



UNIVERSIDAD DE BUENOS AIRES  
Facultad de Ciencias Exactas y Naturales

**Estimación de proporción de clases en muestras no etiquetadas  
mediante modelos de cuantificación**

Tesis presentada para optar al título de Magister en Estadística Matemática de la  
Universidad de Buenos Aires

**Ing. Maximiliano Marufo da Silva**

Director de tesis: Dr. Andrés Farall

Buenos Aires, 2024.



# Índice general

<b>1. Problema</b>	<b>1</b>
1.1. Introducción . . . . .	1
1.2. Tipos de Cuantificación . . . . .	2
1.3. Marco teórico . . . . .	3
1.4. Cambios en las distribuciones de los datos . . . . .	4
1.5. El problema de clasificar y contar . . . . .	6
1.6. Cuantificadores para la mejora de la Clasificación . . . . .	7
<b>2. Estimación Puntual</b>	<b>9</b>
2.1. Métodos Agregativos . . . . .	9
2.1.1. Con clasificadores generales . . . . .	9
2.1.2. Con clasificadores específicos . . . . .	13
2.2. Métodos No Agregativos . . . . .	13
<b>3. Estimación por Intervalos</b>	<b>15</b>
<b>4. Métodos de Evaluación</b>	<b>17</b>
4.1. Propiedades . . . . .	18
4.2. Métricas . . . . .	18
4.2.1. Sesgo . . . . .	18
4.2.2. Error Absoluto . . . . .	19
4.2.3. Error Absoluto Normalizado . . . . .	19
4.2.4. Error Cuadrático . . . . .	19
4.2.5. Error Absoluto Relativo . . . . .	19
4.2.6. Error Absoluto Relativo Normalizado . . . . .	20
4.2.7. Divergencia de Kullback-Leibler . . . . .	20
4.2.8. Divergencia de Kullback-Leibler Normalizada . . . . .	21
4.3. Elección de la Métrica . . . . .	21
4.4. Protocolos . . . . .	21
<b>5. Resultados</b>	<b>23</b>

<b>A. Calibración</b>	<b>25</b>
A.1. Definición . . . . .	26
A.2. Diagramas de confiabilidad . . . . .	26
A.3. Métodos de Evaluación . . . . .	26
A.4. Métodos de Calibración . . . . .	26
A.4.1. Modelos Binarios . . . . .	26
A.4.2. Modelos Multiclase . . . . .	26

# Capítulo 1

## Problema

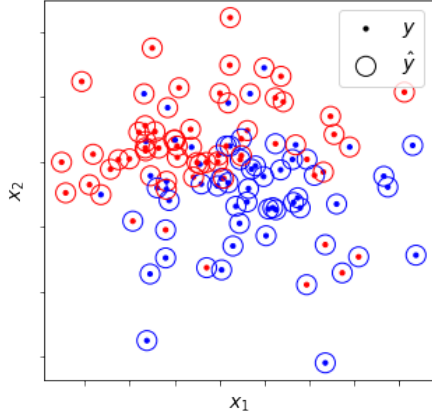
### 1.1. Introducción

En algunas aplicaciones vinculadas a la clasificación, el objetivo final no es determinar a qué clase (o clases) pertenece cada una de las instancias individuales de un conjunto de datos no etiquetado, sino estimar la proporción (también llamada ‘prevalencia’, ‘frecuencia relativa’ o ‘probabilidad prior’) de cada clase en los datos sin etiquetar. En los últimos años se ha señalado que, en estos casos, tiene sentido optimizar directamente algoritmos de aprendizaje automático para este objetivo, en lugar de simplemente optimizar clasificadores para etiquetar instancias individuales.

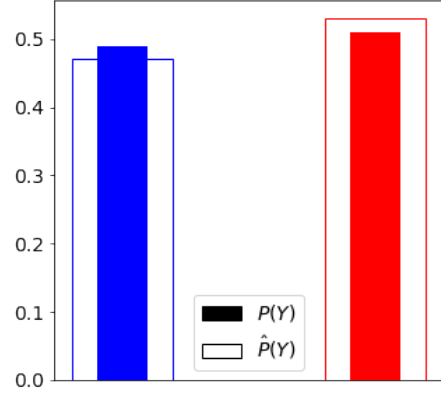
La tarea de ajustar estimadores de prevalencia de clases a través del aprendizaje supervisado se conoce como ‘aprender a cuantificar’ o, más simplemente, cuantificar o *quantification* (término acuñado por Forman [1], quien planteó el problema por primera vez). Se sabe que cuantificar mediante la clasificación de cada instancia sin etiquetar a través de un clasificador estándar y luego contando las instancias que han sido asignadas a cada clase (el método *Classify & Count*) generalmente conduce a estimadores de prevalencia de clases sesgados, es decir, obtienen poca exactitud en la cuantificación. Como resultado, se han desarrollado métodos que abordan la cuantificación como una tarea en sí.

Para ver la importancia de diferenciar el problema de cuantificación del de clasificación, veamos dos ejemplos. En el primero, una empresa que ofrece un servicio a sus clientes realiza una encuesta con varias preguntas para determinar el grado de satisfacción de cada persona. El objetivo de la empresa es determinar aquellos clientes que podrían no estar conformes con el servicio y ofrecerles una mejora en las condiciones para retenerlos. En el segundo ejemplo, una consultora analiza tweets para estimar el grado de aprobación de candidatos políticos. Aquí, la consultora no está interesado en predecir si un individuo específico está a favor o en contra, sino en cuántos encuestados, del número total de encuestados, aprueban al candidato, es decir, en conocer la prevalencia de la clase positiva.

Mientras en el primer escenario el interés es a nivel individual, en el último, el nivel agregado es lo que importa; en otras palabras, en el primer escenario la clasificación es el objetivo, mientras que en el segundo el verdadero objetivo es la cuantificación. De hecho, en la mayoría de las aplicaciones las predicciones que interesan no son a nivel individual sino a nivel colectivo; ejemplos de tales campos son la investigación de mercado, la ciencia política, las ciencias sociales, modelado ecológico y epidemiología.



(a) Clasificación



(b) Cuantificación

**Figura 1.1:** En la clasificación, la predicción es a nivel individual, mientras que en la cuantificación es a nivel agregado.

En resumen, y generalizando no sólo para clasificación sino también a otros problemas (regresión, ordinalidad, etc), la tarea de cuantificación consiste en proporcionar predicciones agregadas para conjuntos de datos, en vez de predicciones particulares sobre los datos individuales. Si bien en principio no es necesario realizar predicciones por cada individuo, muchos de los métodos se basan en obtener la cuantificación de esa manera, ya que hacer predicciones individuales suele ser un requisito de por sí de las aplicaciones prácticas, o porque ya existen en ellas modelos que las generen.

La literatura sobre métodos relacionados con cuantificación está un tanto desconectada. Algunos de los métodos que pueden usarse como cuantificadores han sido ideados para otros fines, principalmente para mejorar la precisión en clasificación cuando cambia el dominio. El desempeño de este último grupo ha sido normalmente estudiado solo en términos de mejora en las tareas de clasificación pero no como cuantificadores. Dado este escenario, y debido a la variedad de campos en los que ha surgido como una necesidad de aplicación, los algoritmos que se pueden aplicar para tareas de cuantificación aparecen en artículos que usan diferentes palabras clave y nombres, como *counting* [2], *prior probability shift* [3, 4], *posterior probability estimation* [5], *class prior estimation* [6–8], *class prior change* [9], *prevalence estimation* [10], *class ratio estimation* [11] o *class distribution estimation* [12–14], por citar solo algunos de ellos.

## 1.2. Tipos de Cuantificación

Aunque el estudio de la cuantificación se ha centrado principalmente en el dominio de clasificación, la cuantificación también aparece en otros tipos de problemas de aprendizaje automático, como la regresión, la clasificación ordinal, el aprendizaje sensible al costo y la cuantificación en redes.

De manera similar a la regresión, aprender a cuantificar admite diferentes problemas de interés aplicativo, basados en cuántas clases distintas existen en el problema, y en cuántas de las clases se pueden atribuir al mismo tiempo al mismo individuo. Así, los problemas de cuantificación se dividen de esta manera:

1. Etiquetado simple (*Single-Label Quantification -SLQ-*): cuando cada individuo pertenece exactamente a una de las clases en  $C = \{c_1, \dots, c_{\#C}\}$ .
2. Etiquetado múltiple (*Multi-Label Quantification -MLQ-*): cuando cada individuo puede pertenecer a cualquier número de clases (cero, una o varias) en  $C = \{c_1, \dots, c_{\#C}\}$ .
3. Cuantificación Binaria (*Binary Quantification -BQ-*):
  - a) en *SLQ* con  $\#C = 2$ , (en este caso  $C = \{c_1, c_2\}$ , y cada individuo pertenece a  $c_1$  o  $c_2$ )
  - b) en *MLQ* con  $\#C = 1$ , (en este caso  $C = \{c\}$ , y cada individuo pertenece o no a  $c$ )
4. Cuantificación Ordinal (*Ordinal Quantification -OQ-*): cuando existe un orden  $c_1 \prec \dots \prec c_{\#C}$  en  $C = \{c_1, \dots, c_{\#C}\}$ .
5. Cuantificación de Regresión (*Regression Quantification -RQ-*): cuando no hay un conjunto de clases involucradas, sino que cada individuo está etiquetado con una puntuación de valor real y la cuantificación equivale a estimar la fracción de ítems cuya puntuación está en un intervalo dado  $[a, b]$  con  $a, b \in \mathbb{R}^d$ .

