A Tutorial on Quantification

Predicting Class Frequencies via Supervised Learning

Mirko Bunse (1) **Alejandro Moreo** (2), and Fabrizio Sebastiani (2)

(1) LAMARR Institute for Machine Learning and Artificial Intelligence
(2) Istituto di Scienza e Tecnologie dell'Informazione
Consiglio Nazionale delle Ricerche

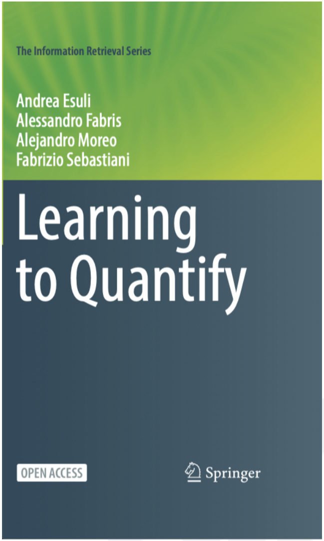
Learning to Quantify: Methods and Applications (LQ 2024)
@ECML-PKDD, Vilnius, Lithuania
September 13, 2024

Structure of this tutorial

1. Introduction
2. Applications of quantification in ML, DM, NLP
3. Evaluation measures and evaluation protocols for quantification
4. Supervised learning methods for quantification
5. QuaPy: an Open-source library for quantification
6. Advanced topics
7. Conclusions

What this tutorial is about

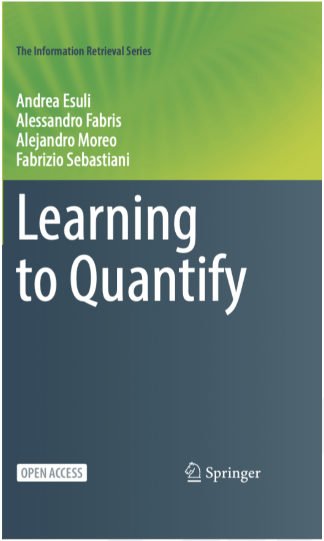
- A tutorial about a special type of application contexts for supervised learning technologies in which our interest is not at the individual level but at the aggregate level
- Many fields of human activity like:
  - social sciences,
  - political science
  - epidemiology
  - ecological modelling
  - market research
- ... don't care about individuals but about populations; in other words don't care about the needle, but about the haystack
- Still a fairly unknown task among potential users



Andrea Esuli, Alessandro Fabris, Alejandro Moreo, Fabrizio Sebastiani. Learning to Quantify. Springer Nature, 2023. Download for free at https://bit.ly/3JgEMJO

What this tutorial is about

- Learning to Quantify (aka quantification) stands to classification as aggregate data stand to individual data
- This tutorial is an introduction to this field, to its applications, to the methods for performing quantification, and to the methods for evaluating quantification systems



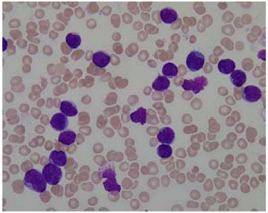
Andrea Esuli, Alessandro Fabris, Alejandro Moreo, Fabrizio Sebastiani. Learning to Quantify. Springer Nature, 2023. Download for free at https://bit.ly/3JgEMJO

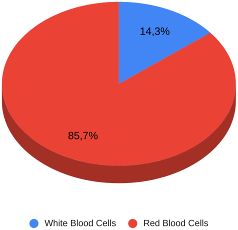
Example applications of quantification

- In many applications of classification, the real goal is determining the relative frequency of each class in the unlabelled data.

Example applications of quantification

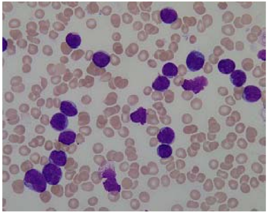
- In many applications of classification, the real goal is determining the relative frequency of each class in the unlabelled data.

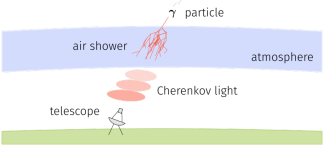


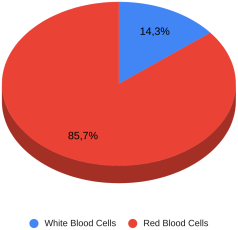


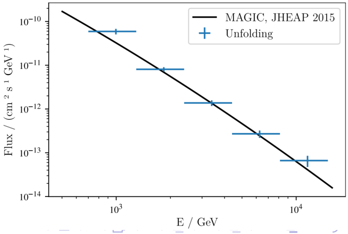
Example applications of quantification

- In many applications of classification, the real goal is determining the relative frequency of each class in the unlabelled data.









