare it to all the other systems under study. Thus, in Figure 5.3 we present pair-wise comparisons of each system with respect to PWK$^\alpha$. Each point represents the cross-validated AE values of the compared system on the y-axis and of PWK$^\alpha$ on the x-axis, for the same dataset and test prevalence. The red diagonal depicts the boundary where both systems perform equally. Therefore, when the points are located above the diagonal, PWK$^\alpha$ yields a lower AE value, and vice-versa. It should be noted that as we are using PWK$^\alpha$ as a pivot model for all comparisons, there is always the same number of points at each value of the x-axis. Thus, the movement of these points along the y-axis, among all the comparisons, provides visual evidence of which systems are more competitive with respect to PWK$^\alpha$.

We also include several metrics within each plot. The inner triplet shows the number of wins, losses and ties of PWK$^\alpha$ versus the compared system. The values below each plot reveal the difference between wins and losses (DWL), and within parentheses the mean of the differences between AE results of both algorithms ($MDAE$). Positive values of DWL and $MDAE$ indicate better results for PWK$^\alpha$, though they are only conceived for clarification purposes during visual interpretation. The aim of the DWL metric is to show the degree of *competitiveness* between two systems, values close to zero indicating that they are less differentiable, in terms of wins and losses, than systems with higher values. Moreover, $MDAE$ can also be used as a measure of the *symmetry* of both models. Note that being symmetric in this context does not refer to similarity of results, but to compensation of errors. This means that systems with an $MDAE$ value close to zero are less differentiable in terms of differences of errors.

From the shape drawn by the plots in Figure 5.3, we can observe some interesting interactions between models, always with respect to PWK$^\alpha$. As expected, the comparison with PWK, for example, shows a clear connection between both systems; all points present a strong trend towards the diagonal. Moreover, DWL indicates that PWK is the most competitive approach, while $MDAE$ shows that the average difference of errors is only 0.26, being highly symmetric.

The points in KNN's plot are not so close to the diagonal, being mainly situated slightly upwards. This behavior suggests that KNN is less competitive (78) and less symmetric (2.28) than PWK. Nevertheless, in general, NN-based algorithms present the best performance.

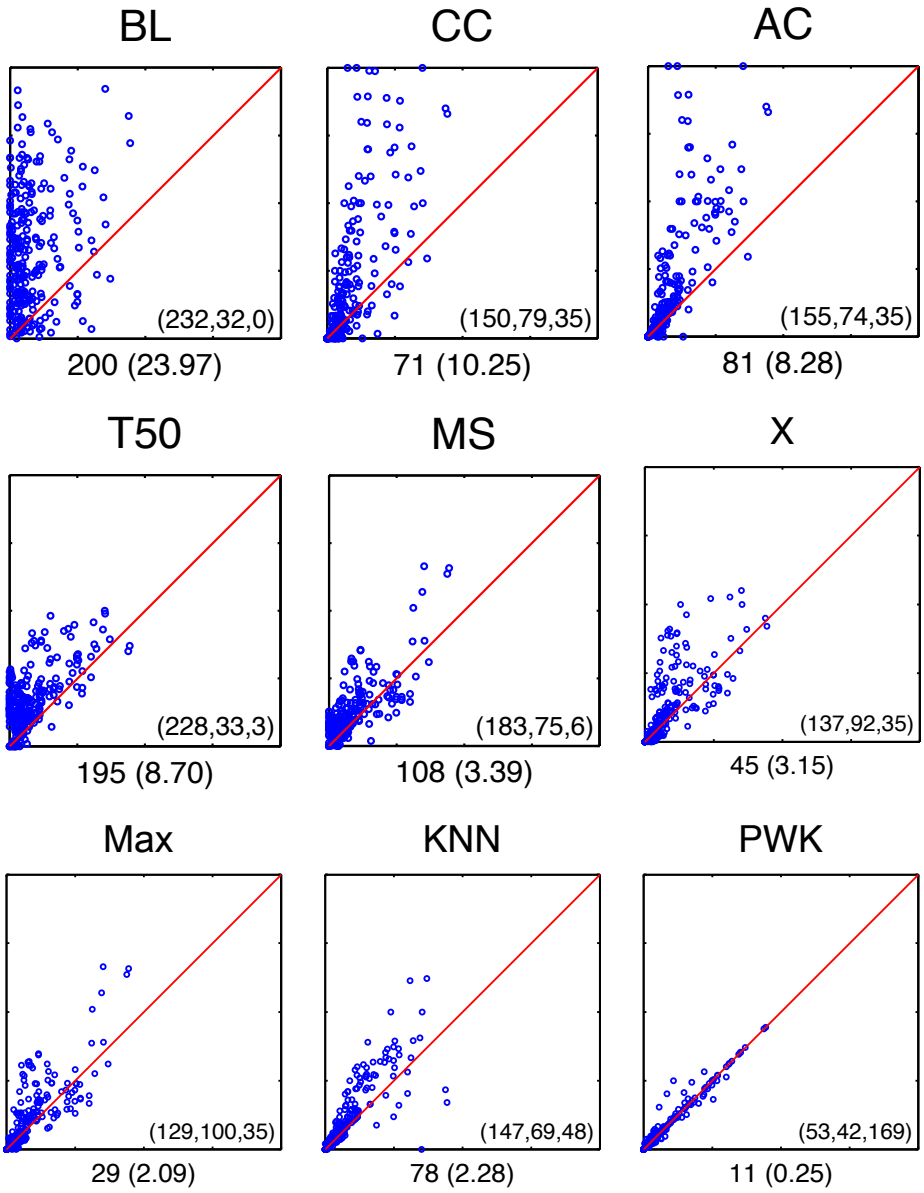


Figure 5.3: Pair-wise comparisons with respect to PWK$^{\alpha}$, in terms of AE. The results over all test prevalences are aggregated into a single plot, where each one represents 264 cross-validated results. The inner triplet shows the number of wins, losses and ties of PWK$^{\alpha}$ versus the compared system. The numbers below each plot reveal the difference between wins and losses (DWL), and within parentheses the mean of the differences between AE values ($MDAE$).

Although Max, X, MS and T50 are all based on threshold selection policies, the DWL and $MDAE$ values differ noticeably among them. As already observed in Figure 5.1b, Max seems to outperform the others, both in competitiveness (29) and symmetry (2.09), while T50 stands out as the less competitive approach among these quantification models.

The distribution of errors in Figure 5.1a for BL, CC and AC is once again evidenced in Figure 5.3. The presence of outliers in CC and AC is emphasized through high values of $MDAE$, combined with intermediate values of DWL. As regards BL, this algorithm shows the worst values in Figure 5.3 for competitiveness (200) and symmetry (23.97). This poor behavior can be also observed in Figure 5.1b.

5.3.5 Analysis of results by test prevalence

Although Figures 5.1a, 5.1b and 5.3 provide interesting evidence, they fail to show other important issues. For instance, we cannot properly analyze the performance of each system with respect to specific prevalences. Furthermore, they only offer a general overview of the limits and distribution of AE values, without taking into account the magnitude of the error with respect to the actual test proportions.

Figure 5.4 follows the same guidelines as those introduced for Figure 5.3; however, in this case we split each plot into eleven subplots, placed by rows. Each of these subplots represents the comparative results of a particular system with respect to PWK$^{\alpha}$ for a specific test prevalence. This decision is again supported by the fact that PWK$^{\alpha}$ appears to be the system that performs best in terms of AE metric. Moreover, despite the overload of information available, this summarization allows us to represent the values of all systems with fewer plots, to simplify the comparison of every system with respect to the best of our proposed models, and to visualize the degree of improvement among systems, all at the same time. The axes of those comparisons where DWL has negatives values are highlighted in red, while ties in DWL values are visualized by means of a gray axis. Notice that there are also cases where values of DWL and $MDAE$ have a different sign.

The average training prevalence among all datasets is 34.22%; hence, test prevalences at 30% and 40% are the closest to the original training distribution for the average case. This can be observed in Figure 5.4 through the BL results, which always predict the proportion observed during training. As expected, when the test distribution resembles that of the training, it yields competitive results, although the performance is significantly degraded to the worst case when the test proportions are different from those observed during training. Taking the plots of BL as reference, we observe that the behavior of PWK$^{\alpha}$ seems to be heading in the right direction in terms of both DWL and $MDAE$. Notice that the $MDAE$ values in this column rise and fall in keeping with changes in test prevalence.

The CC method performs well over low prevalence conditions, obtaining the best

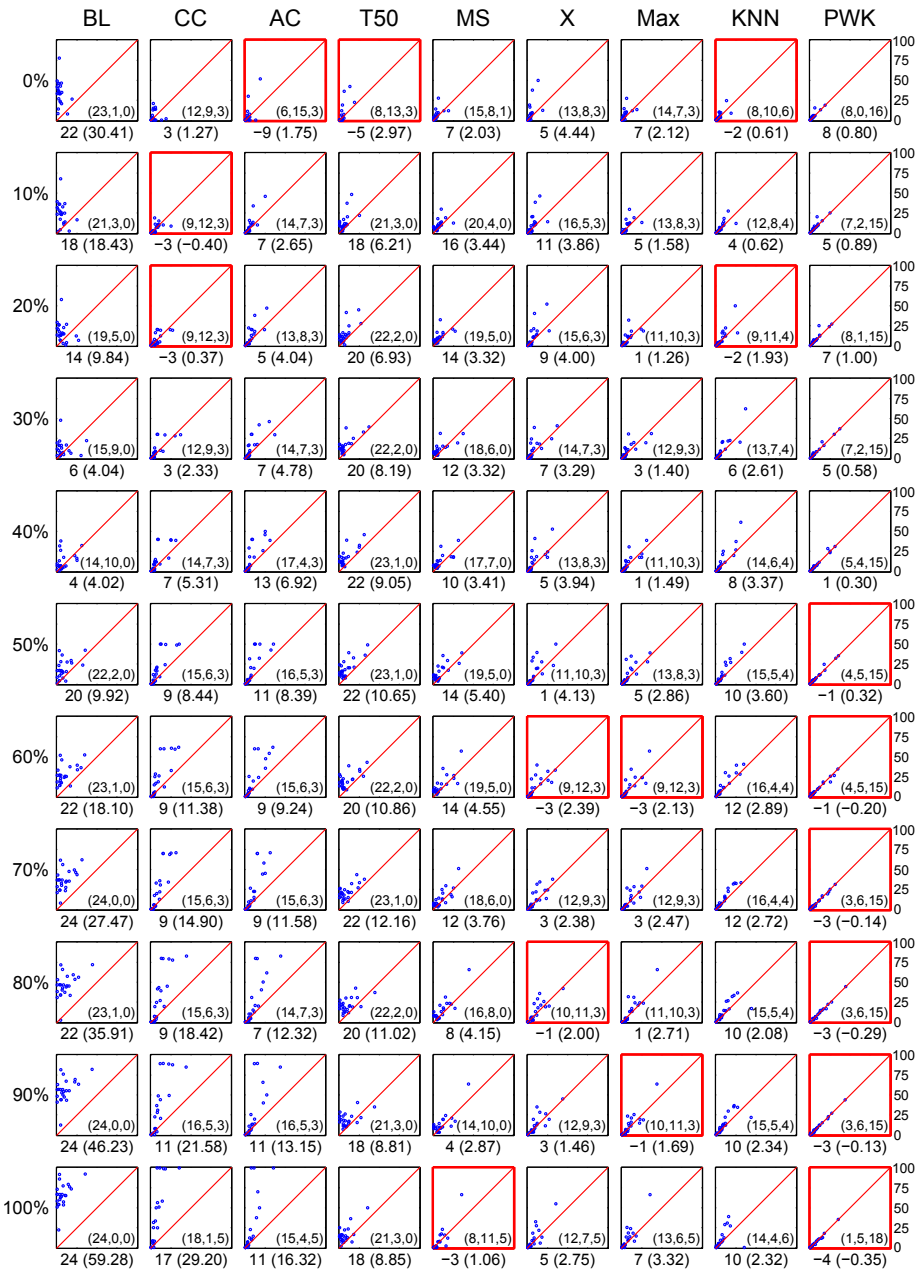


Figure 5.4: Pair-wise comparisons with respect to PWK$^{\alpha}$, in terms of AE. The results over different test prevalences are plotted individually (by rows), where each plot represents the cross-validated results over 24 datasets. See caption of Figure 5.3 for further details about the metrics placed below each graph.

DWL results for 10% and 20%. However, it apparently tends to increasingly underestimate for higher proportions of positives, as evidenced by the $MDAE$ values. This supports the conclusions regarding uncalibrated quantifiers drawn by Forman [2008]. On the other hand, we expected a more decisive improvement of AC over CC results in general. Actually, when the positive class becomes the majority class, for test prevalences greater than 50%, the AC correction produces an observable improvement in terms of DWL, and especially for $MDAE$. From a general point of view, however, the results that we have obtained with this experiment show that simply adjusting SVM outputs may not be sufficient, providing even worse results than traditional uncalibrated classifiers, mainly when testing low prevalence scenarios. This fact is mostly highlighted by the $MDAE$ results of CC and AC over prevalences below 50%.

The most promising results among state-of-the-art quantifiers are obtained by Max and X, although the former provides more competitive results for the average case. The greatest differences between $MDAE$ results are observed for test prevalences below 50%, where Max yields lower values. These differences are softened in favor of X for higher prevalences. We suspect that these threshold selection policies could entail an intrinsic compensation of the underlying classification bias shown by CC, which tends to overestimate the majority class. This intuition is supported by the observation that they still perform worse than CC for low test prevalences, as they may tend to overestimate the minority class. Additionally, both provide better DWL and $MDAE$ results than CC or AC for prevalences higher than 40%.

T50 presents the worst results of this family of algorithms, showing surprisingly good performance in test prevalence at 0%. Conversely, MS shows an intermediate behavior, performing appealingly in $MDAE$ but discouragingly in DWL, obtaining competitive results when the test prevalence is 100%. This good performance for extreme test prevalences could be due to the fact that corrected values are clipped into the feasible range after applying Equation (3.1), as described in Section 3.2. Therefore, this kind of behavior is not representative, unless it is reinforced with more stable results in near test prevalences. Moreover, Figures 5.1a, 5.3 and 5.4 highlight cases where Max and MS share some results. As described in Section 5.2.3, this is due to missing values in the latter method, which happens to be linked with outlier cases in Max. This suggests a possible connection between the complexity of these cases and their lack of thresholds where the denominator in (3.1) is big enough, being less robust with respect to estimation errors in tpr and fpr.

At first glance, KNN yields interesting results. Excluding CC, it improves DWL below 30% with respect to SVM-based models. Actually, both CC and KNN are the most competitive models over lowest prevalences, probably because they tend to misclassify the minority class, so that they are biased to overestimate the majority class. Thus, when the minority class shrinks, the quantification error also decreases. Notwithstanding, KNN behaves more consistently, providing stable

$MDAE$ results over higher prevalences. Comparing KNN with AC, we also observe that, in general, KNN also appears to be more robust in terms of $MDAE$. This suggests that KNN produces AE results with lower variance and less outliers than CC and AC, as previously observed in Figures 5.1a and 5.3.