### 1.3. Marco teórico

Si hablamos entonces de cuantificación binaria, se tiene que por cada muestra  $i \in \{1, \dots, n\}$ ,  $(\mathbf{X}_i, Y_i, S_i)$  es un vector de variables aleatorias tal que  $\mathbf{X}_i \in \mathbb{R}^d$  son las características de la muestra,  $Y_i \in C$  con  $C = \{1, 0\}$  indica la clase a la que pertenece y  $S_i \in \{1, 0\}$  indica si fue etiquetada (y pertenece entonces al conjunto de entrenamiento) o no. Es decir, cuando  $S_i = 0$ , entonces  $Y_i$  no es observable. El objetivo es estimar  $\theta := \mathbb{P}(Y = 1 | S = 0)^1$ , es decir, la prevalencia de etiquetas positivas entre muestras no etiquetadas. Esta prevalencia no se asume de ser la misma que en las muestras etiquetadas,  $\mathbb{P}(Y = 1 | S = 1)$ . Además, el estimador de  $\theta$  debe depender sólo de los datos disponibles, es decir, de las características de todas las muestras y de las etiquetas que fueron obtenidas. Los supuestos que se asumen [15] son:

- $(\mathbf{X}_1, Y_1, S_1) \dots (\mathbf{X}_n, Y_n, S_n)$  son independientes
- Por cada  $s \in \{0, 1\}$ ,  $(\mathbf{X}_1, Y_1) | S_1 = s, \dots, (\mathbf{X}_n, Y_n) | S_n = s$  son idénticamente distribuidas.
- Por cada  $(y_1, \dots, y_n) \in \{0, 1\}^n$ ,  $(\mathbf{X}_1, \dots, \mathbf{X}_n)$  es independiente de  $(S_1, \dots, S_n)$  condicionado a  $(Y_1, \dots, Y_n) = (y_1, \dots, y_n)$

Usando la distribución de probabilidad conjunta, podemos factorizar usando las distribuciones condicionales:

$$\mathbb{P}(\mathbf{X}, Y, S) = \mathbb{P}(\mathbf{X} | Y, S) \mathbb{P}(Y | S) \mathbb{P}(S) \quad (1.1)$$

Luego, usando el tercer supuesto mencionado, podemos hacer [3]:

$$\mathbb{P}(\mathbf{X}, Y, S) = \mathbb{P}(\mathbf{X} | Y) \mathbb{P}(Y | S) \mathbb{P}(S) \quad (1.2)$$

---

<sup>1</sup>En cuantificación, se lo nombra generalmente como  $p$  (o  $p_1, \dots, p_{\#C}$  o  $p(c)$  para el caso multiclase) en vez de  $\theta$ , por lo que en este trabajo también se usará esta nomenclatura.

Si bien existen varios métodos propuestos para el aprendizaje de cuantificación [16, 17], el mismo es todavía relativamente desconocido incluso para expertos en aprendizaje automático. La razón principal es la creencia errónea de que es una tarea trivial que se puede resolver usando un método directo, como *CC*. La cuantificación requiere métodos más sofisticados si el objetivo es obtener modelos óptimos, y su principal dificultad radica en la definición del problema, ya que las distribuciones de los datos de entrenamiento y de prueba pueden ser distintas. Por ejemplo, si la diferencia entre  $\mathbb{P}(Y = 1|S = 0)$  y  $\mathbb{P}(Y = 1|S = 1)$  es grande, los métodos simples como *CC* suelen tener bajo rendimiento.

## 1.4. Cambios en las distribuciones de los datos

En los últimos años ha habido un interés creciente en las aplicaciones que presentan cambios en las distribuciones de datos (conocido en la bibliografía por su término en inglés *dataset shift*). Estos problemas comparten el hecho de que la distribución de los datos utilizados para entrenar es diferente a la de los datos que se usan para predecir. Al igual que para el área de la cuantificación, aquí también la literatura sobre el tema está dispersa y diferentes autores usan diferentes nombres para referirse a los mismos conceptos, o usan el mismo nombre para diferentes conceptos.

Teniendo en cuenta que en los problemas de clasificación tenemos:

- Un conjunto de características o covariables  $\mathbf{X}$ .
- Una variable de respuesta  $Y$ .
- Una distribución de probabilidad conjunta  $\mathbb{P}(Y = y, \mathbf{X} = \mathbf{x})$ .

La probabilidad conjunta  $\mathbb{P}(Y, \mathbf{X})$  luego se puede escribir como  $\mathbb{P}(Y|\mathbf{X})\mathbb{P}(\mathbf{X})$  o como  $\mathbb{P}(\mathbf{X}|Y)\mathbb{P}(Y)$ . Por otro lado, cuando usamos los términos de entrenamiento (*train*) y prueba (*test*), nos referimos a los datos disponibles para entrenar al clasificador y los datos presentes en el entorno en el que se implementará el clasificador, respectivamente. Podemos entonces también separar los datos en dos distribuciones distintas, condicionando a la variable  $S$  definida en 1.3, siendo  $\mathbb{P}_{tr}(Y, \mathbf{X}) = \mathbb{P}(Y, \mathbf{X}|S = 1)$  y  $\mathbb{P}_{tst}(Y, \mathbf{X}) = \mathbb{P}(Y, \mathbf{X}|S = 0)$ .

El *dataset shift* aparece cuando las distribuciones conjuntas de entrenamiento y de prueba son diferentes, es decir, cuando  $\mathbb{P}_{tr}(Y, \mathbf{X}) \neq \mathbb{P}_{tst}(Y, \mathbf{X})$ . Moreno-Torres et al. [3] distingue las distintas variantes del dataset shift según qué elementos mencionados anteriormente cambian:

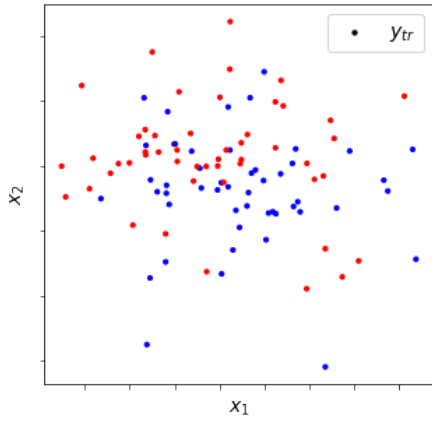
- *Covariate shift*, cuando  $\mathbb{P}_{tr}(Y|\mathbf{X}) = \mathbb{P}_{tst}(Y|\mathbf{X})$  y  $\mathbb{P}_{tr}(\mathbf{X}) \neq \mathbb{P}_{tst}(\mathbf{X})$
- *Prior probability shift*, cuando  $\mathbb{P}_{tr}(\mathbf{X}|Y) = \mathbb{P}_{tst}(\mathbf{X}|Y)$  y  $\mathbb{P}_{tr}(Y) \neq \mathbb{P}_{tst}(Y)$
- *Concept shift*, cuando  $\mathbb{P}_{tr}(Y|\mathbf{X}) \neq \mathbb{P}_{tst}(Y|\mathbf{X})$  y  $\mathbb{P}_{tr}(\mathbf{X}) = \mathbb{P}_{tst}(\mathbf{X})$  o  $\mathbb{P}_{tr}(\mathbf{X}|Y) \neq \mathbb{P}_{tst}(\mathbf{X}|Y)$  y  $\mathbb{P}_{tr}(Y) = \mathbb{P}_{tst}(Y)$

Otros tipos de dataset shift surgen cuando  $\mathbb{P}_{tr}(Y|\mathbf{X}) \neq \mathbb{P}_{tst}(Y|\mathbf{X})$  y  $\mathbb{P}_{tr}(\mathbf{X}) \neq \mathbb{P}_{tst}(\mathbf{X})$  y cuando  $\mathbb{P}_{tr}(\mathbf{X}|Y) \neq \mathbb{P}_{tst}(\mathbf{X}|Y)$  y  $\mathbb{P}_{tr}(Y) \neq \mathbb{P}_{tst}(Y)$ . Sin embargo, estos tipos de cambios no se consideran generalmente en la literatura ya que aparecen mucho más raramente, o incluso porque son difíciles o imposibles de resolver.