What is "Learning to Quantify" (a.k.a. quantification)?

- The task is of independent interest in statistics and data mining, while it is often only functional to generating better classifiers, or to performing other downstream ML tasks
- Studied in different fields like ML, DM, NLP; different terminology:
  - Quantification: "learning to quantify", "supervised prevalence estimation", "class prior estimation", "prior estimation", "class distribution estimation", ...
  - Relative frequencies: "prevalence values", "class priors", "priors", "class fractions", "class percentages", ...
- A fully supervised task, but is an "asymmetric" task, since training examples are individual labelled items and test examples are samples of individual unlabelled items

(Binary) Task	Model	Training		Test		Type
		Examples	Labels	Examples	Labels	
Classification	$h : \mathcal{X} \rightarrow \{+1, -1\}$	Individual items	Classes	Individual items	Classes	Symmetric
Regression	$h : \mathcal{X} \rightarrow \mathbb{R}$	Individual items	Real values	Individual items	Real values	Symmetric
Learning from Label Proportions	$h : 2^{\mathcal{X}} \rightarrow [0, 1]$	Samples of individual items	Real values in [0,1]	Samples of individual items	Real values in [0,1]	Symmetric
Quantification	$h : 2^{\mathcal{X}} \rightarrow [0, 1]$	Individual items	Classes	Samples of individual items	Real values in [0,1]	Asymmetric

What is "Learning to Quantify" (a.k.a. quantification)?

- A task "simpler" (i.e., less general) than classification
  - Vapnik's principle : *If you possess a restricted amount of information for solving some problem, try to solve the problem directly and never solve a more general problem as an intermediate step. It is possible that the available information is sufficient for a direct solution but is insufficient for solving a more general intermediate problem.*
- Quantification is an independent task on its own right, with dedicated:
  - learning methods
  - evaluation measures
  - experimentation protocols

The suboptimality of "Classify and Count"

- Quantification can trivially be solved via the Classify and Count (CC) method:
  1. Train a classifier
  2. Classify all the unlabelled data items in the sample
  3. For each class, count how many unlabelled data items have been attributed to the class
  4. Divide each count by the total number of unlabelled data items
- However, CC proves a suboptimal quantification method, for two main independent reasons:
  1. Classifier bias
  2. Presence of dataset shift

1. CC delivers suboptimal results under classifier bias

- Even if it relies on a good classifier, CC is not necessarily a good quantifier:

	$y = 1$	$y = 0$
$\hat{y} = 1$	TP	FP
$\hat{y} = 0$	FN	TN

h_1	$y = 1$	$y = 0$
$\hat{y} = 1$	95	20
$\hat{y} = 0$	5	480

h_2	$y = 1$	$y = 0$
$\hat{y} = 1$	70	30
$\hat{y} = 0$	30	470

#ActualPositives = 100 (16.7%)
#ActualNegatives = 500 (83.3%)
#Instances = 600

#Errors=25, Accuracy=96%
#PredictedPositives=115 (19.1%)
#ActualPositives=100 (16.7%)

#Errors=60, Accuracy=90%
#PredictedPositives=100 (16.7%)
#ActualPositives=100 (16.7%)

- Paradoxically, for quantification purposes, we should base CC on h_2 rather than on h_1
- Different goals:
  - Classification: minimize $(FN + FP)$
  - Quantification: minimize $|FN - FP|$

2. CC delivers suboptimal results under dataset shift

- Even if it relies on a classifier trained to optimize error balancing, CC may deliver suboptimal results in the presence of dataset shift (DS – aka dataset "drift")
- DS defined as the case in which $P(X, Y) \neq Q(X, Y)$, i.e., as the case in which the IID assumption does not hold
  - P: the data distribution from which the training data are sampled
  - Q: the data distribution from which the unlabelled (test) data are sampled
- When dataset shift is present, test items are also called out-of-distribution (OOD) data

2. CC delivers suboptimal results under dataset shift

- DS may derive
  - from variations in the environment that the data represent (real shift); i.e. the environment is not stationary, and the operating ("test") conditions were not the same at training time;
    - E.g., prevalence of terrorism-related news before or after 9/11;
  - from the fact that the (training) data misrepresent the environment (virtual shift): i.e., the process of labelling training data may have introduced "sample selection bias":
    - intentionally (e.g., when oversampling the minority class)
    - unintentionally (e.g., if active learning is used)
- CC is suboptimal under DS because CC is usually based on classifiers trained under the IID assumption, which is not verified under DS