As already mentioned, the red (black) color in Figure 5.4 represent cases where the compared system yields better (worse) DWL than PWK$^\alpha$, while ties are depicted in gray. Hence, these plots reinforce the conclusion that PWK$^\alpha$ is usually the algorithm that performs best, with a noticeable dominance in terms of $MDAE$. Apparently, adding relatively simple weights offers an appreciable improvement, which is clearly observable when compared with traditional KNN. With the exception of PWK, there exists only one case where both DWL and $MDAE$ produce negative values in Figure 5.4, corresponding to CC at a test prevalence of 10%. This is probably caused by the fact that CC is supposed to yield exact results over a specific prevalence, identified as p^* in Forman's theorem. Therefore, this result is not relevant in terms of global behavior. Furthermore, except for PWK over prevalences higher than 50%, the values for the $MDAE$ metric are positive in all cases. This implies that AE values provided by PWK$^\alpha$ and PWK are generally lower and have less variance than those of all the other systems.

The resemblance between PWK$^\alpha$ and PWK is once again emphasized through low values of $MDAE$ over all test prevalences. However, previous figures failed to shed light on a very important issue. Observing the last column in Figure 5.4, it appears that PWK$^\alpha$ is more conservative and robust over lower prevalences, while PWK is more competitive over higher ones. These differences are softened towards intermediate prevalences. This behavior is supported by the fact that, although PWK$^\alpha$ and PWK use weights based on equivalent formulations, the parameter α in PWK$^\alpha$ tends to weaken the influence of these weights when it increases. Moreover, as already stated in Section 5.1.2, since these weights are designed to compensate the bias towards the majority class, when the parameter α grows, the recall decreases, and vice-versa.

Finally, in order to bring the analysis of the experimental results to an end, Figure 5.5 shows the raw AE scores for each dataset and each test prevalence. We have only included four representative algorithms, being AC, Max, KNN and PWK$^\alpha$. The aim of this figure is to enable us to check graphically whether it does exist any correlation between the original training prevalence and the performance of the systems under study, or not. Each plot depicts the results of all four systems for every test prevalence, sorting the plots by training prevalence. Observing Figure 5.5 we cannot conclude that the training prevalence influences the quantification performance, it seems that the intrinsic complexity of each dataset is more relevant.

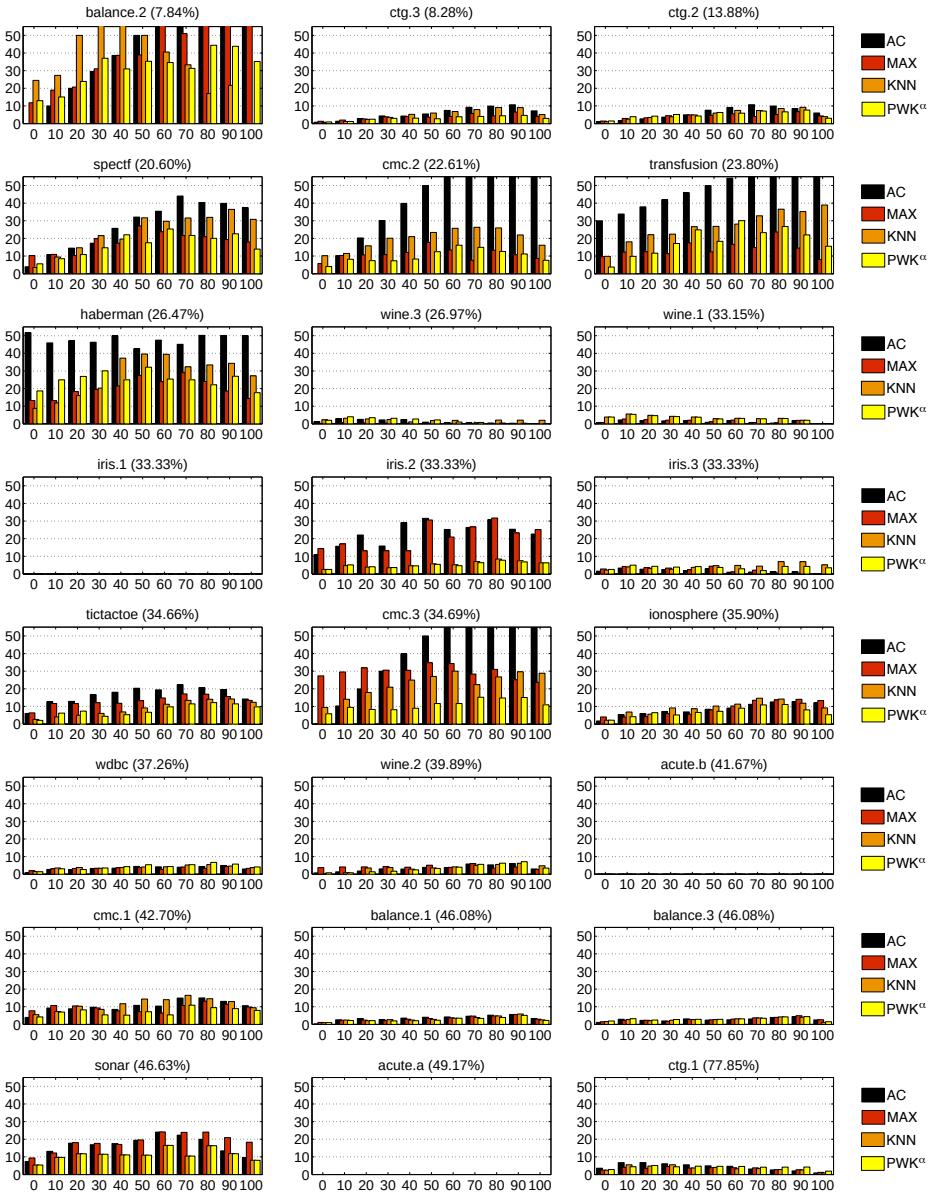


Figure 5.5: Bar plots of *AE* results for AC, Max, KNN and PWK$^{\alpha}$ over all the datasets. Each bar plot depicts the results of all four systems, for every test prevalence. The titles show the acronym of the dataset and its original prevalence, sorting the plots by training prevalence from left to right and from top to bottom. The y-axis range is fixed from 0% to 50% for comparison purposes.

Chapter 6

Design and optimization of quantification loss functions

State-of-the-art quantification models based on classifiers present the drawback of being trained with a loss function aimed at classification, rather than quantification. Other recent attempts to address this issue suffer some limitations regarding reliability, measured in terms of classification abilities. Thus, in this chapter we present a learning method that optimizes an alternative metric that combines quantification and classification performance simultaneously. Our proposal offers a new framework that allows constructing binary quantifiers that are able to accurately estimate the proportion of positives, based on models with reliable classification abilities (high sensitivity). This study is currently under review on *Pattern Recognition* [Barranquero et al., under review].

The two major frameworks described in Chapter 3 may present some disadvantages under specific conditions, as it happens with all learning paradigms. On the one hand, Forman's methods [Forman, 2008] provide estimations that are obtained in terms of modified classification models, optimized to improve their classification accuracy, instead of training them to reduce their quantification error. Although these algorithms showed promising quantification performance in practice, it seems more orthodox to build quantifiers optimizing a quantification metric, as stated by Esuli and Sebastiani [2010].

However, on the other hand, their proposal does not take into account the classification accuracy, as long as the quantifier balances the number of errors between both classes, even at the cost of obtaining a rather poor classifier. That is, Esuli and Sebastiani propose that the learning method should optimize a quantification measure that simply deteriorate with $|FP - FN|$. We strongly believe that it is also important that the learner considers the classification performance as well. Our claim is that this aspect is crucial to guarantee a minimum level of confidence for deployed models. The key problem is that pure quantification measures do not take into account the classification abilities of the model, producing several optimum points within the hypothesis search space (any that fulfills $FP = FN$); with some of these hypotheses being less reliable than others.

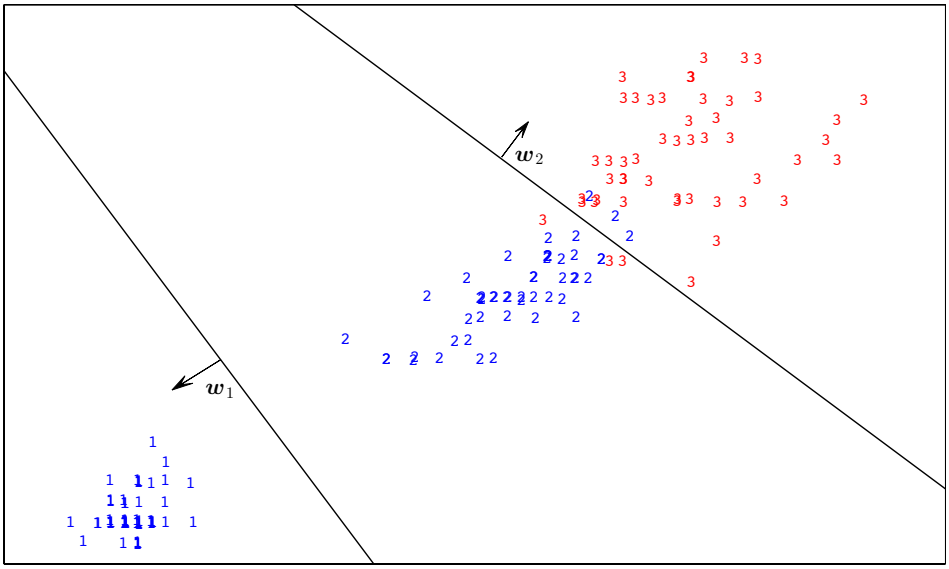


Figure 6.1: Graphical representation of two conflicting perfect quantifiers.

6.1 Balancing quantification and classification

In order to analyze this issue we will use the example in Figure 6.1, which represents all instances of *iris* dataset. This training set contains three classes, with the same percentage for each of them. The learning task is to obtain a quantifier for class 3, that is, class 3 is the positive class, while the negative class is composed by classes 1 and 2. The figure depicts two hypotheses: $\boldsymbol{w}_1$ and $\boldsymbol{w}_2$; the former classifies all examples of class 1 as positives, while the latter predicts the majority of examples of class 3 as positives. Both hypotheses are perfect quantifiers according to training data. Thus, any learning method that only takes into account the quantification performance is not able to distinguish between them. In practice, it will end up choosing one depending on the particular learning bias of its underlying algorithm.

Our claim is that a reliable learner should prefer $\boldsymbol{w}_2$, because it offers better classification abilities, being more robust to changes in class distribution. Actually, $\boldsymbol{w}_1$ will quantify any change in the proportion of class 3 on the opposite direction, due to the fact that the hyperplane defined by $\boldsymbol{w}_1$ is irrelevant to the positive/negative distinction. That is, using $\boldsymbol{w}_1$, any increment in the proportion of class 3 results in a decrement of the quantification of that class, and vice-versa. Conversely, the estimations of $\boldsymbol{w}_2$ increases or decreases in the same direction of these changes.

Table 6.1: Perfect quantifier with worst possible *recall*

	P	N
P'	$TP = 0$	$FP = P$
N'	$FN = P$	$TN = N - P$

As matter of fact, the hyperplane defined by $\boldsymbol{w}_1$ happens to be the worst case in terms of classification performance. This is because even been a perfect quantifier, it misclassifies all the examples of the positive class, counteracting them with the same amount of false positive errors. Table 6.1 shows the contingency table for this hypothesis. Obviously, this is an undesired extreme case, which nonetheless current algorithms have not addressed properly for the time being.

In summary, Forman's methods are designed to build and characterize classifiers in order to apply them as quantifiers. On the other hand, the proposal presented by Esuli and Sebastiani emphasizes quantification ability during optimization; although they do not implement nor evaluate it. This chapter explores this second alternative in detail, presenting also a new proposal that may be able to soften some drawbacks of previous methods, considering both classification and quantification performance during learning, and thus producing more reliable and robust quantifiers.

In fact, given that confidence is always a key issue for the application of machine learning methods in practice, the open question is how to measure the reliability that offers a quantifier, or whether it is reasonable that it was not able to classify correctly a minimum number of examples. We consider that this is the main potential pitfall of the proposal presented by Esuli and Sebastiani, because it is based on learning a binary classifiers that simply compensates the errors between both classes, even when the underlying classifier shows a rather poor classification performance.

The key problem that arises when optimizing a pure quantification measure is that the resulting hypothesis space contains several global optimums. However, as we have analyzed before, those optimum hypotheses are not equally good because they differ in terms of the quality of their future quantification predictions (see Table 6.1 and Figure 6.1). Our claim is that the robustness of a quantifier based on an underlying classifier is directly related to the reliability of such classifier. For instance, given several models showing equivalent quantification performance during training, the learning method should prefer the best one in terms of its potential of generalization.

The formal approach to obtain such quantifiers is to design a metric that combines somehow classification and quantification abilities, and then applying a learning algorithm able to select a model that optimizes such metric. This is the core idea of the proposal presented in this chapter.

Conceptually, the strategy of merging two complementary learning objectives is not new; we find the best example in information retrieval. The systems developed for these tasks are trained to balance two goals, retrieving as many relevant documents as possible, but discarding non-relevant ones. The metric that allows assessing how close are these complementary goals of being accomplished is *F-measure* [van Rijsbergen, 1979, 1974]. Actually, this metric emerges from the combination of two ratios: *recall* (TP/P), which is already defined as *tpr* in Equation (2.7), and *precision* (TP/P'). In some sense, we face a similar problem in quantification.

6.2 *Q-measure*: a new parametric loss function

All previous discussion lead us to present a new metric, termed *Q-measure*, which balances quantification and classification performance simultaneously. The first point to emphasize is that quantification is mostly explored for binary problems, in which the positive class is usually more relevant and must be correctly quantified. Thus, the design of *Q-measure* is focused on a binary quantification setting.

The standard classification metric *F-measure* is defined as

$$F_\beta = (1 + \beta^2) \cdot \frac{precision \cdot recall}{\beta^2 \cdot precision + recall}, \tag{6.1}$$

which balances an adjustable tradeoff between *precision* and *recall*. Analogously, we suggest *Q-measure*, defined as

$$Q_\beta = (1 + \beta^2) \cdot \frac{cperf \cdot qperf}{\beta^2 \cdot cperf + qperf}. \tag{6.2}$$

The β parameter allows weighting *cperf* and *qperf* measures, providing an AND-like behavior. Note that *cperf* and *qperf* stand for *classification performance* and *quantification performance* respectively. The selection of these metrics depends on the final learning goal, keeping in mind that they should be bounded between 0 and 1 in order to be effectively combined, representing worst and best case respectively.

Therefore, the first element of our proposal is a new family of score functions, inspired in the aforementioned *F-measure*. We need two core ingredients, a metric for quantification and another for classification. The additional advantage of this approach is flexibility, in terms that almost any combination of measures can be potentially selected by practitioners. This new family is mainly aimed at guiding model selection during the learning stage. But, up to some extent, it also allows comparing quantifiers trained with different approaches, whether or not they are based on these ideas. Evaluating quantifiers from this two-view perspective provides us the possibility of analyzing their reliability.