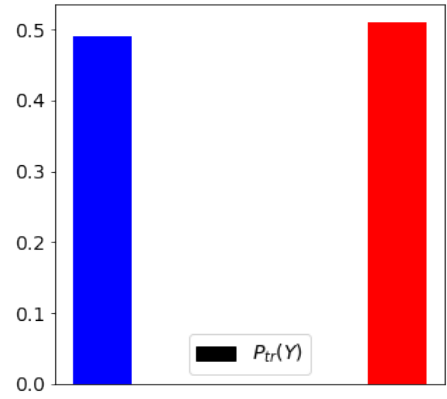


Está claro que el problema de cuantificación se trata de un caso donde  $\mathbb{P}_{tst}(Y)$  es desconocido. Además, la mayoría de los métodos de cuantificación propuestos asumen que  $\mathbb{P}_{tr}(\mathbf{X}|Y) = \mathbb{P}_{tst}(\mathbf{X}|Y)$ , por lo que estarían dentro de los casos de *prior probability shift*.

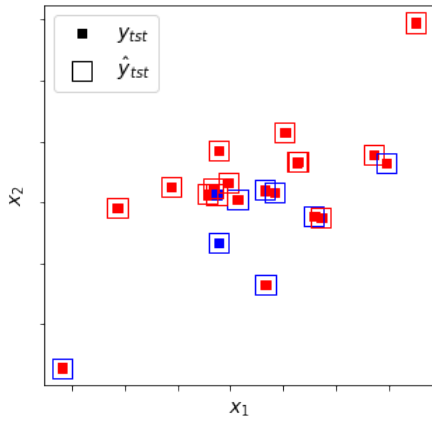
Por otro lado, en la mayoría de los casos el objetivo final de la implementación es estimar algún parámetro de  $\mathbb{P}_{tst}(Y)$ . Por ejemplo, como ya mencionamos anteriormente, en la cuantificación binaria, se desea estimar  $\theta := \mathbb{P}(Y = 1|S = 0)$ , o lo que es lo mismo,  $p_{tst} := \mathbb{P}_{tst}(Y = 1)$ . Es decir, en la cuantificación la tarea indirectamente suele ser aprender a aproximar una distribución desconocida (observando sólo características de una muestra) mediante una distribución conocida. En consecuencia, prácticamente todas las medidas de evaluación para la cuantificación son divergencias, es decir, medidas de cómo una distribución pronosticada difiere de la distribución real.



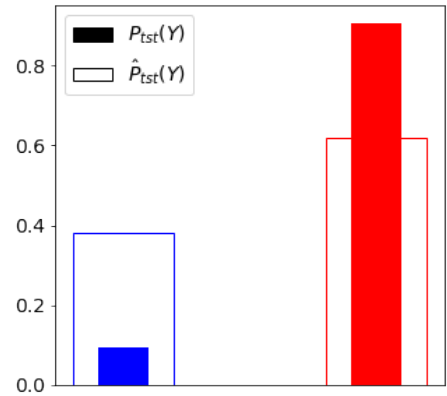
(a) Muestra de entrenamiento



(b) Prevalencia de clases en muestra de entrenamiento



(c) Clasificación en muestra de prueba. Para el modelo, las  $y_{tst}$  son desconocidas.



(d) Prevalencia de clases verdadera y cuantificación en muestra de prueba

**Figura 1.2:** El *prior probability shift* propio de los problemas de cuantificación puede hacer que los métodos simples de cuantificación, como *CC*, tengan grandes errores.

## 1.5. El problema de clasificar y contar

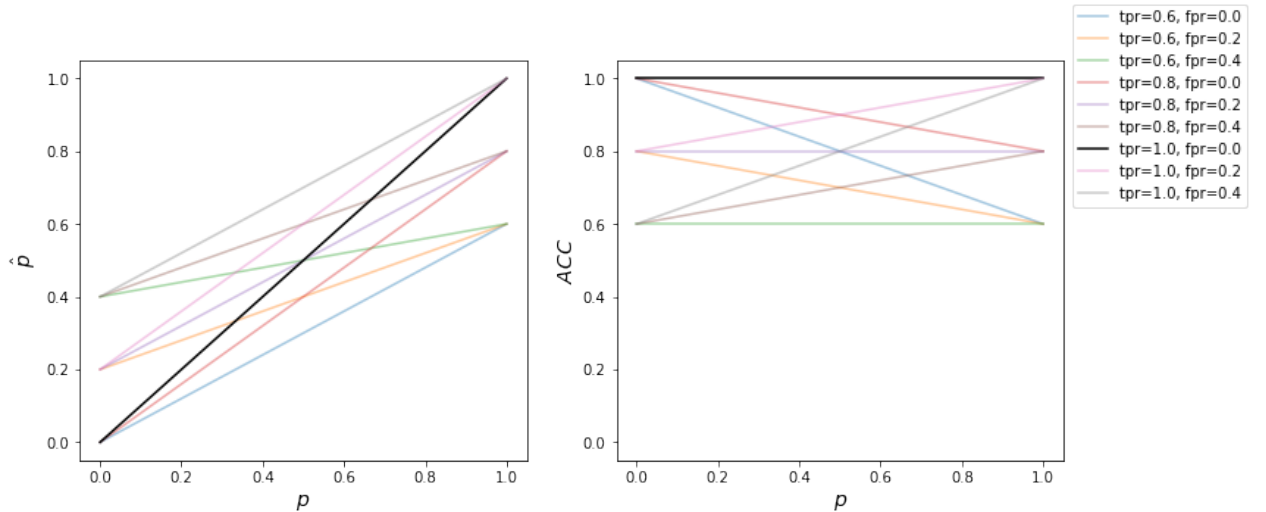
En ausencia de métodos para estimar los valores de prevalencia de clase de forma directa, el primer método que suele pensarse para hacerlo es *Classify & Count*, es decir, clasificar cada individuo sin etiquetar y estimar los valores de prevalencia de clase contando los individuos que fueron asignados a cada clase. Sin embargo, esta estrategia es subóptima: si bien un clasificador perfecto es también un cuantificador perfecto, un buen clasificador puede ser un mal cuantificador. Para ver esto, se puede ver la definición de  $F_1$ , una función de evaluación estándar para la clasificación binaria, que se define como:

$$F_1 = \frac{2 \cdot tp}{2 \cdot tp + fp + fn} \quad (1.3)$$

donde  $tp$ ,  $fp$ ,  $fn$  indican el número de verdaderos positivos, falsos positivos y falsos negativos, respectivamente. Un buen clasificador puede ser un mal cuantificador ya que  $F_1$  considera buenos aquellos clasificadores que mantienen la suma  $fp + fn$  al mínimo; sin embargo, el objetivo de un algoritmo de cuantificación debe ser mantener al mínimo  $|fp - fn|$ .

El análisis teórico de esta cuestión se basa en el supuesto de *prior probability shift* 1.4. Bajo tal supuesto, la estimación  $\hat{p}$  obtenida por el enfoque *CC* depende sólo de las características del clasificador, definido (para el caso binario) por su tasa de verdaderos positivos ( $tpr$ ), su tasa de falsos positivos ( $fpr$ ) y de la prevalencia real ( $p$ ):

$$\hat{p}(p) = p \cdot tpr + (1 - p) \cdot fpr \quad (1.4)$$



**Figura 1.3:** La línea negra representa el cuantificador y clasificador perfecto, respectivamente. Las otras líneas muestran las estimaciones teóricas de  $\hat{p}$  resultantes de aplicar la ecuación 1.4, y el *accuracy* correspondiente al clasificador, según se varían los valores de  $tpr$  y  $fpr$

El mal desempeño de *CC* fue demostrado mediante el siguiente teorema por Forman:

**Teorema 1.1** (Teorema de Forman). [18, p.169] Para un clasificador imperfecto, el método *CC* subestimaré la proporción verdadera de ejemplos positivos  $p$  en un conjunto de prueba para  $p > p^*$ , y sobreestimaré para  $p < p^*$ , donde  $p^*$  es la proporción particular en la cual el método *CC* estima

de forma correcta. Es decir, el método *CC* estima exactamente  $p^*$  para un conjunto de prueba con  $p^*$  muestras positivas.