2. CC delivers suboptimal results under dataset shift

- The DS literature identifies three main DS types, depending on whether we are in the presence of "$X \rightarrow Y$ problems" (causal learning) or "$Y \rightarrow X$ problems" (anti-causal learning)
- In $X \rightarrow Y$ problems we may write $P(X, Y) = P(Y|X)P(X)$
  - E.g., weather forecasting, avalanche prediction from causes
  - In this case we may have covariate shift, defined as the case in which $P(X) \neq Q(X)$ but $P(Y|X) = Q(Y|X)$
  - E.g., avalanche prediction in different geographical areas
- In $Y \rightarrow X$ problems we may write $P(X, Y) = P(X|Y)P(Y)$
  - E.g., handwritten digit recognition, authorship attribution, predicting illnesses from symptoms
  - In this case we may have prior probability shift (aka "label shift"), defined as the case in which $P(Y) \neq Q(Y)$ but $P(X|Y) = Q(X|Y)$
  - E.g., applying to binary digits a handwritten digit recognizer trained on decimal digits
- We have concept shift if either $P(Y|X) \neq Q(Y|X)$ or $P(X|Y) \neq Q(X|Y)$

Classification VS Quantification

Classification

- Given a labeled training set, learn a classifier

$$h : \mathcal{X} \to \mathcal{Y}$$

- $\hat{y} = h(\mathbf{x})$, where $\mathbf{x} \in \mathcal{X}$ is a feature vector, and $\hat{y} \in \{y_1, \ldots y_n\}$ is a class label
- Error:
  false positives, false negatives

Quantification

- Given a labelled training set, learn a quantifier

$$q : \mathbb{N}^{\mathcal{X}} \to \Delta^{n-1}$$

- $\mathbf{p} = q(\sigma)$, with σ a sample of feature vectors, and $\surd$ a vector of class prevalence values
- Error:
  underestimation, overestimation

Classification VS Quantification

Classification

- Given a labeled training set, learn a classifier

$$h : \mathcal{X} \to \mathcal{Y}$$

- $\hat{y} = h(\mathbf{x})$, where $\mathbf{x} \in \mathcal{X}$ is a feature vector, and $\hat{y} \in \{y_1, \ldots y_n\}$ is a class label
- Error:
  false positives, false negatives

- IID assumption

Quantification

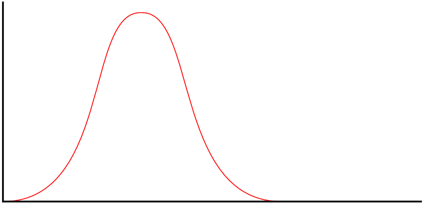
- Given a labelled training set, learn a quantifier

$$q : \mathbb{N}^{\mathcal{X}} \to \Delta^{n-1}$$

- $\mathbf{p} = q(\sigma)$, with σ a sample of feature vectors, and $\surd$ a vector of class prevalence values
- Error:
  underestimation, overestimation

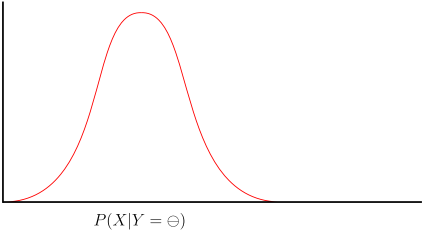
- Prior probability shift (PPS)

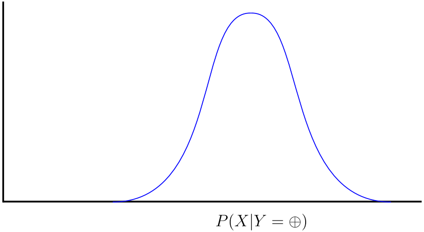
Prior Probability Shift (PPS)



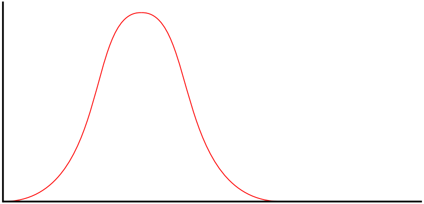
$P(X|Y = \ominus)$

Prior Probability Shift (PPS)



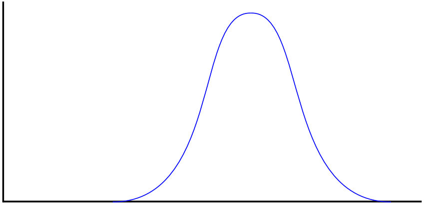


Prior Probability Shift (PPS)