We now explore some alternatives through graphical representations. The motivation behind Figures 6.2–6.7 is to enable us to analyze the behavior of different loss functions with respect to all combinations of values for FP and FN; both under balanced (6.2, 6.4 and 6.6) and unbalanced (6.3, 6.5 and 6.7) training conditions. Each of the 2D plots is the xy-projection of its lower 3D graphic. Darker colors mean better scores. Notice also that 3D views are rotated over z-axis in order to ease the visualization of the surfaces and that x-axis range is different between balanced and unbalanced cases. Intuitively, a well-conceived learning procedure should tend to move towards those models whose scores are within darkest areas. In other words, these graphs illustrate the hypothesis search space of each metric.

6.2.1 Classification performance

In Figures 6.2 and 6.3 we look over some candidate classification metrics. Following with the binary quantification setting introduced before, a natural choice for *cperf* is *accuracy*, defined as $(TP + TN)/S$. However, it has some drawbacks because quantification is usually applied over an unbalanced binary problem, in which negatives are the majority class, resulting from a combination of several related classes (*one-vs-all*).

Other standard alternatives are F_1, defined in Equation (6.1), or the geometric mean of *tpr* (*recall*) and *tnr* (*true negative rate*), defined as

$$GM = \sqrt{\frac{TP}{P} \cdot \frac{TN}{N}}; \tag{6.3}$$

i.e., the geometric mean of *sensitivity* and *specificity*. This measure is particularly useful when dealing with unbalanced problems in order to alleviate the bias towards the majority class during learning [Barandela et al., 2003].

An interesting property of both *tpr* and *tnr* is that their corresponding search space is only defined over one of the two classes, and then they are invariant to changes in the dimension of the other. Notice that the graphical representation of *tnr* is equivalent to *tpr* (*recall*) in Figures 6.2 and 6.3, but rotated 90° over z-axis. That is why GM also shows a constant shape between balanced (Figure 6.2) and unbalanced cases (Figure 6.3), with a proper scaling for y-axis. It is also worth noting that *accuracy* approximate to *tnr* when the size of positive class is negligible

$$\frac{(TP + TN)}{S} \approx \frac{TN}{N}, \text{ when } P \to 0.$$

Therefore, we consider that *accuracy* could be appropriate only in those cases where we were dealing with problems where both classes had similar size, so we discard it for the general case. Regarding, F_1 and GM, although both could

be appropriate, we finally focus on *recall* for our study. A potential benefit of maximizing *recall* is motivated by the fact that this may lead to a greater denominator in Equation (3.1), providing more stable corrections. It is also interesting that this metric is included in *F-measure* and *GM*, in order to weight the relevance of the positive class accordingly. Thus, this decision is also supported by the fact that the goal of the applications described in quantification literature is focused on estimating the prevalence of the positive class, which is usually more relevant.

In practical terms, *Q-measure* is able to discard pointless *qperf* optimums thanks to the use of *recall*. The key idea is that *recall* acts as a *hook*, forcing the quantifier to avoid incoherent classification predictions over the positive class. This reduces the amount of FN errors, restricting in turn the search space for the quantification part in *Q-measure*. Notice also that pure quantification metrics tends to overlook positive class relevance in unbalanced scenarios.

6.2.2 Quantification performance

We have considered several alternatives for *qperf*, starting from standard measures described in Section 2.2. Unfortunately, none of the reviewed metrics fulfill all the requirements imposed by the design of *Q-measure*. Hence, we also analyze the normalized versions of AE and SE. Figures 6.4 and 6.5 provide a graphical representation to assist in interpretation and discussion of these functions. Anyhow, it is worth mentioning that the decision about *qperf* does not depend on whether we need to estimate the prevalence of one or both classes, because in binary problems both values are complementary ($p = 1 - n$, where n is the proportion of negatives or N/S).

Estimation bias, defined in Section 2.2.1, is clearly out of scope because it can yield negative predictions. We also discard KLD because it is not properly bounded and it yields unwieldy results when estimated proportions are near 0% or 100%, like infinity or indeterminate values. According to Forman [2008], this problem can be fixed by backing off by half a count (see Section 2.2.3). Moreover, as can be observed in Figures 6.4 and 6.5, we also have to crop its range after subtracting from 1. These adjustments are not exempt from controversies, so we have focused on other alternatives.

We consider that AE and SE, defined in Section 2.2.2, are the most appropriate candidates because both are bounded between 0 and 1. However, they do not reach value 1 for almost all possible class proportions, except for $p \in \{0, 1\}$, getting further away from 1 in correlation with the degree of imbalance (notice that in Figures 6.4 and 6.5 AE and SE values are subtracted from 1). This may produce an awkward behavior when combining these metrics with *cperf* in Equation (6.2). Observe also that in Figures 6.2 and 6.3 both components of *F-measure* cover the whole range between 0 (worst) and 1 (best case), and so does require *Q-measure*.

Looking at Equation (2.4) and (2.5) in more detail, we observe that given a particular value for p their effective upper bounds are $\max(p, n)$ and $\max(p, n)^2$ respectively. Thus, we need to normalize them. Moreover, as they are defined as loss functions, with optimum at 0, we also need to redefine them as score functions. Taking into account these factors, it is straightforward to obtain two derived measures for quantification, denoted as *Normalized Absolute Score* (*NAS*)

$$NAS = 1 - \frac{|p' - p|}{\max(p, n)} = 1 - \frac{|FN - FP|}{\max(P, N)}, \tag{6.4}$$

and *Nomalized Squared Score* (*NSS*)

$$NSS = 1 - \left(\frac{p' - p}{\max(p, n)}\right)^2 = 1 - \left(\frac{FN - FP}{\max(P, N)}\right)^2. \tag{6.5}$$

Figures 6.4 and 6.5 show that *NAS* and *NSS* are uniform and easily interpretable, following equivalent shapes to those offered by standard quantification loss functions. For instance, *NSS* is quite similar to 1-*KLD*. From Figure 6.4, we can observe that when the problem is balanced, then all functions returns the best scores on the diagonal. This represents where FP and FN values neutralize each other, i.e., where $|FP - FN|$ cancels. On the other hand, Figure 6.5 provide an example of an unbalanced problem. The optimal region lies again over the line where these values compensate each other, as it may be expected.

For the sake of simplicity, we only focus on *NAS* in our study. If we look for the maximum possible value of $|FP-FN|$, we conclude that it is always the number of individuals of the majority class. Assuming that N is greater than P, as it is usual, the proof is that the worst quantification score is achieved when all the examples of the minority class are classified correctly ($TP = P$ and $FN = 0$), but all the examples of the majority class are misclassified ($TN = 0$ and $FP = N$), and thus Equation (6.4) evaluates to 0. With such a simple metric, we can observe that the $|FP - FN|$ count is weighted in terms of the predominant class (denominator), forcing the output on the whole range between 0 and 1.

6.2.3 Graphical analysis of *Q-measure*

The graphical representation in Figures 6.6 and 6.7 provides an intuitive view to understand the behavior of *Q-measure*, selecting *recall* as *cperf* and *NAS* as *qperf* for Equation (6.2). Its interpretation is exactly the same as in previous figures. Again, we present two alternative learning conditions: balanced (Figure 6.6) and unbalanced (Figure 6.7). For each of them, from left to right, we show different search spaces obtained from five target measures: first *NAS*, then those obtained from three different β values (Q_2, Q_1 and $Q_{0.5}$), and finally *recall*. Notice that *recall* and *NAS* are equivalent to Q_0 and Q_∞ respectively. When the value of β

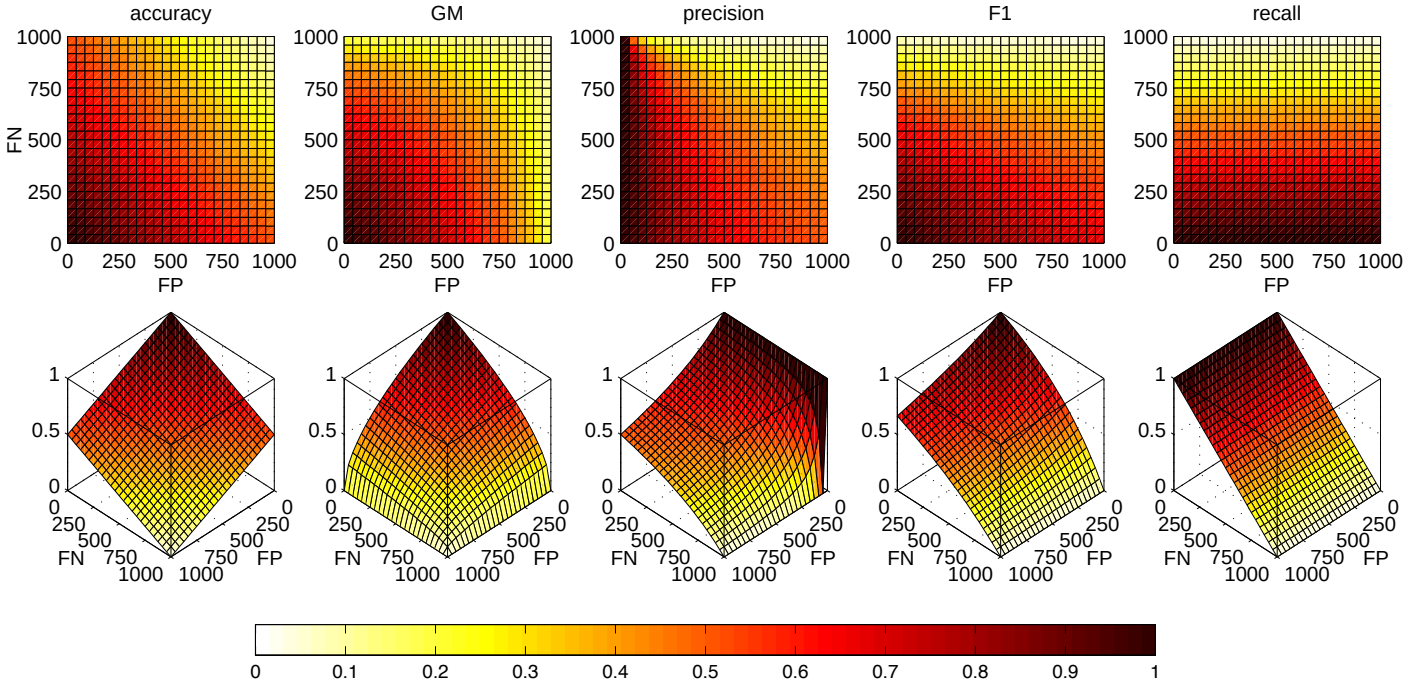


Figure 6.2: Graphical representation of all possible values for different classification loss functions, varying FP and FN between 0 and their maximum value. Balanced case with 1000 examples of each class (P = 1000, N = 1000). Darker colors mean better scores.

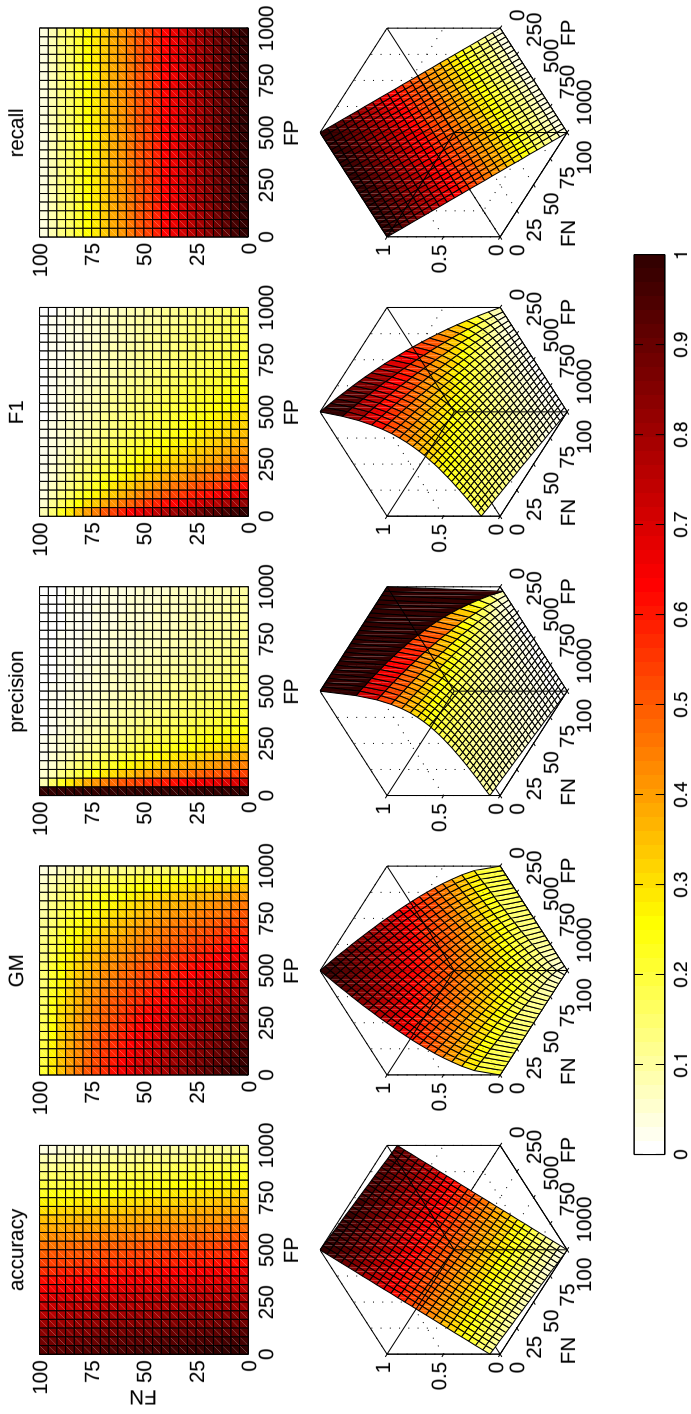


Figure 6.3: Graphical representation of all possible values for different classification loss functions, varying FP and FN between 0 and their maximum value. Unbalanced case (9%) with 1100 examples (P = 100, N = 1000). Darker colors mean better scores.