La demostración (ver todos los detalles en [18, p.170]) supone que el cuantificador *CC* produce una predicción perfecta para una prevalencia concreta, llamada  $p^*$ , y estudia su comportamiento cuando la prevalencia cambia ligeramente. Suponiendo que  $\hat{p}(p^*) = p^*$ , cuando la prevalencia cambia en una cantidad  $\Delta \neq 0$ ,  $p^* + \Delta$ , la estimación del método *CC* en tal caso será:

$$\begin{aligned}\hat{p}(p^* + \Delta) &= (p^* + \Delta) \cdot tpr + (1 - (p^* + \Delta)) \cdot fpr \\ &= \hat{p}(p^*) + (tpr - fpr) \cdot \Delta \\ &= p^* + (tpr - fpr) \cdot \Delta\end{aligned}\tag{1.5}$$

La predicción del método *CC* será perfecta,  $\hat{p}(p^* + \Delta) = (p^* + \Delta)$ , sólo cuando el clasificador también es perfecto ( $tpr = 1$ ,  $fpr = 0$  y por tanto  $tpr - fpr = 1$ ). Pero en el caso habitual, en el cual el clasificador es imperfecto ( $0 \leq tpr - fpr < 1$ ), cuando la prevalencia aumenta ( $\Delta > 0$ ), *CC* la subestima ( $\hat{p} < p^* + \Delta$ ), y cuando la prevalencia disminuye ( $\Delta < 0$ ), *CC* la sobreestima ( $\hat{p} > p^* + \Delta$ ).

Un buen clasificador puede estar sesgado, es decir, puede mantener sus falsos positivos al mínimo sólo a expensas de una cantidad sustancialmente mayor de falsos negativos (o viceversa); si este es el caso, el clasificador es un mal cuantificador. Este fenómeno no es infrecuente, especialmente en presencia de datos desbalanceados. En tales casos, los algoritmos que minimizan las funciones de pérdida de clasificación (*Hamming*, *hinge*, etc) suelen generar clasificadores con tendencia a elegir la clase mayoritaria, lo que implica un número mucho mayor de falsos positivos que de falsos negativos para la clase mayoritaria, lo que significa a su vez que tal algoritmo tenderá a subestimar las clases minoritarias.

Los argumentos anteriores indican que no se debe considerar la cuantificación como un mero subproducto de la clasificación, y debe estudiarse y resolverse como una tarea en sí misma. Hay al menos otros dos argumentos que apoyan esta idea. Uno es que las funciones que se utilizan para evaluar la clasificación no se pueden utilizar para evaluar la cuantificación, ya que estas funciones miden, en general, cuántos individuos han sido mal clasificados, y no cuánto difiere la prevalencia de clase estimada del valor real. Esto significa que los algoritmos que minimizan estas funciones están optimizados para la clasificación, y no para la cuantificación. Un segundo argumento presentado por Forman [18] es que los métodos diseñados específicamente para cuantificar requieren menos datos de entrenamiento para alcanzar la misma precisión de cuantificación que los métodos estándar basados en *CC*. Si bien esta observación es de naturaleza empírica, también existen argumentos teóricos que sustentan este hecho [19].

## 1.6. Cuantificadores para la mejora de la Clasificación

Debido a los problemas mencionados anteriormente de los clasificadores frente a cambios en las distribuciones de los datos y frente a datos desbalanceados, los algoritmos de cuantificación están cada vez más frecuentemente también siendo usados en tareas que requieren predicciones individuales. Los mismos se emplean como suplemento de clasificadores para suplir sus defectos frente a estos problemas, ya que en algunos casos no sólo predicen los valores agregados, sino que también mejoran las predicciones a nivel individual.

Por ejemplo, el *prior probability shift* puede hacer que los clasificadores performen de manera subóptima. En el caso del clasificador óptimo de Bayes, dado por:

$$h(\mathbf{x}) = \arg \max_y p_{Y|\mathbf{X}=\mathbf{x}}(y) = \arg \max_y \frac{p_{\mathbf{X}|Y=y}(\mathbf{x})p_Y(y)}{p_{\mathbf{X}}(\mathbf{x})} \quad (1.6)$$

la decisión del clasificador depende de  $p_Y(y)$ , que es estimado con el dataset de entrenamiento, siendo  $\hat{p}_Y(y=1) = p_{tr}$ . Es decir, que en caso de  $\mathbb{P}_{tr}(Y) \neq \mathbb{P}_{tst}(Y)$ , la decisión final del clasificador puede verse afectada negativamente. Para mejorar el rendimiento del clasificador frente a estos casos, se debería usar  $\hat{p}_Y(y=1) = p_{tst}$ , pero como  $p_{tst}$  es generalmente desconocido, se puede usar un método de cuantificación para estimarlo [5, 8, 14, 20].

Los métodos de cuantificación pueden usarse no sólo para mejorar el rendimiento general de un clasificador, sino también para mejorar su equidad o *fairness* [21], es decir, su posibilidad de predecir resultados independientes de un cierto conjunto de variables que consideramos sensibles y no relacionadas con él (e.j.: género, etnia, orientación sexual, etc.). Por ejemplo, suponiendo que una variable  $S$  debe considerarse sensible, se puede estimar  $\mathbb{P}_{tr}(Y|S)$ . Luego, si los datos de entrenamiento están sesgados, por ejemplo, con  $\mathbb{P}_{tr}(Y=1|S=1) \gg \mathbb{P}_{tr}(Y=1|S=0)$ , pero se sabe que en las muestras a inferir esto no es así, se puede optimizar el modelo de clasificación imponiendo alguna penalidad basada en la estimación de  $\mathbb{P}_{tst}(Y|S)$ , siendo esta última obtenido por un cuantificador.

## Capítulo 2

# Estimación Puntual

Durante los últimos años, se han propuesto varios métodos de cuantificación desde diferentes perspectivas y con diferentes objetivos. En términos generales, se pueden distinguir dos grandes clases de métodos en la literatura. La primera clase es la de métodos agregativos, es decir, métodos que requieren la clasificación de todos los individuos como un paso intermedio. Dentro de los métodos agregativos, se pueden identificar dos subclases. La primera subclase incluye métodos basados en clasificadores de propósito general; en estos métodos la clasificación de los elementos individuales realizados como un paso intermedio puede lograrse mediante cualquier clasificador. La segunda subclase se compone, en cambio, de métodos que para clasificar los individuos, se basan en métodos de aprendizaje diseñados con la cuantificación en mente. La segunda clase es la de métodos no agregativos, es decir, métodos que resuelven la tarea de cuantificación “holísticamente”, es decir, sin clasificar a los individuos.

La idea de esta tesis no es la de mostrar todos los métodos propuestos hasta la actualidad, sino la de mencionar a continuación los métodos más populares. Como ejemplo de muestra se usará el mismo que en las figuras 1.1 y 1.2, en donde  $p_{tr} = 0.5$  y  $p_{tst} = 0.9$ .

### 2.1. Métodos Agregativos

#### 2.1.1. Con clasificadores generales

Dentro de los métodos agregativos, algunos de ellos requieren como entrada las etiquetas de clases predichas (es decir, clasificadores duros), mientras que otros requieren como entrada las probabilidades *a posteriori* de pertenencia a cada clase (es decir, clasificadores blandos)<sup>1</sup>. En estos últimos, además, las probabilidades *a posteriori* deben estar calibradas (para mayor información sobre calibración consultar el Apéndice A).

#### Clasificar y Contar (CC)

El método más sencillo y directo para construir un cuantificador para clasificación (tanto binaria como multiclase) es aplicar el enfoque *Classify & Count* [1]. *CC* juega un papel importante en la investigación de cuantificación ya que siempre se utiliza como el *baseline* que cualquier método

---

<sup>1</sup>Los clasificadores blandos se pueden convertir en duros usando umbrales de clasificación

de cuantificación razonable debe mejorar. Este método consiste simplemente en: (i) ajustar un clasificador duro, y luego (ii), utilizando dicho clasificador, clasificar las instancias de la muestra de prueba, contando la proporción de cada clase. Generalizando el estimador de  $CC$  para el caso multiclase, el mismo queda entonces definido por:

$$\hat{p}_{tst}^{CC}(c) = \frac{\#\{\mathbf{x} \in \mathbf{X}_{tst} | h(\mathbf{x}) = c\}}{\#\mathbf{X}_{tst}} \quad (2.1)$$

donde se uso  $h$  para la función de decisión del clasificador duro.

Es evidente que podemos obtener un cuantificador perfecto si el clasificador es también perfecto. El problema es que obtener un clasificador perfecto es casi imposible en aplicaciones reales, y luego el cuantificador hereda el sesgo del clasificador. Este aspecto se analiza en varios artículos tanto desde una perspectiva teórica como práctica, como lo hizo Forman, y como también ya lo hemos mencionado en 1.5.