$P(X|Y=\ominus)$

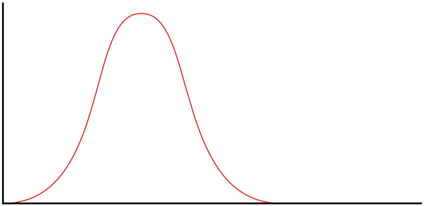
$Q(X|Y=\ominus)$



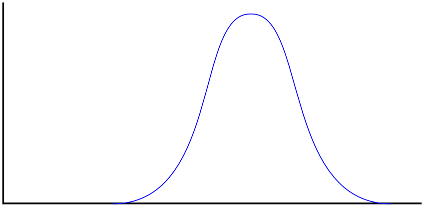
$P(X|Y=\oplus)$

$Q(X|Y=\oplus)$

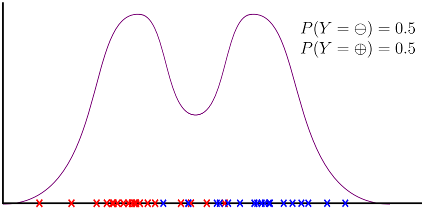
Prior Probability Shift (PPS)



$P(X|Y = \ominus)$
$Q(X|Y = \ominus)$

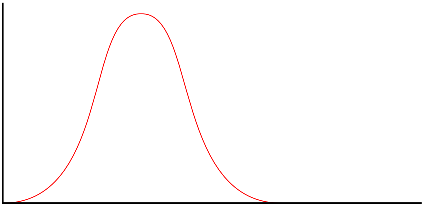


$P(X|Y = \oplus)$
$Q(X|Y = \oplus)$



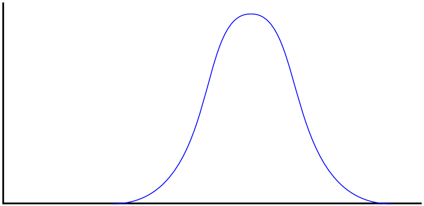
$P(X) = 0.5\ P(X|Y = \ominus) + 0.5\ P(X|Y = \oplus)$

Prior Probability Shift (PPS)