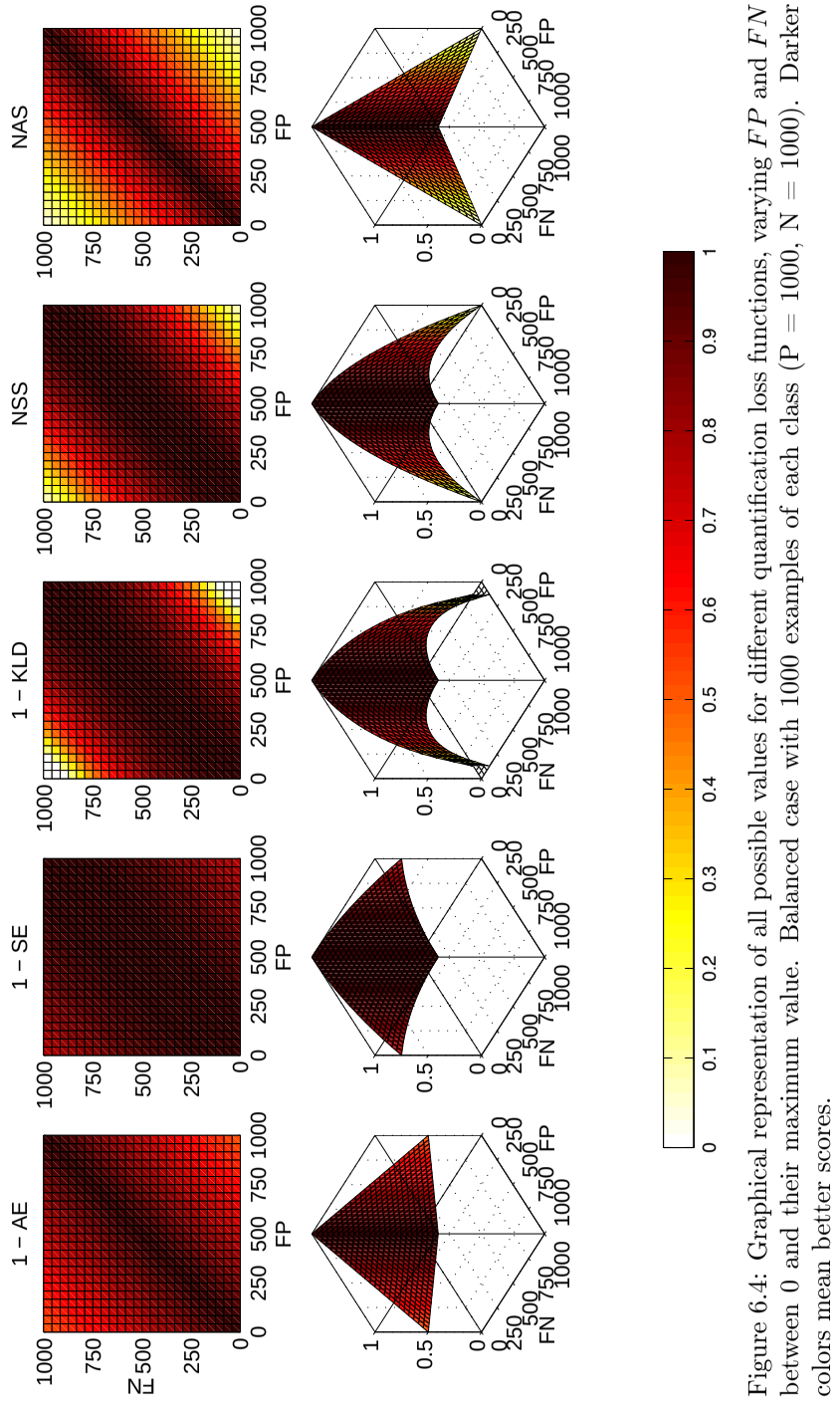


Figure 6.4: Graphical representation of all possible values for different quantification loss functions, varying FP and FN between 0 and their maximum value. Balanced case with 1000 examples of each class (P = 1000, N = 1000). Darker colors mean better scores.

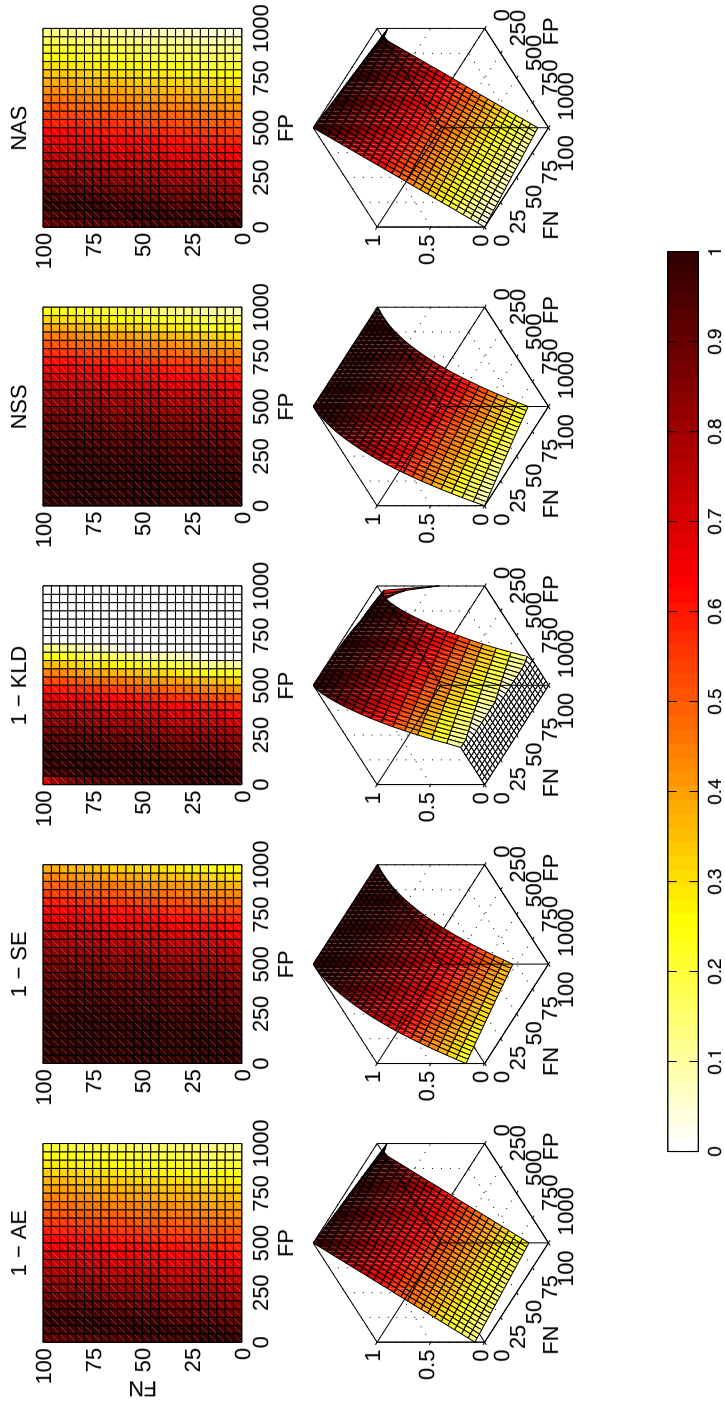


Figure 6.5: Graphical representation of all possible values for different quantification loss functions, varying FP and FN between 0 and their maximum value. Unbalanced case (9%) with 1100 examples (P = 100, N = 1000). Darker colors mean better scores.

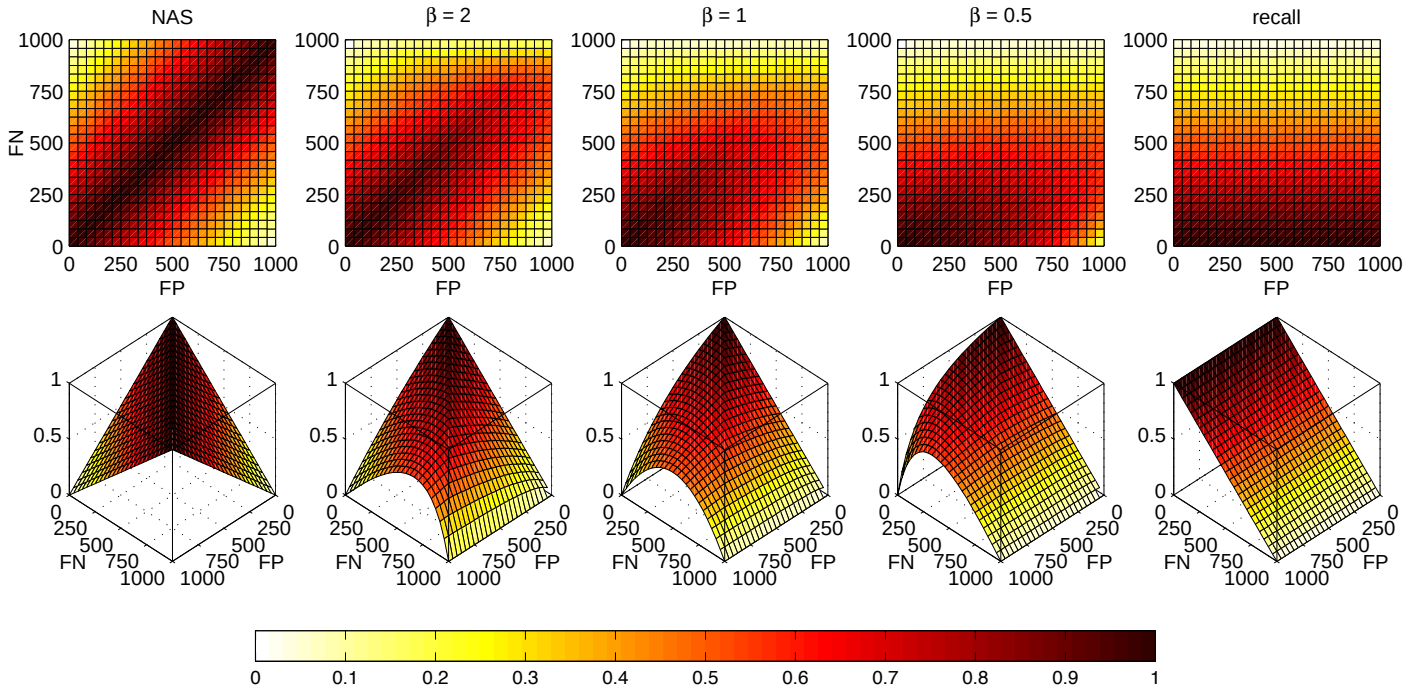


Figure 6.6: Graphical representation of all possible values for our proposed loss function Q_β, varying FP and FN between 0 and their maximum value. Each row shows the progression from *NAS* ($\beta \to \infty$) to *recall* ($\beta = 0$) through different values of β. Balanced case with 1000 examples of each class (P = 1000, N = 1000). Darker colors mean better scores.

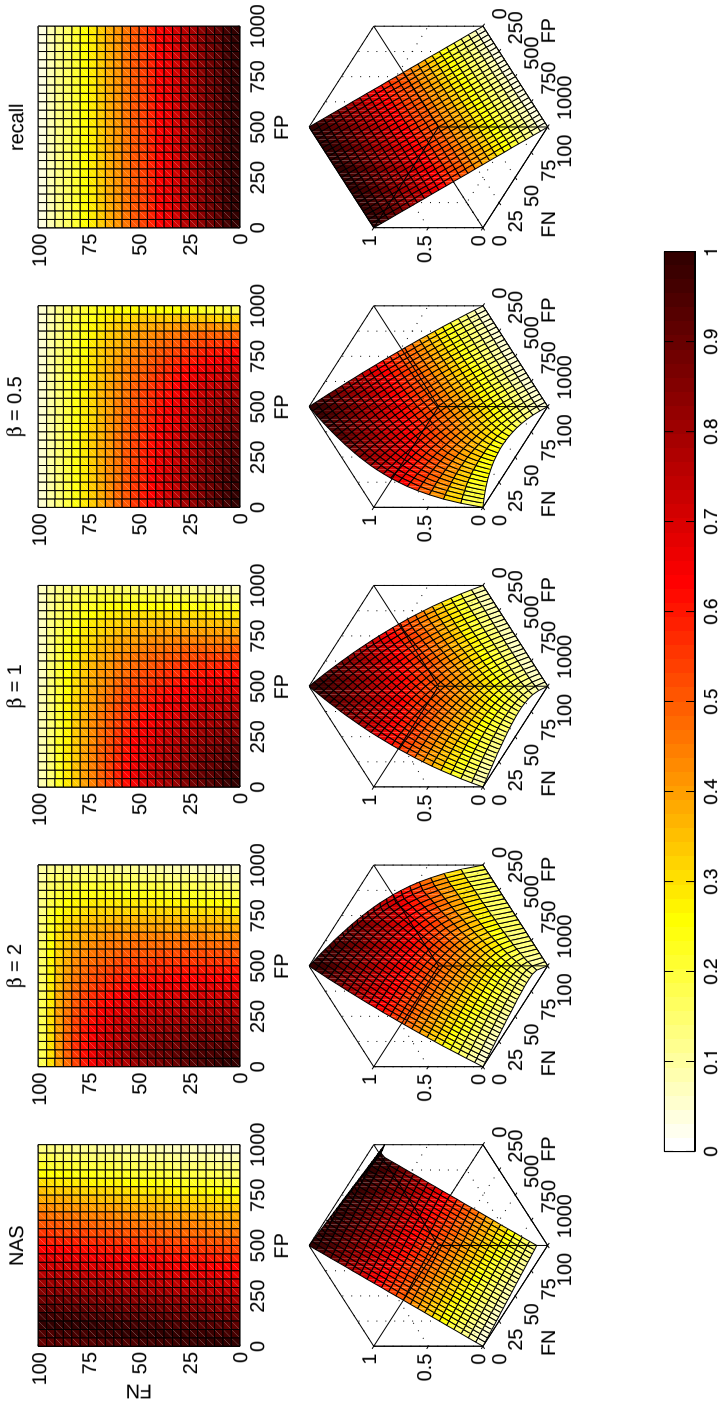


Figure 6.7: Graphical representation of all possible values for our proposed loss function Q_β, varying FP and FN between 0 and their maximum value. Each row shows the progression from NAS ($\beta \to \infty$) to *recall* ($\beta = 0$) through different values of β. Unbalanced case (9%) with 1100 examples (P = 100, N = 1000). Darker colors mean better scores.

is 1 (on the middle graphic), both classification and quantification performance measures are equally weighted; when its value decreases to 0, then *Q-measure* tends to be more similar to *cperf*, and when it rises from 1, it tends to resemble *qperf*. Obviously, for the intermediate values of β, the obtained search spaces are significantly different from that of the seminal metrics.

In summary, *recall* drives the model to yield accurate predictions over the positive class, minimizing FN. While, on the other hand, NAS evaluates the compensation between FP and FN. Hence, we have that *Q-measure* degrades when $|FP - FN|$ is high, but we are also penalizing those models with high FN.

Figures 6.6 and 6.7 suggest that the search space defined by $\beta = 2$ may be able to produce competitive quantifiers. An interesting property of this learning objective is that Q_2 preserves the general shape of the optimal region defined by NAS, while degrading these optimums in consonance with *recall*. That is, it offers the benefits of a quantification-oriented target, avoiding incoherent optimums (see Section 6.1).

We can also observe that with $\beta = 1$ we are forcing the learning method to obtain models in the proximities of the lower values of FP and FN. Specifically, in Figure 6.3 and Figure 6.7 we see that the shape of Q_1 reminds that of GM when the dataset is unbalanced. This similarity is originated by the fact that both share *recall* as one of their components, while NAS is similar to *tnr* on highly unbalanced datasets. On the extreme case, when positive class is minimal, the score $1 - AE$ approximates to NAS, *accuracy* and *tnr*:

$$1 - \frac{|FP - FN|}{N + P} \approx 1 - \frac{FP}{N} = \frac{TN}{N} \approx \frac{(TP + TN)}{N + P}, \text{ when } P \to 0.$$

Therefore, the main motivation for mixing in *recall* is that the alternative of using only a pure quantification metric could imply optimizing a similar target to that of *accuracy* or *tnr* on highly unbalanced problems. In fact, as we will analyze in the following section, the empirical results obtained from our experiments suggest that the behavior of a model learned though NAS is very similar to that of CC, which is a classifier trained with *accuracy*. On balanced cases, we believe that the contribution of *recall* to *Q-measure* also offers a more coherent learning objective, providing more robust quantifiers in practice.