*Ejemplo:* Para el caso de ejemplo, debemos contar la cantidad de predicciones positivas (rojas) en 1.2, y dividir las por el tamaño de la muestra de prueba. Es decir,  $\hat{p}_{tst}^{CC}(c=1) = \frac{19}{23} \approx 0.62$

## Clasificar, Contar y Ajustar (ACC)

Conocido en inglés como *Adjusted Classify & Count*, *Adjusted Count* [18] o también como *Confusion Matrix Method* [20], este método se basa en corregir las estimaciones de  $CC$  teniendo en cuenta la tendencia del clasificador a cometer errores de cierto tipo. Un modelo  $ACC$  está compuesto por dos elementos: un clasificador duro (como en  $CC$ ) y de las estimaciones de  $tpr$  y  $fpr$ . Dichas estimaciones pueden obtenerse usando validación cruzada o *cross-validation* o un conjunto de validación aparte. Luego, en la fase de predicción, el modelo obtiene una primera estimación  $\hat{p}$  de la misma forma que en  $CC$ , que, para el caso binario, luego es ajustado aplicando la siguiente fórmula<sup>2</sup>:

$$\hat{p}_{tst}^{ACC}(c=1) = \frac{\hat{p}_{tst}^{CC}(c=1) - fpr}{tpr - fpr} \quad (2.2)$$

Esta expresión se obtiene despejando la verdadera prevalencia  $p$  de la ecuación 2.1. La prevalencia ajustada  $\hat{p}_{tst}^{ACC}(c)$  depende entonces de la estimación  $\hat{p}_{tst}^{CC}(c)$  dada por  $CC$  y de  $tpr$  y  $fpr$ . Pero como  $tpr$  y  $fpr$  son desconocidas generalmente, se utilizan estimaciones.

El método  $ACC$  es teóricamente perfecto, independientemente de la métrica de *accuracy* obtenida con el clasificador, cuando se cumple el supuesto de *prior probability shift* 1.4 y cuando las estimaciones de  $tpr$  y  $fpr$  son perfectas. Desafortunadamente, es raro que se cumplan ambas condiciones en aplicaciones del mundo real:  $\mathbb{P}(\mathbf{X}|Y)$  puede tener variaciones entre los datos de entrenamiento y los de predicción, y es difícil obtener estimaciones perfectas para  $tpr$  y  $fpr$  en algunos dominios ya que suelen haber pequeñas muestras disponibles y/o están muy desequilibradas. Pero incluso en estos casos, el rendimiento del método  $ACC$  suele ser mejor que el de  $CC$ .

Partiendo de 2.1 y utilizando el teorema de probabilidad total, podemos extender 2.2 para el

---

<sup>2</sup>A veces, esta expresión conduce a un valor inválido de  $\hat{p}_{tst}^{ACC}$  que debe recortarse en el rango  $[0, 1]$  en un último paso.

caso multiclase:

$$\begin{aligned}\hat{p}_{tst}^{CC}(c = c_k) &= \hat{\mathbb{P}}_{tst}(h(\mathbf{x}) = c_k) \\ &= \sum_{j=1}^{\#C} \hat{\mathbb{P}}(h(\mathbf{x}) = c_k | y = c_j) \hat{p}_{tst}^{ACC}(c = c_j)\end{aligned}\quad (2.3)$$

donde  $\hat{p}_{tst}^{CC}(c = c_k)$  es la fracción de datos de *tst* que el clasificador  $h$  asigna a  $c_k$  (y por ende, es conocido), y  $\hat{\mathbb{P}}(h(\mathbf{x}) = c_k | y = c_j)$  es la estimación de probabilidad de que el clasificador  $h$  asigne la clase  $c_k$  a  $\mathbf{x}$  cuando este pertenece a la clase  $c_j$ . Estas probabilidades, al igual que *tpr* y *fpr* en el caso binario, deben estimarse mediante validación cruzada o con un conjunto de validación aparte [1, 10, 18]. Luego,  $\hat{p}_{tst}^{ACC}(c = c_j)$ , nuestras incógnitas (una por cada  $c_j$ ), pueden calcularse mediante un sistema de ecuaciones lineales con  $\#C$  ecuaciones y  $\#C$  incógnitas.

*Ejemplo:* Aquí debemos estimar el *tpr* y *fpr*. Para ello, deberíamos contar con una muestra de validación (distinta a la de prueba y la de entrenamiento) con etiquetas conocidas. Sin embargo, a modo de simplificación para este ejemplo, usaremos la misma muestra de prueba, siendo en este caso  $tn = 1$ ,  $fp = 1$ ,  $fn = 7$  y  $tp = 12$ , entonces  $tpr = 0.5$  y  $fpr \approx 0.63$ , y por lo tanto,  $\hat{p}_{tst}^{ACC}(c = 1) \approx \frac{0.62-0.5}{0.63-0.5} \approx 0.9$

### Clasificar y Contar Probabilístico (PCC)

Este método, conocido en inglés como *Probabilistic Classify and Count* [22, 23], es una variante de *CC* que utiliza un clasificador blando en vez de uno duro. Es decir, que la salida del clasificador será una estimación de la probabilidad *a posteriori*  $s(\mathbf{x}, y) = \hat{p}_{Y|X=\mathbf{x}}(y)$  por cada individuo  $\mathbf{x} \in \mathbb{R}^d$  y cada  $y \in C$ . El método consiste en estimar las  $\hat{p}_{tst}(c = c_j)$  mediante el valor esperado de la proporción de items que se predijeron como pertenecientes a cada clase  $c_j$ :

$$\begin{aligned}\hat{p}_{tst}^{PCC}(c = c_j) &= \hat{\mathbb{E}}[p_{Y|X=\mathbf{x}}(y = c_j)] \\ &= \frac{1}{m} \sum_{i=1}^m \hat{p}_{Y|X=\mathbf{x}_i}(y = c_j) \\ &= \frac{1}{m} \sum_{i=1}^m s(\mathbf{x}_i, y = c_j)\end{aligned}\quad (2.4)$$

con  $m = \#\mathbf{X}_{tst}$ . La intuición detrás de *PCC* es que las probabilidades *a posteriori* contienen mayor información que las decisiones de un clasificador duro y, por lo tanto, deberían ser usadas en su lugar. Sin embargo, Tasche [24, Corolario 6, p. 157 y p.163] demuestra que el comportamiento de *PCC* será similar al de *CC*, en cuanto a que ambos subestiman o sobreestiman la prevalencia verdadera cuando la distribución de clases cambia entre los datos de entrenamiento y de prueba.

*Ejemplo:* Aquí debemos sumar las salidas del clasificador para la clase positiva. Para el clasificador del ejemplo, se obtuvieron las siguientes salidas:

$s(\mathbf{x}_i, y = 1)$	0.76	0.60	0.24	0.19	0.53	0.53	0.99	0.66	0.72	0.60	0.53
	0.59	0.47	0.47	0.49	0.46	0.60	0.52	0.46	0.53	0.38	

es decir, que tenemos  $\hat{p}_{tst}^{PCC}(c = 1) \approx 0.54$

## Clasificar, Contar y Ajustar Probabilístico (PACC)

Presentado como *Probabilistic Adjusted Classify and Count* o también como *Probabilistic Adjusted Count*, este método combina las ideas de *ACC* y de *PCC*.

$$\begin{aligned}
 \hat{p}_{tst}^{PCC}(c = c_k) &= \hat{\mathbb{E}}[\mathbb{P}_{tst}(h(\mathbf{x}) = c_k)] \\
 &= \hat{\mathbb{E}}\left[\sum_{j=1}^{\#C} \mathbb{P}(h(\mathbf{x}) = c_k | y = c_j) p_{tst}^{PACC}(c = c_j)\right] \\
 &= \sum_{j=1}^{\#C} \hat{\mathbb{E}}[\mathbb{P}(h(\mathbf{x}) = c_k | y = c_j) p_{tst}^{PACC}(c = c_j)] \\
 &= \sum_{j=1}^{\#C} \hat{\mathbb{E}}[\mathbb{P}(h(\mathbf{x}) = c_k | y = c_j)] \hat{p}_{tst}^{PACC}(c = c_j) \\
 &= \sum_{j=1}^{\#C} \left[ \frac{1}{\#U_j} \sum_{\mathbf{x} \in U_j} \hat{\mathbb{P}}(h(\mathbf{x}) = c_k) \right] \hat{p}_{tst}^{PACC}(c = c_j)
 \end{aligned} \tag{2.5}$$

donde  $U_j = \{(\mathbf{x}, y) \in (\mathbf{X}_{tst}, Y_{tst}) | y = c_j\}$ . Luego,  $\hat{p}_{tst}^{PCC}(c = c_k)$  se calcula mediante *PCC* y, como en *ACC*, las  $[\frac{1}{\#U_j} \sum_{\mathbf{x} \in U_j} \hat{\mathbb{P}}(h(\mathbf{x}) = c_k)]$  deben estimarse mediante validación cruzada o con un conjunto de validación aparte, quedando nuevamente un sistema de ecuaciones lineales de  $\#C$  ecuaciones y  $\#C$  incógnitas.

Para el caso particular binario, y relacionando con 2.2, tenemos:

$$\hat{p}_{tst}^{PACC}(c = 1) = \frac{\hat{p}_{tst}^{PCC}(c = 1) - fp_{pa}}{tp_{pa} - fp_{pa}} \tag{2.6}$$

donde  $tp_{pa}$  y  $fp_{pa}$  (*pa*: *probability average*) son los dos parámetros propios del cuantificador a estimar mediante validación cruzada o con un conjunto de validación aparte, siendo  $tp_{pa}$  el promedio de las probabilidades *a posteriori* para la clase positiva estimadas por el clasificador correspondientes a los individuos cuya etiqueta es positiva, y del mismo modo  $fp_{pa}$  pero para individuos con etiqueta negativa. En este método hay que tener en cuenta ambas consideraciones sobre las estimaciones de  $\hat{p}$  dentro del rango  $[0, 1]$  y sobre la calibración -ver A-.