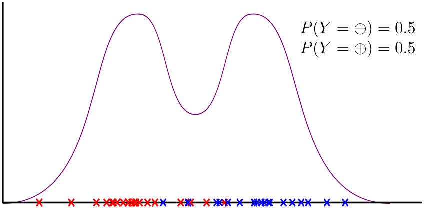
$P(X|Y=\ominus)$

$Q(X|Y=\ominus)$

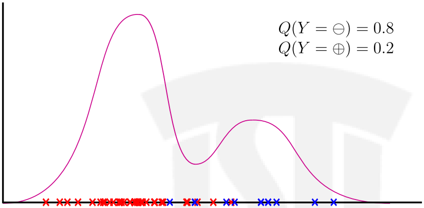


$P(X|Y=\oplus)$

$Q(X|Y=\oplus)$

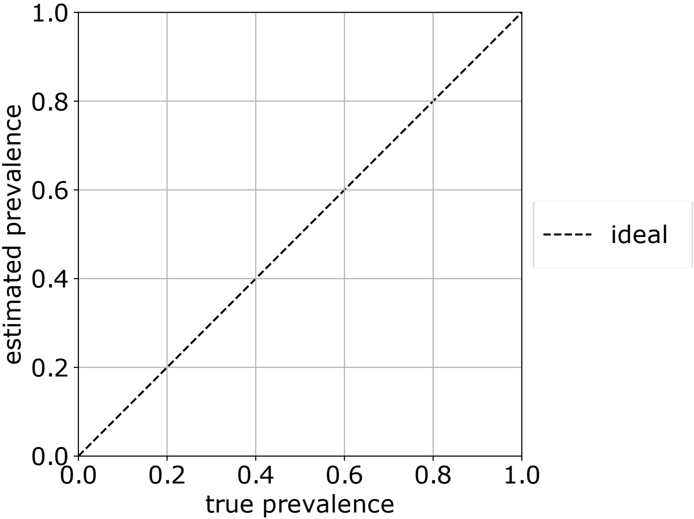


$P(X)=0.5\;P(X|Y=\ominus)+0.5\;P(X|Y=\oplus)$

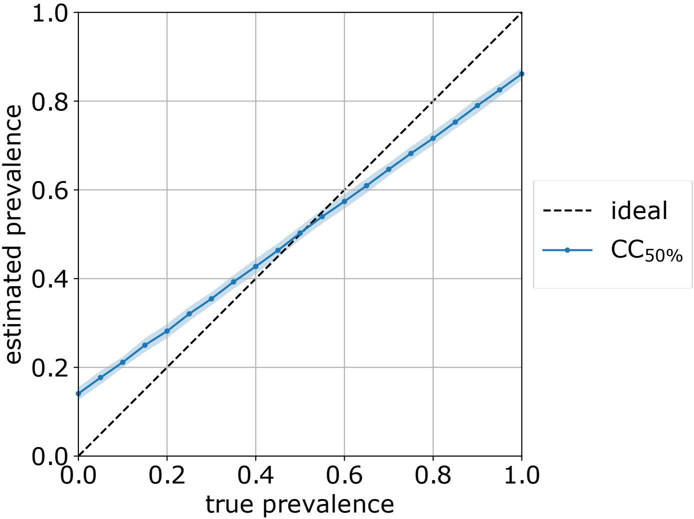


$Q(X)=0.8\;Q(X|Y=\ominus)+0.2\;Q(X|Y=\oplus)$

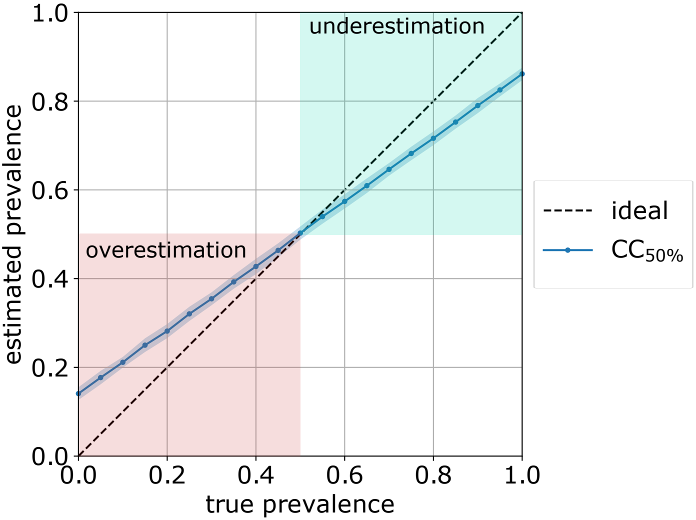
CC delivers biased results



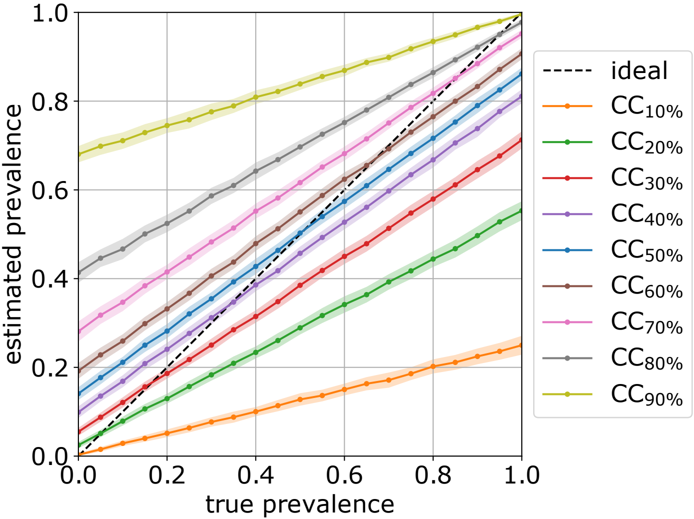
CC delivers biased results



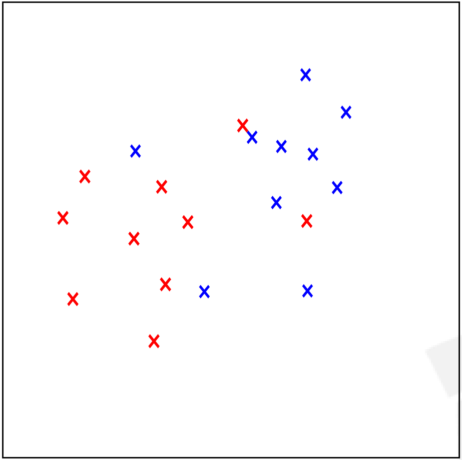