6.3 Multivariate learning hypothesis

The main challenge of our proposed *Q-measure* is that not all binary learners are capable of optimizing this kind of metrics, because such functions are not decomposable as a linear combination of the individual errors. Hence, this approach requires a more advanced learning machine, like SVM^{Δ}_{multi} [Joachims, 2005], which provides an efficient base algorithm for optimizing non-linear

functions computed from the contingency table (see Table 2.1). Nevertheless, the straightforward benefit is that these methods address the quantification problem from an aggregated perspective, taking into account the performance over whole samples, which seems more appropriate for the problem.

Therefore, rather than learning a traditional classification model like

$$h : \mathcal{X} \rightarrow \mathcal{Y},$$

the core idea of SVM^{Δ}_{multi} is to transform the learning problem into a multivariate prediction one. That is, the goal is to induce a hypothesis $\bar{h}$ that maps all feature vectors of a sample $\bar{\boldsymbol{x}} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_S)$, to a tuple $\bar{y} = (y_1, \ldots, y_S)$ of S labels

$$\bar{h} : \bar{\mathcal{X}} \rightarrow \bar{\mathcal{Y}},$$

in which $\bar{\boldsymbol{x}} \in \bar{\mathcal{X}} = \mathcal{X}^S$ and $\bar{y} \in \bar{\mathcal{Y}} = \{-1, +1\}^S$. This multivariate mapping is implemented with a linear discriminant function

$$\bar{h}_{\boldsymbol{w}}(\bar{\boldsymbol{x}}) : \arg\max_{\bar{y}' \in \bar{\mathcal{Y}}} \left\{ \langle \boldsymbol{w}, \Psi(\bar{\boldsymbol{x}}, \bar{y}') \rangle \right\},$$

where $\bar{h}_{\boldsymbol{w}}(\bar{\boldsymbol{x}})$ yields the tuple $\bar{y}' = (y'_1, \ldots, y'_S)$ of S predicted labels with higher score according to the linear function defined by the parameter vector $\boldsymbol{w}$. The joint feature map Ψ describes the match between a tuple of inputs and a tuple of outputs. For the quantification-oriented methods presented in this dissertation, we use the same form proposed by Joachims for binary classification

$$\Psi(\bar{\boldsymbol{x}}, \bar{y}') = \sum_{i=1}^{S} \boldsymbol{x}_i y'_i.$$

This setup allows the learner to consider the predictions for all the examples, and optimize in turn a sample-based loss function Δ. The optimization problem for obtaining $\boldsymbol{w}$ given a non-negative Δ is as follows

$$\begin{aligned} \min_{\boldsymbol{w}, \xi \geq 0} \quad & \frac{1}{2}\langle \boldsymbol{w}, \boldsymbol{w} \rangle + C\xi \\ \text{s.t.} \quad & \langle \boldsymbol{w}, \Psi(\bar{\boldsymbol{x}}, \bar{y}) - \Psi(\bar{\boldsymbol{x}}, \bar{y}') \rangle \geq \Delta(\bar{y}', \bar{y}) - \xi, \quad \forall \bar{y}' \in \bar{\mathcal{Y}} \setminus \bar{y}. \end{aligned} \tag{6.6}$$

Notice that the constraint set of this optimization problem is extremely large, including one constraint for each tuple $\bar{y}'$. Solving this problem directly is intractable due to the exponential size of $\bar{\mathcal{Y}}$. Instead, we obtain an approximate solution applying Algorithm 2 [Joachims, 2005], which is the sparse approximation algorithm of [Tsochantaridis et al., 2005, 2004] adapted to the multivariate SVM^{Δ}_{multi}.

The key idea of this algorithm is to iteratively construct a *sufficient subset* of the set of constraints CS. In each iteration, the most violated constraint is added

Algorithm 2 Algorithm for solving quadratic program of multivariate $\text{SVM}^{\Delta}_{multi}$

Input: $\bar{\boldsymbol{x}} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n)$ and $\bar{y} = (y_1, \ldots, y_n)$, ϵ
$CS \leftarrow \varnothing$
$\xi \leftarrow 0$
repeat
  $\bar{y}' \leftarrow \arg\max_{\bar{y}' \in \bar{Y}} \{\Delta(\bar{y}', \bar{y}) + \langle \boldsymbol{w}, \Psi(\bar{\boldsymbol{x}}, \bar{y}')\rangle\}$
  $\xi' \leftarrow \Delta(\bar{y}', \bar{y}) - \langle \boldsymbol{w}, \Psi(\bar{\boldsymbol{x}}, \bar{y}) - \Psi(\bar{\boldsymbol{x}}, \bar{y}')\rangle$
  if $\xi' \geq \xi + \epsilon$ **then**
    $CS \leftarrow CS \cup \{\bar{y}'\}$
    $\boldsymbol{w}, \xi \leftarrow$ optimize $\text{SVM}^{\Delta}_{multi}$ objective over CS
  end if
until CS has not changed during last iteration
Return: $\boldsymbol{w}$

to the active subset of constraints, i.e., the constraint corresponding to the label that maximizes $H(\bar{y}) = \{\Delta(\bar{y}', \bar{y}) + \langle \boldsymbol{w}, \Psi(\bar{\boldsymbol{x}}, \bar{y}')\rangle\}$. Obviously, the search of this constraint depends on the target loss function. After adding each new constraint, the next approximation to the solution of optimization problem (6.6) is computed on the new set of constraints. The algorithm stops when no constraint is violated by more than ξ.

This method guarantees that the solution returned fulfills all constraints up to precision ξ, while the norm of $\boldsymbol{w}$ is no bigger than the norm of the exact solution of (6.6). Furthermore, Tsochantaridis et al. [2004] demonstrate that this algorithm terminates after a polynomial number of iterations, while Joachims [2005] restate this theorem for the $\text{SVM}^{\Delta}_{multi}$ optimization problem. Hence, if the search for the most violated constraint can be performed in polynomial time:

$$\arg\max_{\bar{y}' \in \bar{Y}} \{\Delta(\bar{y}', \bar{y}) + \langle \boldsymbol{w}, \Psi(\bar{\boldsymbol{x}}, \bar{y}')\rangle\}, \tag{6.7}$$

the overall algorithm has polynomial time complexity.

It is worth noting that an exhaustive search over all $\bar{y}' \in \bar{\mathcal{Y}} \setminus \bar{y}$ is not feasible. However, the computation of the argmax in Eq. (6.7) can be stratified over all different contingency tables. Therefore, given any metric computed from the contingency table, including any variation of *Q-measure* based on different seminal metrics for *cperf* and *qperf*, Algorithm 3 efficiently returns the most violated constraint [Joachims, 2005]. Note also that the non-negativity condition imposed to Δ implies that estimation bias cannot be optimized because it may return negative values.

Algorithm 3 is based on the observation that there are only order $O(n^2)$ different contingency tables for a binary classification problem with n examples. Therefore, any loss function that can be computed from the contingency table can take at

Algorithm 3 Algorithm for computing arg max with non-linear loss functions

Input: $\bar{\boldsymbol{x}} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n)$ and $\bar{y} = (y_1, \ldots, y_n)$, $\bar{Y}$
$(i^p_1, \ldots, i^p_{\#pos}) \leftarrow$ sort $\{i : y_i = +1\}$ by $\langle \boldsymbol{w}, \boldsymbol{x}_i \rangle$
$(i^n_1, \ldots, i^n_{\#neg}) \leftarrow$ sort $\{i : y_i = -1\}$ by $\langle \boldsymbol{w}, \boldsymbol{x}_i \rangle$
for $a = 0$ to $\#pos$ **do**
  $c \leftarrow \#pos - a$
  set $y'_{i^p_1}, \ldots, y'_{i^p_a}$ to $+1$ AND $y'_{i^p_{a+1}}, \ldots, y'_{i^p_{\#pos}}$ to -1
  for $d = 0$ to $\#neg$ **do**
    $b \leftarrow \#neg - d$
    set $y'_{i^n_1}, \ldots, y'_{i^n_b}$ to $+1$ AND $y'_{i^n_{b+1}}, \ldots, y'_{i^n_{\#neg}}$ to -1
    $v \leftarrow \Delta(a, b, c, d) + \boldsymbol{w}^T \sum_{i=1}^{n} y'_i \boldsymbol{x}_i$
    if v is the largest so far **then**
      $\bar{y}^* \leftarrow (y'_1, \ldots, y'_n)$
    end if
  end for
end for
Return: $\bar{y}^*$

most $O(n^2)$ different values. Thus, by slightly rewriting the algorithm, it can be implemented to run in time $O(n^2)$; while exploiting that many loss functions are upper bounded, pruning can further improve the runtime of the algorithm.

6.4 Experimental setup

The main objective of this section is to study the behavior of the method that optimizes the quantification loss presented in this chapter, comparing its performance with other state-of-the-art approaches. The main difference with first experimental designs followed for quantification is that our empirical analysis is neither focused on a particular domain, nor a specific range of train or test prevalences. We aim to cover a broader or more general scope, following the methodology that we have previously introduced Chapter 4 and applied with success in Chapter 5. Again, we use standard datasets with known positive prevalence, along with the adaptations of stratified cross-validation and Friedman-Nemenyi statistical test. Specifically, the experiments are designed to answer the following questions:

1. Do the empirical results support the use of a learner optimizing a quantification loss function instead of a classification one?
2. Do we obtain any clear benefit by considering both classification and quantification simultaneously during learning?

The rest of section is organized as follows. First we describe the experimental setup, including datasets, algorithms and statistical tests. Then we present the results obtained from the experiments, evaluating them in terms of AE and KLD. Finally we discuss these results, trying to answer the questions stated before.

6.4.1 Datasets

In summary, we collected results from the last 22 datasets in Table 5.2, excluding *acute.a* and *acute.b* because they are too simple to extract any relevant insight by applying a complex learning algorithm like Joachim's SVM^{Δ}_{multi} [Joachims, 2005].. We then apply a stratified 10-fold cross-validation for each of them, preserving their original class distribution. After each training, we always assess the performance of the resulting model with 11 test sets generated from the remaining fold, varying the positive prevalence from 0% to 100% in steps of 10% (see Section 4.1.1).

We therefore performed 220 training processes and 2,420 tests for every system we evaluated. This setup generates 242 cross-validated results for each algorithm, that is, 22 datasets $\times$ 11 test distributions.

6.4.2 Algorithms

We choose CC, AC, Max, X, T50 and MS as state-of-the-art quantifiers from Forman's proposals (see Sections 3.1 and 3.2), considering CC as baseline. The underlying classifier for all these algorithms is a linear SVM from the *libsvm* library [Chang and Lin, 2011], with default parameters. The process of learning and threshold characterization, discussed in Section 3.2.1, is common to all these models, reducing the total experiment time and guaranteeing an equivalent root SVM for them all. Moreover, as Forman points out, MS method may show an odd behavior when the denominator in Equation (3.1) is too small, recommending to discard any threshold with $tpr - fpr < 1/4$. However, he does not make any recommendation in case there is no threshold that avoids that restriction. Therefore, we decided to fix these missing values with the values obtained by Max method, which provides the threshold with the greatest value for that difference.

The group of models based on learning a classifier optimizing a quantification metric consists of two approaches. On the one hand, our proposed *Q-measure*, using *recall* and NAS as seminal metrics (see Section 6.2). We consider three *Q-measure* variants: Q0.5, Q1 and Q2, representing models that optimize Equation (6.2) with β at 0.5, 1 and 2 respectively. On the other hand, we also include a method called NAS, which represents the approach suggested by Esuli and Sebastiani [2010] using NAS as the target measure. The reasoning for choosing NAS instead of any other quantification loss function is that we consider that both approaches should use the same quantification metric, differing in that our

proposal combines that measure with *recall*. This guarantees a fair comparison. All these systems are learned by means of SVM^{Δ}_{multi} Joachims [2005], described in Section 6.3.

6.4.3 Estimation of *tpr* and *fpr* characteristics

The estimations of *tpr* and *fpr* for quantification correction, defined in Equation (3.1), are obtained through a standard 10-fold cross-validation after learning the root model. Other alternatives like 50-fold or LOO are discarded because they are much more computationally expensive and they are prone to yield biased estimations, producing uneven corrections in practice.

It is also worth noting that we do not apply this correction for Q0.5, Q1, Q2 and NAS. Hence, end models just counts how many items are predicted as positive, like in CC method. This decision is supported by the fact that our main objective is to evaluate the performance of models obtained from the optimization of these metrics, isolated from any other factor. Moreover, given that these systems are based on SVM^{Δ}_{multi}, the estimation of *tpr* and *fpr* is much more expensive and it did not show a clear improvement in our preliminary experiments.

In fact, although the theory behind Equation (3.1) is well founded, in practice there exist cases where this correction involves a greater quantification error. Anyhow, these issues are out of the scope of this dissertation, offering an interesting opportunity to perform a deeper analysis for future works.

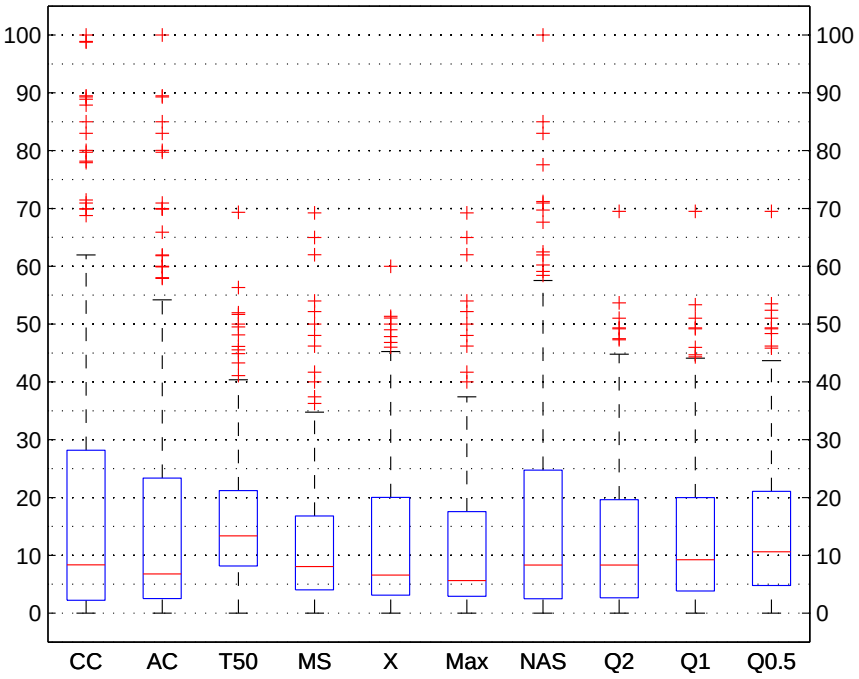
6.5 Empirical analysis

This section presents the experimental results in terms of two standard quantification measures: *AE* and *KLD*. Each of them provides a different perspective. In summary, we collect results from 22 datasets, applying a stratified 10-fold cross-validation for them all and assessing the performance of the resulting model with 11 test sets generated from the remaining fold (see Section 4.1). Recall that only the quantification outputs provided by AC, X, Max, T50 and MS are adjusted by means of Equation (3.1).