*Ejemplo:* Del mismo modo que para el ejemplo de *ACC*, aunque deberíamos usar una muestra de validación para calcular  $tp_{pa}$  y  $fp_{pa}$ , aquí también a modo de simplificación usaremos la muestra de prueba para ello. Teniendo en cuenta entonces ahora tanto las salidas del clasificador como las etiquetas de la muestra, tenemos:



$s(\mathbf{x}_i, y = 0)$	$s(\mathbf{x}_i, y = 1)$	$c$
0.24	0.76	1
0.40	0.60	1
0.76	0.24	1
0.81	0.19	1
0.47	0.53	1
0.47	0.53	1
0.01	0.99	1
0.34	0.66	1
0.28	0.72	1
0.40	0.60	1
0.47	0.53	1
0.41	0.59	1
0.53	0.47	1
0.53	0.47	1
0.51	0.49	1
0.54	0.46	1
0.40	0.60	1
0.48	0.52	1
0.54	0.46	1
0.47	0.53	0
0.62	0.38	0

siendo entonces  $tp_{pa} \approx 0.55$  y  $fp_{pa} \approx 0.46$ , por lo que  $\hat{p}_{tst}^{PACC}(c = 1) \approx \frac{0.54-0.46}{0.55-0.46} \approx 0.9$

### 2.1.2. Con clasificadores específicos

## 2.2. Métodos No Agregativos



## Capítulo 3

# Estimación por Intervalos



## Capítulo 4

# Métodos de Evaluación

La evaluación de métodos de cuantificación es más compleja que en otros problemas. En aprendizaje supervisado, típicamente se mide el rendimiento estimando la probabilidad de predecir correctamente ejemplos individuales no observados (sin condicionar -para la exactitud o *accuracy*- o condicionando las probabilidades a las clases de pertenencia -para la exhaustividad o *recall*- o predichas -para la precisión o *precision*-). Sin embargo, en cuantificación, el rendimiento se evalúa para conjuntos de datos. Esto implica que necesitamos una colección de muestras para evaluar el rendimiento de un modelo. Dado un modelo  $\bar{h}$ , una función de pérdida  $L(\cdot, \cdot)$ , y un conjunto de muestras de evaluación  $T_1, \dots, T_s$ , el rendimiento de  $\bar{h}$  es:

$$\text{Rendimiento}(\bar{h}, L, T_1, \dots, T_s) = \frac{1}{s} \sum_{j=1}^s L(\bar{h}, T_j) \quad (4.1)$$

Calcular la pérdida de un modelo sobre una muestra de prueba,  $L(\bar{h}, T_j)$  no requiere de promediar sobre ejemplos individuales. Por ejemplo, en la cuantificación binaria, sólo la prevalencia real  $p$  y la prevalencia predicha  $\hat{p}$  se comparan por cada muestra.

En cuantificación, el problema de evaluación se relaciona con el cambio en la distribución de datos entre la fase de entrenamiento y la de implementación del modelo. Se requiere una colección de muestras de prueba variada y que represente diversas distribuciones para evaluar correctamente el rendimiento del modelo y evitar sesgos. Por esta razón, la mayoría de los experimentos reportados en la literatura emplean conjuntos de datos tomados de otros problemas y se crean conjuntos de prueba con cambios en las distribuciones creados artificialmente. Este enfoque tiene la ventaja de que la cantidad del *dataset shift* se puede controlar para estudiar el rendimiento de los modelos en diferentes situaciones.

Las funciones de pérdida  $L(\cdot, \cdot)$  serán elegidas de acuerdo al tipo de problema y al objetivo particular de la aplicación. Como ya se mencionó, el rendimiento de  $\bar{h}$  será el promedio del resultado de la función de pérdida por cada muestra de evaluación, de acuerdo a la ecuación 4.1. Se han propuesto en la literatura distintas métricas de evaluación para problemas de *Single-Label Quantification (SLQ)*. Estas también se pueden usar para *Binary Quantification (BQ)*, ya que es un caso espacial del anterior, y para *Multi-Label Quantification*, ya que se pueden usar para cada  $y \in C$ . Esencialmente todas las medidas de evaluación que se han propuesto son divergencias, es decir, medidas de cómo una distribución difiere de otra. No se desarrollarán en esta tesis métricas para *Ordinal Quantification* ni para *Regression Quantification*, ya que no son útiles para nuestro

objeto de estudio.

## 4.1. Propiedades

Sebastiani [25] define una serie de propiedades interesantes para medidas de evaluación en *SLQ*. Un importante resultado de este artículo es que ninguna medida de evaluación existente para *SLQ* satisface todas las propiedades identificadas como deseables; aún así, se ha demostrado que algunas medidas de evaluación son “menos inadecuadas” que otras. Aquí mencionamos brevemente las cuatro propiedades principales que habría que considerar en cada métrica  $M$  a emplear (el resto son propiedades que suelen ser satisfechas por todas las métricas).

- **Máximo (MAX)**: si  $\exists \beta > 0, \beta \in \mathbb{R}$  tal que por cada  $c \in C$  y por cada  $p$ , (i) existe  $\hat{p}$  tal que  $M(p, \hat{p}) = \beta$ , y (ii) para ninguna  $\hat{p}$  se cumple que  $M(p, \hat{p}) > \beta$ . Si se cumple **MAX**, la imagen de  $M$  es independiente del problema, y esto permite juzgar si un valor dado significa un error de cuantificación alto o bajo. Si  $M$  no cumple **MAX**, cada muestra de evaluación tendrá un peso distinto en el resultado final.
- **Imparcial (IMP)**: si  $M$  penaliza igualmente la subestimación de  $p$  por una cantidad  $a$  (es decir, con  $\hat{p} = p - a$ ) o su sobreestimación por la misma cantidad  $a$  (es decir, con  $\hat{p} = p + a$ ). Si se cumple **IMP**, la subestimación y la sobreestimación se consideran igualmente indeseables. Esto es generalmente lo deseable, a menos que exista una razón específica para no hacerlo.
- **Relativo (REL)**: si  $M$  penaliza más gravemente un error de magnitud absoluta  $a$  (es decir, cuando  $\hat{p} = p \pm a$ ) si  $p$  es menor. Por ejemplo, predecir  $\hat{p} = 0.0101$  cuando  $p = 0.0001$  es un error mucho más serio que predecir  $\hat{p} = 0.1100$  cuando  $p = 0.1000$ .
- **Absoluto (ABS)**: si  $M$  penaliza un error de magnitud independientemente del valor de  $p$ . Mientras algunas aplicaciones requieren **REL**, otras requieren **ABS**. Si bien **REL** y **ABS** son mutuamente excluyentes, ninguna cubre el caso cuando  $M$  considera un error de magnitud absoluta  $a$  menos grave cuando  $p$  es menor (como en el caso de la *distancia coseno*).

## 4.2. Métricas

### 4.2.1. Sesgo

El sesgo o *bias* técnicamente no es una medida de evaluación para la cuantificación, ya que no se aplica a toda una distribución  $p$  sino solo a una clase específica  $c \in C$ , y se define como:

$$B(c) = \hat{p}(c) - p(c) \quad (4.2)$$

Incluso usado en cuantificación binaria, se debe especificar a cuál de las clases hacer referencia (en este caso, suele hacer referencia a la clase positiva). Si se usa como una medida de evaluación para la cuantificación, un problema con  $B$  es que promediar los puntajes de diferentes clases produce resultados poco intuitivos, ya que el sesgo positivo de una clase y el sesgo negativo de otra clase se anulan entre sí. El mismo problema ocurre cuando se trata de la misma clase pero se promedia entre diferentes muestras. Como resultado, esta medida se puede utilizar como mucho para determinar

si un método tiene una tendencia a subestimar o sobrestimar la prevalencia de una clase específica (típicamente la clase minoritaria) en  $BQ$ , y no como una medida de evaluación general para usar.

#### 4.2.2. Error Absoluto

El error absoluto o *absolute error* es una de las medidas más empleadas ya que, al ser simplemente la diferencia entre ambas magnitudes, es simple y fácilmente interpretable.

$$AE(p, \hat{p}) = \frac{1}{\#C} \sum_{j=1}^{\#C} |\hat{p}(c = c_j) - p(c = c_j)| \quad (4.3)$$

Como en este caso las diferencias positivas y negativas son igualmente indeseables, promediar el AE entre varias clases, o varias muestras, no es problemático. Como se muestra en [25], AE cumple **IMP** y **ABS** pero no cumple **MAX** (ni tampoco **REL**). Su rango va de 0 (mejor) a:

$$z_{AE} = \frac{2(1 - \min_{j \in \{1, \dots, \#C\}} p(c = c_j))}{\#C} \quad (4.4)$$

(peor), por lo que su rango depende de la distribución de  $p$  y de  $\#C$ .