CC delivers biased results



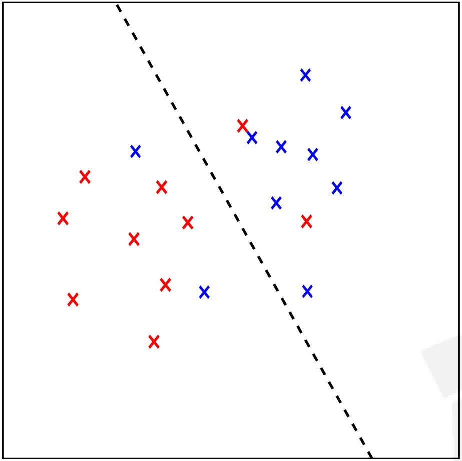
CC delivers biased results



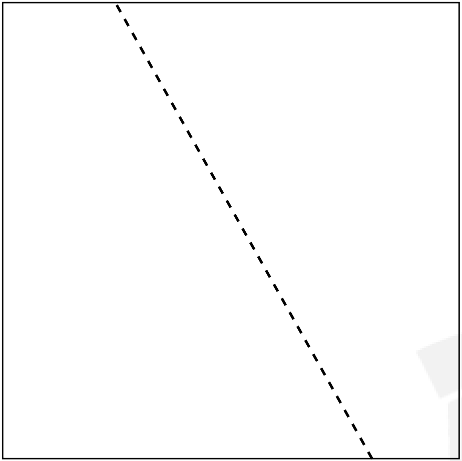
Why does CC deliver biased results?



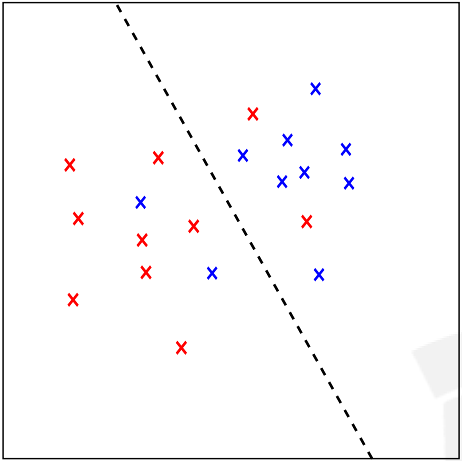
Why does CC deliver biased results?



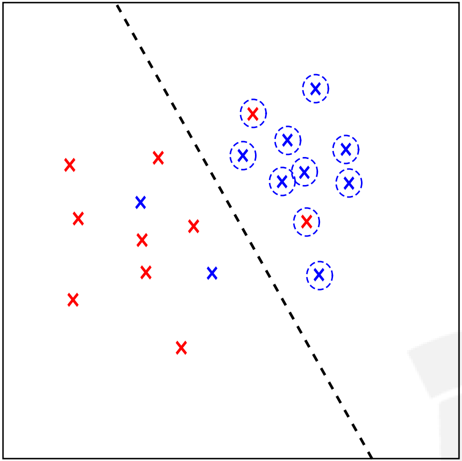
Why does CC deliver biased results?



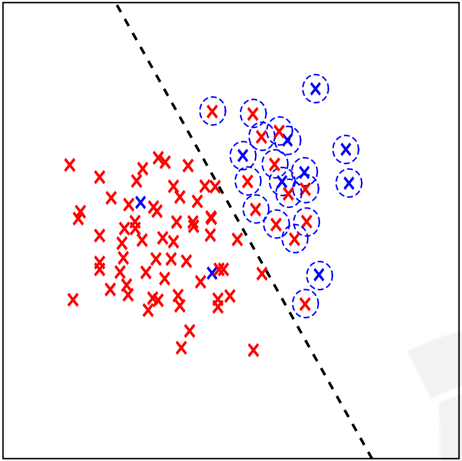
Why does CC deliver biased results?



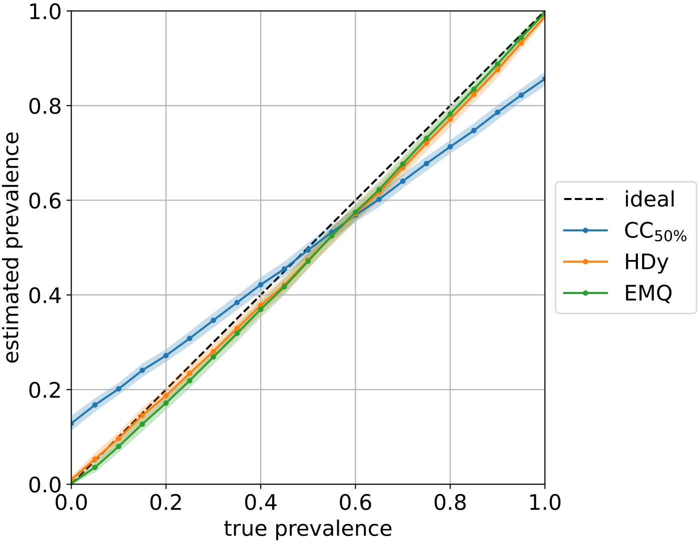
Why does CC deliver biased results?