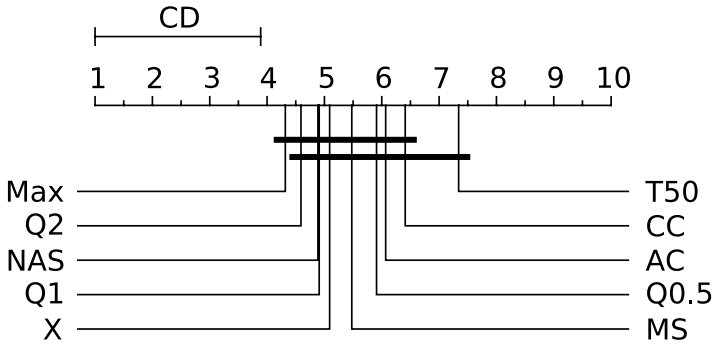
Thus, we consider that the 10 algorithms are compared over 22 domains, regardless of the number of prevalences that are tested for each of them, resulting in a CD of 2.8883 for the Nemenyi post-hoc test at 5% significance level.

6.5.1 Analysis of AE measurements

The first approach that we follow is to represent the results for all test conditions in all datasets with a boxplot of each system under study. Thus, in Figure 6.8a

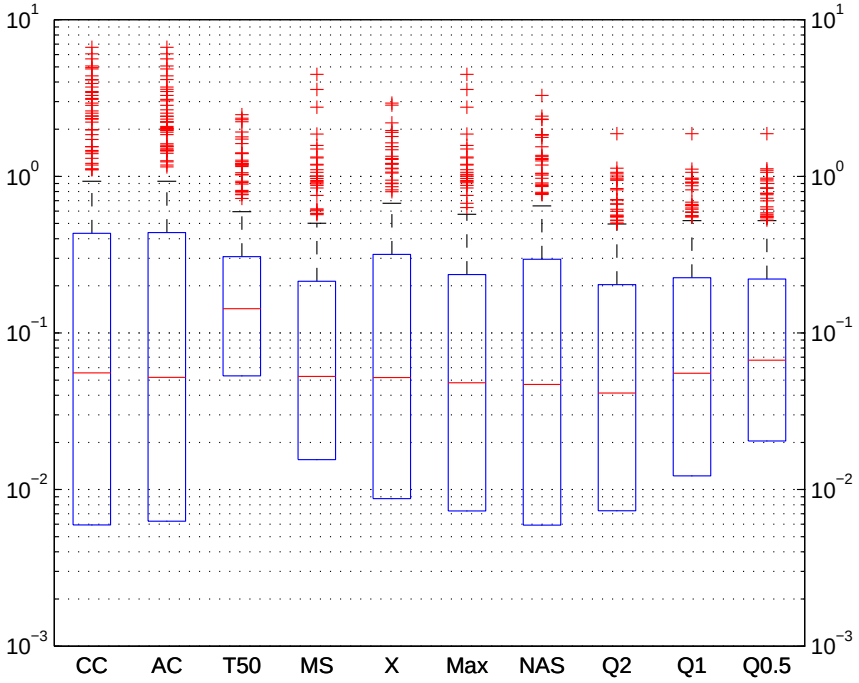


(a) Boxplots of all AE results

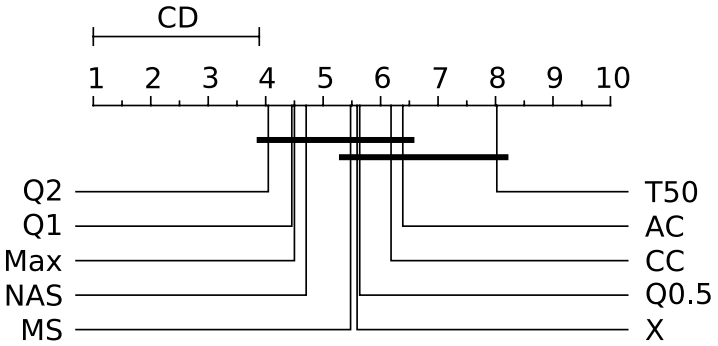


(b) Nemenyi at 5% (CD = 2.8883)

Figure 6.8: Statistical comparisons for Q_β experiments in terms of AE



(a) Boxplots of all *KLD* results



(b) Nemenyi at 5% (CD = 2.8883)

Figure 6.9: Statistical comparisons for Q_β experiments in terms of *KLD*

we can observe the range of errors for every system for AE measurements. Each box represents the first and third quartile by means of the lower and upper side respectively and the median or second quartile by means of the inner red line. The whiskers extend to the most extreme results that are not considered outliers, while the outliers are plotted individually. In this representation, we consider as outliers any point greater than the third quartile plus 1.5 times the inter-quartile range.

We distinguish three main groups in Figure 6.8a according to the learning procedure followed. The first one, including CC and AC, shows strong discrepancies between actual and estimated prevalences of up to 100%. These systems appear to be very unstable under specific circumstances. The second group includes T50, MS, X and Max, all of which are based on threshold selection policies (see Section 3.2.1). The T50 method stands out as the worst approach in this group due to the upward shift of its box. The final group comprises SVM^{Δ}_{multi} models: Q0.5, Q1, Q2 and NAS. The Q_β versions of this last group seems more stable than NAS, without extreme values over 70 and showing more compact boxes.

Friedman's null hypothesis is rejected at the 5% significance level. The overall results of the Nemenyi test are shown in Figure 6.8b, in which each system is represented by a thin line, linked to its name on one side and to its average rank on the other. The thick horizontal segments connect non significantly different methods at a confidence level of 5%. Therefore, this plot suggests that Max and our proposal, represented by Q2, are the methods that perform best in this experiment in terms of AE score comparison for Nemenyi's test. In this setting, we have no statistical evidence of differences between the two approaches. Neither do they show clear differences with other systems. We can only appreciate that Max is significantly better than T50.

It is worth noting that the results of Friedman-Nemenyi test are exactly the same for AE and NAS. The reason is that given any two systems, their ranking order is equal in both metrics. The mathematical proof is straightforward. Note that this is not fulfilled for other metrics, like KLD.

6.5.2 Analysis of KLD measurements

Although in most cases the analysis of AE results could be sufficient in order to discriminate an appropriate model for a specific real-world task, we also provide a complementary analysis of our experiments in terms of KLD. Looking at Figure 6.8, we can notice that the differences between some systems are quite subtle in terms of AE, while in Figure 6.9 we observe that these differences are evidenced a little bit more. For instance, Max and MS shows larger outliers in terms of KLD, due to the fact that this metric is closer to a quadratic error (see Figures 6.4 and 6.5).

Analyzing the results of Nemenyi test in Figure 6.9b, our approach obtains the

best rank, represented again by Q2, which is designed to give more weight to quantification metric during learning. However, except for T50, this system is not significantly better than other models. Q1, Max and NAS are also statistically differentiable from T50.

6.5.3 Discussion of results

In order to make more clear the discussion of the results, we try to answer the questions raised at the beginning of Section 6.4:

1. Do the empirical results support the use of a learner optimizing a quantification loss function instead of a classification one?

   The fact is that the best ranks are dominated by this kind of methods, in conjunction with Max. Although the differences with respect to other systems are not statistically significant in general.

   In any case, our approach, initially suggested by Esuli and Sebastiani, is theoretically well-founded and it is not based on any heuristic rule. From that point of view, we strongly believe that the methods presented here should be considered for future studies in the field of quantification. At least, they offer a different learning bias with respect to current approaches, which can produce better results in some domains.

   Moreover, it is also remarkable that none of the quantification methods evaluated in this experiment are corrected by means of Equation (3.1), as discussed in Section 6.4.3. Thus, these methods can be considered as variants of CC, which can be further improved with similar strategies as those applied in AC, Max, X, MS or T50.

2. Do we obtain any clear benefit by considering both classification and quantification simultaneously during learning?

   As we suspected, our variant obtains better results than the original proposal of Esuli and Sebastiani in terms of pure quantification performance (see AE results in Figure 6.8 and KLD results in Figure 6.9).

   In some cases NAS induces very poor classification models, despite it benefits from the definition of the optimization problem of SVM^{Δ}_{multi}, presented in Equation (6.6). Note that the constraints of the optimization problem $(\Psi(\bar{\boldsymbol{x}}, \bar{y}) - \Psi(\bar{\boldsymbol{x}}, \bar{y}'))$ are established with respect to the actual class of each example, which would be produced by the *perfect classifier*. Thus, the algorithm is biased to those models that are more similar to the perfect classifier even when the target loss function is not. Nevertheless, in practice, this learning bias is not able to rectify the drawbacks derived from the intrinsic design of pure quantification metrics, which assigns equal score

to any model that simply neutralizes false positive errors with the same amount of false negative errors. Actually, our first intuition was that their proposal should provide even worse classifiers due to this fact. As we discuss in Section 6.1, the key problem is that pure quantification metrics produce several optimum points within the hypothesis search space, conversely to what occurs with other metrics, in which there is only one.

In summary, not only our approach provides better quantification results than NAS, but also we consider that it is more reliable in general. Moreover, it is more flexible, allowing the practitioner to adjust the weight of both components of *Q-measure* taking into account the specific requirements of the problem under study by means of β parameter. In fact, provided that when $\beta \to \infty$ our method optimizes only the quantification component, it includes NAS as a particular case. This calibration is not needed in general and can be fixed from the experimental design. As rule of thumb, we suggest $\beta = 2$, because according to discussion of Figures 6.6 and 6.7, and to the analysis of empirical results, it effectively combines the best features of both components.

Chapter 7

Conclusions and Future Work

After a comprehensive discussion of the general background related to dataset-shift in Chapter 2 and state-of-the-art quantification algorithms in Chapter 3, we have discussed the core research studies of this dissertation. We started by formalizing a new methodology for statistical comparisons of quantifiers in Chapter 4, which is then applied to validate our proposed quantification approaches studied in Chapter 5 and Chapter 6.

This final chapter is divided in two main parts, first one highlights our most relevant contributions, in order to present directions for future work on last section.

7.1 Conclusions

The three main contributions of this dissertation are:

- Presentation of the first research study that formalizes an specialized methodology for statistical comparisons of several quantifiers over multiple test prevalences [Barranquero et al., 2013].
- Design and implementation of two simple and cost-effective weighting strategies for nearest neighbour algorithms, offering competitive quantification performance in practice [Barranquero et al., 2013].
- Design and implementation of the first learning method that optimizes a quantification metric, proposing a new family of parametric loss functions that are able to balance quantification and classification measurements [Barranquero et al., under review].

7.1.1 Methodology for statistical comparisons of quantifiers

Given that the required experiment methodology for quantification is relatively uncommon and has yet to be properly standardized and validated by machine learning community, in Chapter 4 we have proposed a new methodology for

statistical comparisons of several quantifiers over multiple test prevalences through stratified resampling.

The key contribution of this new methodology for quantification is that it allows us to analyze relevant properties of these comparatives from a statistical point of view. Furthermore, it also provides meaningful insights about which algorithms are significantly better, with a certain confidence degree, thanks to the adaptation of the two Friedman post-hoc statistical tests proposed by Demšar [2006], and the redesign of test set generation in stratified k-fold cross-validation.

The main difference with respect to standard cross-validation procedures and related statistical tests is that we need to evaluate performance over whole sets, rather than by means of individual classification outputs. Moreover, quantification assessment requires evaluating performance over a broad spectrum of test sets with different class distributions, instead of using a single test set. That is why traditional machine learning techniques for statistical comparisons of classification models are not directly applicable and need to be adapted.

Therefore we perform an exhaustive review of these related statistical approaches, discussing their main strengths and weaknesses. After this study we describe our proposed methodology in detail, adapting these existing procedures to the specific requirements of quantification comparatives. We consider that the core strength of our proposal is that we have prevailed robustness, in terms of lower Type I errors, against reducing Type II errors (see Section 4.2.4).

7.1.2 Cost-effective nearest neighbour quantification

In Chapter 5, we present several alternative quantification algorithms based on traditional NN rules, including the well-known KNN and two simple weighting strategies, identified as PWK and PWK$^{\alpha}$. From the main objective of studying the behavior of NN methods in the context of quantification, we propose an instance-based approach able to provide competitive performance while balancing simplicity and effectiveness. This study establishes a new baseline approach for dealing with prevalence estimation in binary problems.

We have found that, in general, weighted NN-based algorithms offer cost-effective performance. The conclusions drawn from Nemenyi post-hoc tests analized in Section 5.3 suggest that PWK and PWK$^{\alpha}$ stand out as the best approaches, without statistical differences between them, but offering clear statistical differences with respect to less robust models like CC, AC or T50.

Our experiments do not provide any discriminative indicator regarding which of these two weighting strategies is more recommendable for real-world applications. The final decision should be taken in terms of the specific needs of the problem, the constraints of the environment, or the complexity of the data, among other factors.

Notwithstanding, taking into account the observations discussed in Section 5.3.5, it appears that PWK could be more appropriate when the minority class is much more relevant, while PWK$^\alpha$ seems to behave more conservatively with respect to the majority class. Furthermore, PWK is simpler, its weights are more easily interpretable and it only requires calibrating the number of neighbors.

7.1.3 Quantification optimization via robust loss functions

Finally, in Chapter 6 we study the problem from a completely different perspective. As Esuli and Sebastiani [2010] point out, state-of-the art quantification algorithms do not optimize the loss function applied during model validation or comparison. Following their line of research, we claim that optimizing only a quantification metric during model training do not address sufficiently the problem, because we could obtain quantifiers having poor quantification behavior, due to an incoherent underlying model in terms of classification abilities (see Section 6.1).

In this regard, the most important question behind this study is whether is it really advisable to rely on quantification models that do not distinguish between positives and negatives at an individual level. But, how could this issue be alleviated during quantifier training? Formally, the way to solve any machine learning problem comprises two steps: we have to define a suitable metric and design an algorithm that optimizes it. Therefore, the combination of *Q-measure*, defined in Section 6.2, and the multivariate algorithm by Joachims [2005], presented in Section 6.3, offers a formal solution for quantifier learning.

The main contributions of this research are the study of the first quantification-oriented learning approach, that is, the first algorithm that optimizes a quantification metric; and the definition of a parametric loss function for quantification. This proposal is not only theoretically well-founded, but also offers competitive performance compared with state-of-the-art quantifiers.

7.1.4 Overall discussion of contributions

Although NN-based proposals may seem technically simple, it is worth noting the valuable effort that have been invested in analyzing the problem in order to adapt these algorithms to a relatively new optimization task. This study has also allowed us to understand the problem more deeply, setting the foundations for designing our proposed methodology.

Moreover, the value of simple solutions has been praised many times in the literature. As a matter of fact, one can never tell, a priori, how much of the structure in a domain can be captured by a very simple decision rule, while simplicity is advantageous for theoretical, practical and even academical reasons.