#### 4.2.3. Error Absoluto Normalizado

El error absoluto normalizado *normalised absolute error*, definido como:

$$NAE(p, \hat{p}) = \frac{AE(p, \hat{p})}{z_{AE}} = \frac{\sum_{j=1}^{\#C} |\hat{p}(c = c_j) - p(c = c_j)|}{2(1 - \min_{j \in \{1, \dots, \#C\}} p(c = c_j))} \quad (4.5)$$

es una versión de AE que oscila entre 0 (mejor) y 1 (peor), por lo que cumple **MAX**. A pesar de su nombre, NAE no disfruta de **ABS** (ni tampoco **REL**).

#### 4.2.4. Error Cuadrático

El error cuadrático o *squared error*, definido como:

$$SE(p, \hat{p}) = \frac{1}{\#C} \sum_{j=1}^{\#C} (\hat{p}(c = c_j) - p(c = c_j))^2 \quad (4.6)$$

comparte los mismos pros y contras de AE, pero penalizando más cuanto mayor es la diferencia entre el valor real y el predicho, por lo que se usa cuando se quiere castigar los valores atípicos u *outliers*.

#### 4.2.5. Error Absoluto Relativo

El error absoluto relativo o *relative absolute error* es una adaptación del AE que impone **REL** al hacer que AE sea relativo a  $p$ .

$$RAE(p, \hat{p}) = \frac{1}{\#C} \sum_{j=1}^{\#C} \frac{|\hat{p}(c = c_j) - p(c = c_j)|}{p(c = c_j)} \quad (4.7)$$

RAE cumple **IMP** y **REL** pero no cumple **MAX** (ni **ABS**, a pesar de su nombre). Su rango va de 0 (mejor) a:

$$z_{\text{RAE}} = \frac{\#C - 1 + \frac{1 - \min_{j \in \{1, \dots, \#C\}} p(c = c_j)}{\min_{j \in \{1, \dots, \#C\}} p(c = c_j)}}{\#C} \quad (4.8)$$

(peor), por lo que su rango depende de la distribución de  $p$  y de  $\#C$ .

#### 4.2.6. Error Absoluto Relativo Normalizado

El error absoluto relativo normalizado *normalised relative absolute error*, definido como:

$$\text{NRAE}(p, \hat{p}) = \frac{\text{RAE}(p, \hat{p})}{z_{\text{RAE}}} = \frac{\sum_{j=1}^{\#C} \frac{|\hat{p}(c=c_j) - p(c=c_j)|}{p(c=c_j)}}{1 - \frac{\min_{j \in \{1, \dots, \#C\}} p(c=c_j)}{\min_{j \in \{1, \dots, \#C\}} p(c=c_j)}} \quad (4.9)$$

es una versión de RAE que oscila entre 0 (mejor) y 1 (peor), por lo que cumple **MAX**. A pesar de su nombre, NRAE no disfruta de **REL** (ni tampoco **ABS**).

Tanto RAE como NRAE no están definidas cuando sus denominadores sean nulos. Para resolver este problema, se puede suavizar tanto  $p(c = c_j)$  como  $\hat{p}(c = c_j)$  mediante suavizado aditivo:

$$\underline{p}(c = c_j) = \frac{\epsilon + p(c = c_j)}{\epsilon \#C + \sum_{j=1}^{\#C} p(c = c_j)} \quad (4.10)$$

donde  $\underline{p}(c = c_j)$  es la versión suavizada de  $p(c = c_j)$  y el denominador es solo un factor de normalización (lo mismo para  $\underline{\hat{p}}(c = c_j)$ ).

#### 4.2.7. Divergencia de Kullback-Leibler

Para distribuciones de probabilidad discretas  $P$  y  $Q$  definidas en el mismo espacio muestral  $\mathcal{X}$  su divergencia KL se define como:

$$\text{DKL}(P \parallel Q) = \sum_{x \in \mathcal{X}} P(x) \log \left( \frac{P(x)}{Q(x)} \right) \quad (4.11)$$

En cuantificación, se quiere comparar la prevalencia real  $p$  y la prevalencia predicha  $\hat{p}$ , y el espacio muestral corresponde a las posibles clases, con lo cuál será:

$$\text{DKL}(p \parallel \hat{p}) = \sum_{j=1}^{\#C} p(c = c_j) \log \left( \frac{p(c = c_j)}{\hat{p}(c = c_j)} \right) \quad (4.12)$$

que va de 0 (mejor) a  $+\infty$  (peor) -por lo tanto, no cumple con **MAX**-. Si bien esta medida es una distancia, no es una métrica verdadera, ya que no obedece a la desigualdad del triángulo y no es simétrica. Además, es menos interpretable que otras métricas de rendimiento y no está definido cuando  $\hat{p}$  es 0 o 1.



#### 4.2.8. Divergencia de Kullback-Leibler Normalizada

Para suplir los problemas de DKL, se puede utilizar la función logística, quedando:

$$\text{NDKL}(p \parallel \hat{p}) = 2 \cdot \frac{e^{\text{DKL}(p \parallel \hat{p})}}{1 + e^{\text{DKL}(p \parallel \hat{p})}} - 1 \quad (4.13)$$

que también va de 0 (mejor) a  $+\infty$  (peor) -por lo tanto, si cumple con **MAX**-. Sin embargo, como se muestra en [25], ni DKL ni NDKL cumplen con **IMP**, **REL** y **ABS**, lo que hace que su uso como medidas de evaluación para cuantificación sea cuestionable, además de ser difíciles de interpretar.

### 4.3. Elección de la Métrica

Es evidente que ninguna de las medidas propuestas hasta ahora es completamente satisfactoria. DKL y NDKL son los menos satisfactorios y parecen fuera de discusión. Respecto a los demás, el problema es que **MAX** parece ser incompatible con **REL/ABS**, y viceversa.

Sebastiani [25] sostiene que cumplir con **REL** o **ABS** parece más importante que cumplir con **MAX**, ya que reflejan las necesidades de la aplicación; si no se satisfacen estas propiedades, se puede argumentar que el error de cuantificación que se está midiendo está vagamente relacionado a lo que el usuario realmente quiere. Si **MAX** no está satisfecho, los resultados obtenidos en muestras caracterizadas por diferentes distribuciones no serán comparables. A pesar de esto, los resultados obtenidos por diferentes sistemas en el mismo conjunto de muestras siguen siendo comparables.

Esto sugiere que AE, RAE y SE son las mejores medidas a elegir. Se debe preferir AE cuando un error de estimación de una magnitud absoluta dada debe considerarse más grave cuando la verdadera prevalencia de la clase afectada es menor. RAE debe ser elegido cuando un error de estimación de una magnitud absoluta dada tiene el mismo impacto independientemente de la verdadera prevalencia de la clase afectada. Si se quiere penalizar mayormente errores atípicos, considerando mucho más graves a los errores cuanto mayor es la diferencia entre el valor real y el predicho, entonces SE es la métrica más conveniente.

### 4.4. Protocolos

Mientras que en la clasificación, un conjunto de datos de tamaño  $k$  proporciona  $k$  puntos de evaluación, para la cuantificación, el mismo conjunto solo proporciona 1 punto. Evaluar algoritmos de cuantificación es por lo tanto un reto, debido a que la disponibilidad de datos etiquetados con fines de prueba es más restringido. Hay principalmente dos protocolos experimentales que se han tomado para tratar con este problema: el Protocolo de Prevalencia Natural (*NPP*) y el Protocolo de Prevalencia Artificial (*APP*).

- *NPP*: Consiste en, una vez entrenado un cuantificador, tomar un conjunto de prueba (no observado en el entrenamiento) lo suficientemente grande, dividirlo en un número de muestras de manera uniformemente aleatoria, y llevar a cabo la evaluación individualmente en cada muestra.
- *APP*: Consiste en, previo al entrenamiento, tomar un conjunto de datos, dividirlo en un conjunto de entrenamiento y en un conjunto de evaluación de manera aleatoria, y realizar

experimentos repetidos en los que la prevalencia del conjunto de entrenamiento o la prevalencia del conjunto de prueba de una clase se varía artificialmente a través del submuestreo.

Ambos protocolos tienen diferentes pros y contras. Una ventaja de *APP* es que permite crear muchos puntos de prueba de la misma muestra. Además, *APP* permite simular distintos *Prior probability shift*, mientras que con *NPP* se estaría evaluando sólo con las distribuciones originales de los datos de entrenamiento y prueba. Sin embargo, una desventaja de *APP* es que puede no saberse cuán realistas son estas diferentes situaciones en la aplicación real, por lo que se podría estar destinando recursos a una evaluación errónea o pobre. Una solución intermedia podría ser utilizar un protocolo que utilice conocimientos previos sobre la distribución de prevalencias “probables” que se podría esperar encontrar en el dominio específico en cuestión.