Why does CC deliver biased results?



CC delivers biased results



Historical development

- The history of quantification research is highly non-linear (task discovered and re-discovered from within different disciplines)
- 1st stage : interest in the "estimation of class priors" in machine learning
  - Goal : making classifiers robust to the presence of prior probability shift and better attuned to the characteristics of the data to which they need to be applied
  - Earliest recorded method is (Vucetic & Obradovic, 2001), most influential one is (Saerens et al., 2002)
- 2nd stage : interest in "quantification" from data mining / text mining
  - Goal : estimating quantities and trends from unlabelled data
  - Earliest recorded work is (Forman, 2005), where the term "quantification" was coined
  - It is the applications from these fields that have provided the impetus behind the most recent wave of research in quantification

---

G. Forman. Counting positives accurately despite inaccurate classification. ECML 2005.

Slobodan Vucetic, Zoran Obradovic: Classification on Data with Biased Class Distribution. ECML 2001.

Marco Saerens, Patrice Latinne, Christine Decaestecker: Adjusting the Outputs of a Classifier to New a Priori Probabilities: A Simple Procedure. Neural Computation, 2002

Related but different: Screening tests in epidemiology

- Quantification is reminiscent of "prevalence estimation from screening tests" in epidemiology
- Screening test : a test that a patient undergoes in order to check if she has a given pathology; can be used for epidemiological purposes when administered to a certain population
- Screening tests are often imperfect, i.e., they may generate
  - false positives (patient incorrectly diagnosed with the pathology)
  - false negatives (patient incorrectly diagnosed to be free from the pathology)
- Testing a patient is thus akin to classifying an item
- Main differences:
  - no supervised learning is involved
  - a screening test typically has known and fairly constant "sensitivity" (recall) and "specificity" ($1-$fallout), while the same usually does not hold for a classifier
- Some currently used quantification methods indeed derive from methods used for prevalence estimation from screening tests

Related but different: Density estimation

- Quantification is similar to density estimation (e.g., estimating the prevalence of yellow balls in a large urn containing coloured balls).
- However, in traditional density estimation
  1. We can deterministically assess whether each item belongs to the class (variable y_j can be observed); in quantification this does not hold
  2. It is impossible / economically not viable to assess class membership for each single item (e.g., we do not want to inspect every single ball in the urn); in quantification this does not hold

Structure of this tutorial

1. Introduction
2. Applications of quantification in ML, DM, NLP
3. Evaluation measures and evaluation protocols for quantification
4. Supervised learning methods for quantification
5. QuaPy: an Open-source library for quantification
6. Advanced topics
7. Conclusions

1. Characterizing the haystack

- Many fields of human activity are not concerned with individual data but with aggregate data only, often broken down according to variables of interest (e.g., age group, gender, religion, job type, geographical region). Examples are
  - Social sciences and political sciences
  - Epidemiology
  - Market research
  - Ecological modelling
  - ...
- In these fields, whenever the variable of study (Y) is not explicit, quantification (instead of classification) is what is needed

1. Characterizing the haystack: The social sciences

- Computational social science: the big new paradigm spurred by the availability of big data from social networks
- Within the social sciences, the individuals on which we perform quantification are persons
- Example quantification endeavours are
    - Quantification by topic, e.g., as in establishing the prevalence of a certain topical class within respondents of an open-ended survey
    - Sentiment quantification, e.g., the goal of most works that do "sentiment classification of Twitter data" is estimating class prevalences
    - Stance quantification, i.e., detecting the prevalence of individuals that have a certain stance towards a given issue or topic ("target")
- Political science : e.g., predicting election results / monitoring support for a political party by estimating the prevalence of blog posts / tweets that have a certain stance towards the party
- Most works in these fields still use "classify and count", mostly due to lack of awareness of the existence of alternative quantification methods

1. Characterizing the haystack: The social sciences

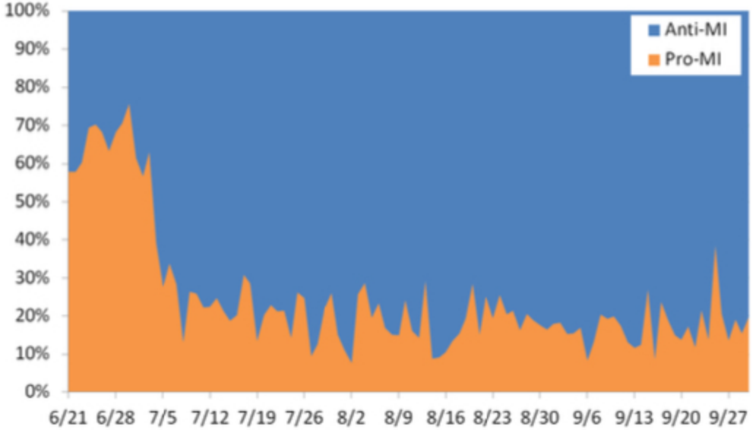


Figure: Temporal trend in the proportions of tweets supporting or opposing military intervention in Egypt during the "Arab spring" in summer 2013.