Actually, simple (early) models usually provide largest gains, which can be over 90% of the predictive power that can be achieved, while they are less likely to overfit [Hand, 2006; Holte, 2006]. Obviously, this does not mean that the more complex decision rules should be cast aside, but that the simple decision rules should not be dismissed out of hand. This is the case for our proposed NN solutions, which are based on euclidean distance and simple weighting strategies. However, this neither mean that NN approaches could not provide more complex decision rules in turn.

Conversely, the proposed multivariate approach for optimizing *Q-measure* may suffer from excessive complexity, measured in terms of computational cost. This is the one of the reasons why we have not applied Forman's correction during its experimental validation, resulting in that these experiments are not directly comparable with our previous NN study. Nevertheless, we consider that both approaches are complementary.

Interestingly, Demšar [2006] draws attention to an alternative opinion among statisticians about that significance tests should not be performed at all since they are often misused, either due to misinterpretation or by putting too much stress on their results. Nevertheless he claims that statistical tests provide certain reassurance about the validity and non-randomness of published results, though, they should be performed correctly and the resulting conclusions should be drawn cautiously.

But the most relevant point raised by him is that statistical tests should not be the deciding factor for or against publishing a work. Other merits of the proposed algorithm that are beyond the grasp of statistical testing should also be considered and possibly even favored over pure improvements in predictive power.

In this regard, our *Q-measure* approach provides competitive quantification results with respect to state-of-the-art quantifiers, although it seems that it is not as cost-effective as our NN-proposals. However, as we have already discussed, it is theoretically well-founded and it is not based on any heuristic rule. From that point of view, we strongly believe that it may be considered for future studies in the field of quantification. At least, it offers a different learning bias with respect to current approaches, which can produce better results in some domains.

Moreover, it is also worth remarking that *Q-measure* experimental results are not adjusted by means of Equation (3.1). Thus, these methods can be considered as variants of CC, which can be further improved with similar strategies as those applied in AC, Max, X, MS or T50.

7.2 Future work

Some of the results obtained during the development of our research suggest that correcting by means of Equation (3.1) may produce undesired results, obtaining even worse quantification predictions than without correction in some particular cases. The big problem is behind the estimation of *tpr* and *fpr*, which theoretically are invariant distribution characteristics, but in practice, estimating them through empirical data or evidence is not exempt of pitfalls. In fact, all threshold selection policies proposed by Forman [2008] are more or less focused on finding thresholds where the estimation of those rates have less variance, i.e., where these estimations are more *stable*.

7.2.1 Optimization of root models for threshold calibration

However, there is another source of problems, originated by the fact that threshold search is performed over a base hyperplane that may not be the best seed model. In other words, given that the classification accuracy of the underlying classifier may be irrelevant as long as the estimations of *tpr* and *fpr* are *reliable*; therefore, it is possible that this root model may play a crucial role.

If this was the case, a new question arises as an interesting area of research for optimizing classification models that were more robust in terms of minimizing the variability of *tpr* and *fpr* estimations. A completely different approach, where new bodies of knowledge may be applied, could be addressed through meta-heuristic optimization of root models.

Possible future directions for NN-based quantification could involve the selection of parameters through grid-search procedures, optimizing metrics with respect to equivalent rules as those applied for Max, X or T50, or even using these rules to calibrate the weights of each class during learning.

7.2.2 Analysis of power and stability of statistical tests

As might be expected, the experimental setting proposed in this dissertation have been redesigned several times before reaching the final form followed in this dissertation and its associate publications. In this regard, we would like to point out that having more test prevalences may reduce variance, but imply more dependence among test results.

This issue should deserve more research for future work, where a straightforward research study may be based on experimental analysis of power and stability of statistical tests for quantification comparatives, as it has been already done for equivalent statistical test for classification [García et al., 2010; Garcia and

Herrera, 2008; Demšar, 2006]. While analyzing alternative statistical tests based on resampling [Westfall and Young, 2004].

Moreover, appropriate collections of data, extracted directly from different snapshots of the same populations and showing natural shifts in their distributions, are required in order to further analyze the quantification problem from a real-world perspective.

Bibliography

R. Alaiz-Rodríguez and N. Japkowicz. Assessing the impact of changing environments on classifier performance. In *Advances in Artificial Intelligence, LNCS 5032 (Proceedings of the 21st Canadian Conference on Artificial Intelligence, AI'08)*, pages 13–24. Springer-Verlag, 2008.

R. Alaiz-Rodríguez, A. Guerrero-Curieses, and J. Cid-Sueiro. Minimax regret classifier for imprecise class distributions. *Journal of Machine Learning Research*, 8:103–130, 2007.

R. Alaiz-Rodríguez, A. Guerrero-Curieses, and J. Cid-Sueiro. Improving classification under changes in class and within-class distributions. *Bio-Inspired Systems: Computational and Ambient Intelligence (Proceedings of IWANN'09)*, pages 122–130, 2009.

R. Alaiz-Rodríguez, A. Guerrero-Curieses, and J. Cid-Sueiro. Class and subclass probability re-estimation to adapt a classifier in the presence of concept drift. *Neurocomputing*, 74(16):2614–2623, 2011.

E. Alpaydm. Combined 5 × 2 cv F test for comparing supervised classification learning algorithms. *Neural computation*, 11(8):1885–1892, 1999.

R. Barandela, J.S. Sánchez, V. García, and E. Rangel. Strategies for learning in class imbalance problems. *Pattern Recognition*, 36(3):849–851, 2003.

J. Barranquero, P. González, J. Díez, and J. J. del Coz. On the study of nearest neighbour algorithms for prevalence estimation in binary problems. *Pattern Recognition*, 46(2):472—482, 2013.

J. Barranquero, J. Díez, and J. J. del Coz. Quantification-oriented learning based on reliable classifiers. *Pattern Recognition*, under review.

M. Barreno, B. Nelson, A.D. Joseph, and J.D. Tygar. The security of machine learning. *Machine Learning*, 81(2):121–148, 2010.

A. Bella, C. Ferri, J. Hernández-Orallo, and M.J. Ramírez-Quintana. Quantification via probability estimators. In *Proceedings of the 10th IEEE International Conference on Data Mining (ICDM'10)*, pages 737–742. IEEE, 2010.

A. Bella, C. Ferri, J. Hernández-Orallo, and M.J. Ramírez-Quintana. On the effect of calibration in classifier combination. *Applied Intelligence*, 38(4):566–585, 2013a.

A. Bella, C. Ferri, J. Hernández-Orallo, and M.J. Ramírez-Quintana. Aggregative quantification for regression. *Data Mining and Knowledge Discovery*, pages 1–44, 2013b.

B. Bergmann and G. Hommel. Improvements of general multiple test procedures for redundant systems of hypotheses. In *Multiple Hypothesenprüfung (Multiple Hypotheses Testing)*, pages 100–115. Springer, 1988.

S. Bickel and T. Scheffer. Discriminative learning under covariate shift. *Journal of Machine Learning Research*, 10:2137–2155, 2009.

B. Biggio, G. Fumera, and F. Roli. Multiple classifier systems for robust classifier design in adversarial environments. *International Journal of Machine Learning and Cybernetics*, 1(1-4):27–41, 2010.

P. Broos and K. Branting. Compositional instance-based learning. In *Proceedings of the 12th AAAI National Conference*, volume 1, pages 651–656, 1994.

M. Budka and B. Gabrys. Density-preserving sampling: Robust and efficient alternative to cross-validation for error estimation. *Neural Networks and Learning Systems, IEEE Transactions on*, 24(1):22 –34, 2013. ISSN 2162-237X. doi: 10.1109/TNNLS.2012.2222925.

Y.S. Chan and H.T. Ng. Estimating class priors in domain adaptation for word sense disambiguation. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, pages 89–96. ACL, 2006.

C.-C. Chang and C.-J. Lin. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 2(3:27):1–27, 2011.

D.A. Cieslak and N.V. Chawla. A framework for monitoring classifiers performance: when and why failure occurs? *Knowledge and Information Systems*, 18(1):83–108, 2009.

S. Cost and S. Salzberg. A weighted nearest neighbor algorithm for learning with symbolic features. *Machine Learning*, 10(1):57–78, 1993.

T. Cover and P. Hart. Nearest neighbor pattern classification. *IEEE Transactions on Information Theory*, 13(1):21–27, 1967.

J. Demšar. Statistical comparisons of classifiers over multiple data sets. *Journal of Machine Learning Research*, 7:1–30, 2006.

T.G. Dietterich. Approximate statistical tests for comparing supervised classification learning algorithms. *Neural Computation*, 10:1895–1923, 1998.

C. Domeniconi, J. Peng, and D. Gunopulos. Locally adaptive metric nearest-neighbor classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 24(9):1281–1285, 2002.

C. Drummond, R.C. Holte, et al. Cost curves: An improved method for visualizing classifier performance. *Machine Learning*, 65(1):95–130, 2006.

M.C. du Plessis and M. Sugiyama. Semi-supervised learning of class balance under class-prior change by distribution matching. In *Proceedings of ICML'12*, 2012.

R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern Classification*. John Wiley & Sons, 2nd edition, 2001.

M. Dudık, R.E. Schapire, and S.J. Phillips. Correcting sample selection bias in maximum entropy density estimation. In *Proceedings of NIPS'05*, volume 18. The MIT Press, 2005.

M. Dudık, S.J. Phillips, and R.E. Schapire. Maximum entropy density estimation with generalized regularization and an application to species distribution modeling. *Journal of Machine Learning Research*, 8:1217–1260, 2007.

O.J. Dunn. Multiple comparisons among means. *Journal of the American Statistical Association*, 56(293):52–64, 1961.

B. Efron and G. Gong. A leisurely look at the bootstrap, the jackknife, and cross-validation. *The American Statistician*, 37(1):36–48, 1983.

B. Efron and R. Tibshirani. Improvements on cross-validation: The .632+ bootstrap method. *Journal of the American Statistical Association*, 92(438):pp. 548–560, 1997. ISSN 01621459. URL `http://www.jstor.org/stable/2965703`.

G.G. Enas and S.C. Choi. Choice of the smoothing parameter and efficiency of k-nearest neighbor classification. *Computers & Mathematics with Applications*, 12(2):235–244, 1986.

A. Esuli and F. Sebastiani. Sentiment quantification. *IEEE Intelligent Systems*, 25(4):72–75, 2010.

A. Esuli and F. Sebastiani. Variable-constraint classification and quantification of radiology reports under the acr index. *Expert Systems with Applications*, under review. doi: 10.1016/j.eswa.2012.12.052.

T. Fawcett. ROC graphs: Notes and practical considerations for researchers. *Machine Learning*, 31:1–38, 2004.

T. Fawcett and P.A. Flach. A response to Webb and Ting's on the application of ROC analysis to predict classification performance under varying class distributions. *Machine Learning*, 58(1):33–38, 2005.

R.A. Fisher. *Statistical methods and scientific inference (2nd edition).* Hafner Publishing Co., New York, 1959.

J.L. Fleiss, B. Levin, and M.C. Paik. *Statistical Methods for Rates and Proportions.* Wiley Series in Probability and Statistics. John Wiley & Sons, 3rd edition, 2003.

G. Forman. A method for discovering the insignificance of one's best classifier and the unlearnability of a classification task. In *In Proceedings of ICML'02, Data Mining Lesson Learned Workshop*, 2002.

G. Forman. Counting positives accurately despite inaccurate classification. In *Proceedings of ECML'05*, pages 564–575. Springer-Verlag, 2005.

G. Forman. Quantifying trends accurately despite classifier error and class imbalance. In *Proceedings of ACM SIGKDD'06*, pages 157–166, 2006.

G. Forman. Quantifying counts and costs via classification. *Data Mining and Knowledge Discovery*, 17(2):164–206, 2008.

G. Forman, E. Kirshenbaum, and J. Suermondt. Pragmatic text mining: minimizing human effort to quantify many issues in call logs. In *Proceedings of ACM SIGKDD'06*, pages 852–861. ACM, 2006.

A. Frank and A. Asuncion. UCI machine learning repository. University of California, Irvine, 2010. URL `http://archive.ics.uci.edu/ml/`.

M. Friedman. The use of ranks to avoid the assumption of normality implicit in the analysis of variance. *Journal of the American Statistical Association*, 32 (200):675–701, 1937.

M. Friedman. A comparison of alternative tests of significance for the problem of m rankings. *The Annals of Mathematical Statistics*, 11(1):86–92, 1940.

S. Garcia and F. Herrera. An extension on "statistical comparisons of classifiers over multiple data sets" for all pairwise comparisons. *Journal of Machine Learning Research*, 9:2677–2694, 2008.

S. García, A. Fernández, J. Luengo, and F. Herrera. Advanced nonparametric tests for multiple comparisons in the design of experiments in computational intelligence and data mining: Experimental analysis of power. *Information Sciences*, 180(10):2044–2064, 2010.

A. Globerson, C.H. Teo, A. Smola, and S. Roweis. *Dataset Shift in Machine Learning*, chapter An Adversarial View of Covariate Shift and A Minimax Approach, pages 179—198. Adaptive Computation and Machine Learning. The MIT Press, Cambridge, Massachusetts, 2009.

P. González, E. Alvarez, J. Barranquero, J. Díez, R. Gonzalez-Quiros, E Nogueira, A. Lopez-Urrutia, and J. J. del Coz. Multiclass support vector machines with example-dependent costs applied to plankton biomass estimation. *IEEE Transactions on Neural Networks and Learning Systems*, 24(11):1901–1905, 2013.

V. González-Castro. *Adaptive Texture Description and Estimation of the Class Prior Probabilities for Seminal Quality Control*. PhD thesis, Department of Electrical, Systems and Automatic Engineering. University of León, 2011.

V. González-Castro, R. Alaiz-Rodríguez, L. Fernández-Robles, R. Guzmán-Martínez, and E. Alegre. Estimating class proportions in boar semen analysis using the hellinger distance. In Nicolás García-Pedrajas, Francisco Herrera, Colin Fyfe, José Benítez, and Moonis Ali, editors, *Trends in Applied Intelligent Systems*, volume 6096 of *Lecture Notes in Computer Science*, pages 284–293. Springer Berlin / Heidelberg, 2010.

V. González-Castro, R. Alaiz-Rodríguez, and E. Alegre. Class distribution estimation based on the hellinger distance. *Information Sciences*, 218(1):146–164, 2013.

M. Grbovic and S. Vucetic. Tracking concept change with incremental boosting by minimization of the evolving exponential loss. In *Proceedings of ECML'11*, volume Part I, pages 516–532. Springer-Verlag, 2011.

A. Gretton, A. Smola, J. Huang, M. Schmittfull, K. Borgwardt, and B. Schoelkopf. *Dataset Shift in Machine Learning*, chapter Covariate Shift and Local Learning by Distribution Matching, pages 131–160. Adaptive Computation and Machine Learning. The MIT Press, Cambridge, Massachusetts, 2009.

P. Groot, A. ten Teije, and F. van Harmelen. A quantitative analysis of the robustness of knowledge-based systems through degradation studies. *Knowledge and Information Systems*, 7(2):224–245, 2005.