## Capítulo 5

# Resultados



# Apéndice A

## Calibración

En problemas de clasificación, el subproblema de la predicción de estimaciones de probabilidad representativas de las probabilidades verdaderas es conocido como calibración. En los sistemas del mundo real, los clasificadores no sólo deben ser precisos, sino que también deben indicar cuando es probable que sean incorrectos. Es decir, deben estimar su nivel de incertidumbre o confiabilidad. En otras palabras, las probabilidades asociadas con la etiquetas de clase predichas deben reflejar su verosimilitud real.

Uno de los casos en donde se debe contemplar este problema es en la toma de decisiones (es decir, en casi toda aplicación real). Por ejemplo, en sistemas utilizados para la salud, un diagnóstico con un bajo nivel de confianza puede significar realizar otro tipo de chequeo. A su vez, las estimaciones de probabilidad, pueden ser utilizadas para ser incorporadas en otro modelo probabilístico. Por ejemplo, se podrían combinar distintas salidas de distintos modelos de forma ponderada para obtener una predicción más robusta frente los casos en donde cada modelo individual falla.

La mayoría de los métodos de aprendizaje supervisado producen clasificadores que generan puntuaciones  $s(x)$  que se pueden utilizar para ranquear los ejemplos en el conjunto de pruebas de la etiqueta más probable a la menos probable de una clase  $c$ . Es decir, para dos ejemplos  $\mathbf{x}_1$  y  $\mathbf{x}_2$ , si  $s(\mathbf{x}_1) < s(\mathbf{x}_2)$  entonces  $\mathbb{P}(c|x) < \mathbb{P}(c|y)$ . Sin embargo, la clasificación según el rango de la probabilidad de pertenencia a una clase no es suficiente. Lo que se necesita es una estimación precisa de la probabilidad de que cada ejemplo de prueba sea un miembro de la clase de interés.

El modelo de regresión logística es un caso especial ya que está bien calibrado por diseño dado que su función objetivo minimiza la función de pérdida logarítmica o *log-loss* [26]. Sin embargo, si no se cuenta con un conjunto de entrenamiento lo suficientemente grande, es posible que el modelo no tenga suficiente información para calibrar las probabilidades.

Otros modelos, en cambio, no presentan esta propiedad (por ejemplo, los clasificadores de Naive Bayes, Random Forest o redes neuronales [27–29]). Incluso, hay modelos que no devuelven probabilidades *a posteriori*, sino genéricas puntuaciones de confianza, como es el caso de SVM [30]. En estos dos últimos casos es posible mapear las salidas de los clasificadores a probabilidades *a posteriori* calibradas a través de algunos método de calibración [27–30].

**A.1. Definición**

**A.2. Diagramas de confiabilidad**

**A.3. Métodos de Evaluación**

**A.4. Métodos de Calibración**

**A.4.1. Modelos Binarios**

**A.4.2. Modelos Multiclase**

# Referencias

- [1] George Forman. Counting positives accurately despite inaccurate classification. In *European conference on machine learning*, pages 564–575. Springer, 2005.
- [2] David D Lewis. Evaluating and optimizing autonomous text classification systems. In *Proceedings of the 18th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 246–254, 1995.
- [3] Jose G Moreno-Torres, Troy Raeder, Rocío Alaiz-Rodríguez, Nitesh V Chawla, and Francisco Herrera. A unifying view on dataset shift in classification. *Pattern recognition*, 45(1):521–530, 2012.
- [4] Amos Storkey et al. When training and test sets are different: characterizing learning transfer. *Dataset shift in machine learning*, 30(3-28):6, 2009.
- [5] Rocío Alaíz-Rodríguez, Alicia Guerrero-Curieses, and Jesús Cid-Sueiro. Class and subclass probability re-estimation to adapt a classifier in the presence of concept drift. *Neurocomputing*, 74(16):2614–2623, 2011.
- [6] Marthinus Christoffel Du Plessis and Masashi Sugiyama. Class prior estimation from positive and unlabeled data. *IEICE TRANSACTIONS on Information and Systems*, 97(5):1358–1362, 2014.
- [7] Yee Seng Chan and Hwee Tou Ng. Estimating class priors in domain adaptation for word sense disambiguation. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 89–96, 2006.
- [8] Zhihao Zhang and Jie Zhou. Transfer estimation of evolving class priors in data stream classification. *Pattern Recognition*, 43(9):3151–3161, 2010.
- [9] Marthinus Christoffel Du Plessis and Masashi Sugiyama. Semi-supervised learning of class balance under class-prior change by distribution matching. *Neural Networks*, 50:110–119, 2014.
- [10] Jose Barranquero, Pablo González, Jorge Díez, and Juan José Del Coz. On the study of nearest neighbor algorithms for prevalence estimation in binary problems. *Pattern Recognition*, 46(2):472–482, 2013.
- [11] Hideki Asoh, Kazushi Ikeda, and Chihiro Ono. A fast and simple method for profiling a population of twitter users. In *The Third International Workshop on Mining Ubiquitous and Social Environments*, page 19. Citeseer, 2012.

- [12] Víctor González-Castro, Rocío Alaiz-Rodríguez, and Enrique Alegre. Class distribution estimation based on the Hellinger distance. *Information Sciences*, 218:146–164, 2013.
- [13] Nachai Limsetto and Kitsana Waiyamai. Handling concept drift via ensemble and class distribution estimation technique. In *Advanced Data Mining and Applications: 7th International Conference, ADMA 2011, Beijing, China, December 17-19, 2011, Proceedings, Part II* 7, pages 13–26. Springer, 2011.
- [14] Jack Chongjie Xue and Gary M Weiss. Quantification and semi-supervised classification methods for handling changes in class distribution. In *Proceedings of the 15th ACM SIGKDD international conference on knowledge discovery and data mining*, pages 897–906, 2009.
- [15] Afonso Fernandes Vaz, Rafael Izbicki, and Rafael Bassi Stern. Quantification under prior probability shift: The ratio estimator and its extensions. *The Journal of Machine Learning Research*, 20(1):2921–2953, 2019.
- [16] Andrea Esuli, Alessandro Fabris, Alejandro Moreo, and Fabrizio Sebastiani. *Learning to Quantify*. Springer Nature, 2023.
- [17] Pablo González, Alberto Castaño, Nitesh V Chawla, and Juan José Del Coz. A review on quantification learning. *ACM Computing Surveys (CSUR)*, 50(5):1–40, 2017.
- [18] George Forman. Quantifying counts and costs via classification. *Data Mining and Knowledge Discovery*, 17:164–206, 2008.
- [19] Vladimir N Vapnik. An overview of statistical learning theory. *IEEE transactions on neural networks*, 10(5):988–999, 1999.
- [20] Marco Saerens, Patrice Latinne, and Christine Decaestecker. Adjusting the outputs of a classifier to new a priori probabilities: a simple procedure. *Neural computation*, 14(1):21–41, 2002.
- [21] Arpita Biswas and Suvam Mukherjee. Ensuring fairness under prior probability shifts. In *Proceedings of the 2021 AAAI/ACM Conference on AI, Ethics, and Society*, pages 414–424, 2021.
- [22] Antonio Bella, Cesar Ferri, José Hernández-Orallo, and Maria Jose Ramirez-Quintana. Quantification via probability estimators. In *2010 IEEE International Conference on Data Mining*, pages 737–742. IEEE, 2010.
- [23] Lei Tang, Huiji Gao, and Huan Liu. Network quantification despite biased labels. In *Proceedings of the Eighth Workshop on Mining and Learning with Graphs*, pages 147–154, 2010.
- [24] Dirk Tasche. Exact fit of simple finite mixture models. *Journal of Risk and Financial Management*, 7(4):150–164, 2014.
- [25] Fabrizio Sebastiani. Evaluation measures for quantification: An axiomatic approach. *Information Retrieval Journal*, 23(3):255–288, 2020.
- [26] Geoffrey Stewart Morrison. Tutorial on logistic-regression calibration and fusion: converting a score to a likelihood ratio. *Australian Journal of Forensic Sciences*, 45(2):173–197, 2013.



- [27] Bianca Zadrozny and Charles Elkan. Transforming classifier scores into accurate multiclass probability estimates. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 694–699, 2002.
- [28] Alexandru Niculescu-Mizil and Rich Caruana. Predicting good probabilities with supervised learning. In *Proceedings of the 22nd international conference on Machine learning*, pages 625–632, 2005.
- [29] Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger. On calibration of modern neural networks. In *International conference on machine learning*, pages 1321–1330. PMLR, 2017.
- [30] John Platt et al. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. *Advances in large margin classifiers*, 10(3):61–74, 1999.