A. Guerrero-Curieses, R. Alaiz-Rodriguez, and J. Cid-Sueiro. Cost-sensitive and modular land-cover classification based on posterior probability estimates. *International Journal of Remote Sensing*, 30(22):5877–5899, 2009.

D.J. Hand. Classifier technology and the illusion of progress. *Statistical Science*, 21(1):1–14, 2006.

D.J. Hand and V. Vinciotti. Choosing k for two-class nearest neighbour classifiers with unbalanced classes. *Pattern Recognition Letters*, 24(9-10):1555–1562, 2003.

W. Hardle. *Applied nonparametric regression*. Cambridge University Press, Cambridge, 1992.

K. Hattori and M. Takahashi. A new nearest-neighbor rule in the pattern classification problem. *Pattern recognition*, 32(3):425–432, 1999.

K. Hechenbichler and K.P. Schliep. Weighted k-nearest-neighbor techniques and ordinal classification. Technical Report 399 (SFB 386), Ludwig-Maximilians University, Munich, 2004. URL `http://www.stat.uni-muenchen.de/sfb386/papers/dsp/paper399.ps`.

D.P. Helmbold and P.M. Long. Tracking drifting concepts by minimizing disagreements. *Machine Learning*, 14(1):27–45, 1994.

José Hernández-Orallo, Peter Flach, and César Ferri. Roc curves in cost space. *Machine Learning*, 93(1):71–91, 2013.

Yosef Hochberg. A sharper bonferroni procedure for multiple tests of significance. *Biometrika*, 75(4):800–802, 1988.

T.R. Hoens, R. Polikar, and N.V. Chawla. Learning from streaming data with concept drift and imbalance: an overview. *Progress in Artificial Intelligence*, 1 (1):89–101, 2012.

S. Holm. A simple sequentially rejective multiple test procedure. *Scandinavian Journal of Statistics*, 6(2):65–70, 1979.

R.C. Holte. Elaboration on two points raised in "classifier technology and the illusion of progress". *Statistical Science*, 21(1):24–26, 2006.

G. Hommel. A stagewise rejective multiple test procedure based on a modified bonferroni test. *Biometrika*, 75(2):383–386, 1988.

J. Huang, A.J. Smola, A. Gretton, and K.M. Borgwardt. Correcting sample selection bias by unlabeled data. In *Proceedings of NIPS'07*, pages 601–608. The MIT Press, 2007.

R.L. Iman and J.M. Davenport. Approximations of the critical region of the friedman statistic. *Communications in Statistics*, 9(6):571–595, 1980.

T. Joachims. A support vector method for multivariate performance measures. In *Proceedings of ICML'05*, pages 377–384. ACM, 2005.

Pilsung Kang and Sungzoon Cho. Locally linear reconstruction for instance-based learning. *Pattern Recognition*, 41(11):3507–3518, 2008.

M.G. Kelly, D.J. Hand, and N.M. Adams. The impact of changing populations on classifier performance. In *Proceedings of ACM SIGKDD'99*, pages 367–371. ACM, 1999.

R. Klinkenberg and T. Joachims. Detecting concept drift with support vector machines. In *Proceedings of ICML'00*, pages 487–494. Morgan Kaufmann Publishers Inc., 2000.

R. Kohavi. A study of cross-validation and bootstrap for accuracy estimation and model selection. In *Proceedings of IJCAI'95*, 1995.

Pavel Laskov and Richard Lippmann. Machine learning in adversarial environments. *Machine learning*, 81(2):115–119, 2010.

P. Latinne, M. Saerens, and C. Decaestecker. Adjusting the outputs of a classifier to new a priori probabilities may significantly improve classification accuracy: Evidence from a multi-class problem in remote sensing. In *Proceedings of ICML'01*, pages 298–305. Morgan Kaufmann Publishers Inc., 2001.

V. López, A. Fernández, J.G. Moreno-Torres, and F. Herrera. Analysis of preprocessing vs. cost-sensitive learning for imbalanced classification. open problems on intrinsic data characteristics. *Expert Systems with Applications*, 39(7):6585—6608, 2011.

J.G. Moreno-Torres, X. Llora, D.E. Goldberg, and R. Bhargava. Repairing fractures between data using genetic programming-based feature extraction: A case study in cancer diagnosis. *Information Sciences*, 222:805–823, 2010. ISSN 0020-0255. doi: 10.1016/j.ins.2010.09.018. URL `http://www.sciencedirect.com/science/article/pii/S0020025510004585`.

J.G. Moreno-Torres, T. Raeder, R. Alaiz-Rodríguez, N.V. Chawla, and F. Herrera. A unifying view on dataset shift in classification. *Pattern Recognition*, 45(1): 521–530, 2012a.

J.G. Moreno-Torres, J.A. Sáez, and F. Herrera. Study on the impact of partition-induced dataset shift on k-fold cross-validation. *IEEE Transactions on Neural Networks and Learning Systems*, 23(8):1304–1312, 2012b.

P. Nemenyi. *Distribution-free Multiple Comparisons*. PhD thesis, Princeton University, 1963.

M. Núñez, R. Fidalgo, and R. Morales. Learning in environments with unknown dynamics: Towards more robust concept learners. *Journal of Machine Learning Research*, 8:2595–2628, 2007.

S.J. Phillips, M. Dudık, and R.E. Schapire. A maximum entropy approach to species distribution modeling. In *Proceedings of the twenty-first international conference on Machine learning*. ACM, 2004.

S.J. Phillips, M. Dudík, J. Elith, C.H. Graham, A. Lehmann, J. Leathwick, and S. Ferrier. Sample selection bias and presence-only distribution models: implications for background and pseudo-absence data. *Ecological Applications*, 19(1):181–197, 2009.

F. Provost and T. Fawcett. Robust classification for imprecise environments. *Machine Learning*, 42(3):203–231, 2001.

J. Quionero-Candela, M. Sugiyama, A. Schwaighofer, and N.D. Lawrence, editors. *Dataset Shift in Machine Learning*. Adaptive Computation and Machine Learning. The MIT Press, Cambridge, Massachusetts, 2009.

T. Raeder, T.R. Hoens, and N.V. Chawla. Consequences of variability in classifier performance estimates. In *Proceedings of the 2010 IEEE International Conference on Data Mining*, pages 421–430. IEEE Computer Society, 2010.

T. Rakthanmanon, E.J. Keogh, S. Lonardi, and S. Evans. MDL-based time series clustering. *Knowledge and Information Systems*, 33(2):371–399, 2012.

E. Ramentol, Y. Caballero, R. Bello, and F. Herrera. SMOTE-RSB*: a hybrid preprocessing approach based on oversampling and undersampling for high imbalanced data-sets using SMOTE and rough sets theory. *Knowledge and Information Systems*, 33(2):245–265, 2012.

P Refaeilzadeh, L. Tang, and H. Liu. *Encyclopedia of Database System*, chapter Cross-validation, pages 532–538. Springer US, 2009.

D.M. Rom. A sequentially rejective test procedure based on a modified bonferroni inequality. *Biometrika*, 77(3):663–665, 1990.

M. Saerens, P. Latinne, and C. Decaestecker. Adjusting the outputs of a classifier to new a priori probabilities: A simple procedure. *Neural Computation*, 14(1): 21–41, 2002.

L. Sánchez, V. González-Castro, E. Alegre-Gutiérrez, and R. Alaiz-Rodríguez. Classification and quantification based on image analysis for sperm samples with uncertain damaged/intact cell proportions. In *Image Analysis and Recognition, LNCS 5112 (Proceedings of the 5th International Conference on Image Analysis and Recognition, ICIAR'08)*, pages 827–836. Springer-Verlag, 2008.

J.C. Schlimmer and R.H. Granger. Incremental learning from noisy data. *Machine learning*, 1(3):317–354, 1986.

J.P. Shaffer. Modified sequentially rejective multiple test procedures. *Journal of the American Statistical Association*, 81(395):826–831, 1986.

J.P. Shaffer. Multiple hypothesis testing. *Annual Review of Psychology*, 46:561–584, 1995.

D.J. Sheskin. *Handbook of parametric and nonparametric statistical procedures*. Chapman & Hall/CRC, 2003.

H. Shimodaira. Improving predictive inference under covariate shift by weighting the log-likelihood function. *Journal of Statistical Planning and Inference*, 90(2): 227–244, 2000.

M. Stone. Cross-validatory choice and assessment of statistical predictions. *Journal of the Royal Statistical Society. Series B (Methodological)*, 36(2):111–147, 1974a.

M. Stone. Cross-validation and multinomial prediction. *Biometrika*, 61(3):509–515, 1974b.

M. Stone. Asymptotics for and against cross-validation. *Biometrika*, 64(1):pp. 29–35, 1977. ISSN 00063444. URL http://www.jstor.org/stable/2335766.

A. J. Storkey. *Dataset Shift in Machine Learning*, chapter When Training and Test Sets are Different: Characterising Learning Transfer, pages 3–28. Adaptive Computation and Machine Learning. The MIT Press, Cambridge, Massachusetts, 2009.

M. Sugiyama and M. Kawanabe. *Machine Learning in Non-Stationary Environments: Introduction to Covariate Shift Adaptation*. Adaptive Computation and Machine Learning. The MIT Press, Cambridge, Massachusetts, 2012.

M. Sugiyama, M. Krauledat, and K.R. Müller. Covariate shift adaptation by importance weighted cross validation. *Journal of Machine Learning Research*, 8:985–1005, 2007a.

M. Sugiyama, S. Nakajima, and H. Kashima. Direct importance estimation with model selection and its application to covariate shift adaptation. In *Proceedings of NIPS'07*, 2007b.

M. Sugiyama, N. Rubens, and K.R. Müller. *Dataset Shift in Machine Learning*, chapter A Conditional Expectation Approach to Model Selection and Active Learning under Covariate Shift, pages 131–160. Adaptive Computation and Machine Learning. The MIT Press, Cambridge, Massachusetts, 2009.

N.A. Syed, H. Liu, and K.K. Sung. Handling concept drifts in incremental learning with support vector machines. In *Proceedings of ACM SIGKDD'99*, pages 317–321. ACM, 1999.

S. Tan. Neighbor-weighted k-nearest neighbor for unbalanced text corpus. *Expert Systems with Applications*, 28(4):667 – 671, 2005. ISSN 0957-4174. doi: DOI: 10.1016/j.eswa.2004.12.023. URL http://www.sciencedirect.com/science/article/pii/S0957417404001708.

S. Tan. An effective refinement strategy for KNN text classifier. *Expert Systems with Applications*, 30(2):290–298, 2006.

L. Tang, H. Gao, and H. Liu. Network quantification despite biased labels. In *Proceedings of the 8th Workshop on Mining and Learning with Graphs (MLG'10) at ACM SIGKDD'10*, pages 147–154. ACM, 2010.

I. Tsochantaridis, T. Hofmann, T. Joachims, and Y. Altun. Support vector machine learning for interdependent and structured output spaces. In *Proceedings of the 21st international Conference on Machine Learning (ICML)*, pages 823–830, 2004.

I. Tsochantaridis, T. Joachims, T. Hofmann, and Y. Altun. Large margin methods for structured and interdependent output variables. *Journal of Machine Learning Research*, 6:1453–1484, 2005.

A. Tsymbal. The problem of concept drift: definitions and related work. Technical report, Computer Science Dept., Trinity College, Dublin, 2004. URL `https://www.scss.tcd.ie/publications/tech-reports/reports.04/TCD-CS-2004-15.pdf`.

J.W. Tukey. Comparing individual means in the analysis of variance. *Biometrics*, 5(2):99–114, 1949.

C. J. van Rijsbergen. *Information retrieval*. Butterworths, London, 1979.

C.J. van Rijsbergen. Foundations of evaluation. *Journal of Documentation*, 30(4): 365–373, 1974.

S. Vucetic and Z. Obradovic. Classification on data with biased class distribution. In *Proceedings of ECML'01*, pages 527–538. Springer-Verlag, 2001.

K. Wang, S. Zhou, C.A. Fu, and J.X. Yu. Mining changes of classification by correspondence tracing. In *Proceedings of the Third SIAM International Conference on Data Mining (SDM'03)*, 2003.

G.I. Webb and K.M. Ting. On the application of ROC analysis to predict classification performance under varying class distributions. *Machine Learning*, 58(1):25–32, 2005.

G.M. Weiss. Mining with rarity: a unifying framework. *ACM SIGKDD Explorations Newsletter*, 6(1):7–19, 2004.

G.M. Weiss and F. Provost. Learning when training data are costly: The effect of class distribution on tree induction. *Journal of Artificial Intelligence Research*, 19:315–354, 2003.

P.H. Westfall and S.S. Young. *Resampling-based Multiple Testing: Examples and Methods for p-Value Adjustment*. John Wiley and Sons, 2004.

G. Widmer and M. Kubat. Effective learning in dynamic environments by explicit context tracking. In *In Proceedings of ECML'93*, pages 227–243. Springer, 1993.

G. Widmer and M. Kubat. Learning in the presence of concept drift and hidden contexts. *Machine learning*, 23(1):69–101, 1996.

F. Wilcoxon. Individual comparisons by ranking methods. *Biometrics Bulletin*, 1 (6):80–83, 1945.

M.A. Wong and T. Lane. A kth nearest neighbour clustering procedure. *Journal of the Royal Statistical Society, Series B, Methodological*, pages 362–368, 1983.

J.C. Xue and G.M. Weiss. Quantification and semi-supervised classification methods for handling changes in class distribution. In *Proceedings of ACM SIGKDD'09*, pages 897–906. ACM, 2009.

K. Yamazaki, M. Kawanabe, S. Watanabe, M. Sugiyama, and K.R. Müller. Asymptotic bayesian generalization error when training and test distributions are different. In *Proceedings of ICML'07*, pages 1079–1086, 2007.

Y. Yang, G.I. I Webb, J. Cerquides, K.B. Korb, J. Boughton, and K.M. Ting. To select or to weigh: A comparative study of linear combination schemes for superparent-one-dependence estimators. *IEEE Transactions on Knowledge and Data Engineering*, 19(12):1652–1665, 2007a.

Y. Yang, X. Wu, and X. Zhu. Conceptual equivalence for contrast mining in classification learning. *Data & Knowledge Engineering*, 67(3):413–429, 2008.

Ying Yang, Geoff Webb, Kevin Korb, and Kai Ming Ting. Classifying under computational resource constraints: anytime classification using probabilistic estimators. *Machine Learning*, 69(1):35–53, 2007b.

B. Zadrozny. Learning and evaluating classifiers under sample selection bias. In *Proceedings of ICML'04*. ACM, 2004.

J. Zar. *Biostatistical analysis (5th edition)*. Prentice Hall, Englewood Clifs, New Jersey, 2009.

M.L. Zhang and Z.H. Zhou. ML-KNN: A lazy learning approach to multi-label learning. *Pattern Recognition*, 40(7):2038–2048, 2007.

I. Žliobaitė. Learning under concept drift: an overview. Technical report, Faculty of Mathematics and Informatics, Vilnius University, Lithuania, 2010. URL `http://arxiv.org/abs/1010.4784`.