Borge-Holthoefer, J., Magdy, W., Darwish, K., and Weber, I. (2015). Content and network dynamics behind Egyptian political polarization on Twitter. In Proceedings of CSCW 2015.

1. Characterizing the haystack: Ecological modelling

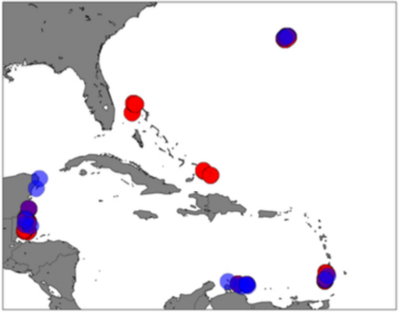


Figure: Using quantification for estimating the prevalence of different species of living beings on the seabed; red circles indicate the locations where the training data were collected while blue circles indicate the locations where the unlabelled data to which the trained model was applied were collected.

Beijbom, O., Hoffman, J., Yao, E., Darrell, T., Rodriguez-Ramirez, A., Gonzalez-Rivero, M., and Hoegh-Guldberg, O. (2015). Quantification in-the-wild: Datasets and baselines. NIPS 2015 Workshop on Transfer and Multi-Task Learning.

1. Characterizing the haystack: Ecological modelling

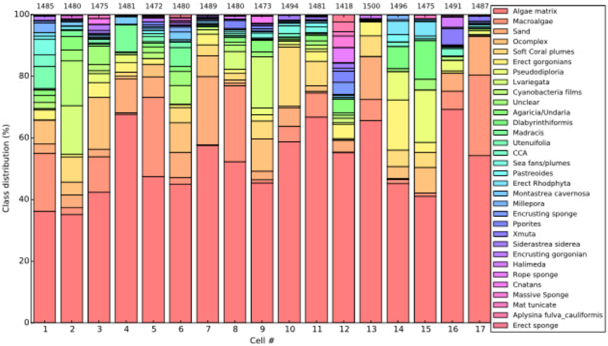


Figure: Class prevalence of each of 32 living species in seabed cover as estimated via quantification technology; the different columns represent different samples on which quantification has been performed.

Beijbom, O., Hoffman, J., Yao, E., Darrell, T., Rodriguez-Ramirez, A., Gonzalez-Rivero, M., and Hoegh-Guldberg, O. (2015). Quantification in-the-wild: Datasets and baselines. NIPS 2015 Workshop on Transfer and Multi-Task Learning.

1. Characterizing the haystack

- Market Research : estimating the distribution of consumers' attitudes towards products, product features, or marketing strategies; e.g.,
  - quantifying customers' attitudes from verbal responses to open-ended questions (Esuli and Sebastiani, 2010)
- Epidemiology : tracking the incidence and the spread of diseases; e.g.,
  - estimate pathology prevalence from clinical reports where pathologies are diagnosed
  - estimate the prevalence of different causes of death from "verbal autopsies", i.e., from verbal accounts of symptoms
- Other :
  - estimating the proportions of different types of cells in blood samples
  - estimating the proportion of no-shows within a set of bookings

A. Esuli and F. Sebastiani. Machines that learn how to code open-ended survey data. *International Journal of Market Research* 52(6):775–800, 2010.

2. Applications to downstream tasks

- Improving classification accuracy : improving the performance of classifiers when deployed on data characterized by prior probability shift
- Estimating the accuracy of a classifier on out-of-distribution data
- Improving word sense disambiguation accuracy : e.g., tuning a word sense disambiguator to a domain characterized by sense priors different from those of the training set
  - e.g., sense of "bank" in financial documents vs. hydraulic engineering papers
- Estimating the fairness under unawareness wrt a sensitive attribute
  - classifiers (automatic decision making, e.g., loan approved/denied)
  - rankers (e.g., job candidates)

Marco Saerens, Patrice Latinne, Christine Decaestecker: Adjusting the Outputs of a Classifier to New a Priori Probabilities: A Simple Procedure. Neural Computation, 2002

YS Chan and HT Ng. Estimating class priors in domain adaptation for word sense disambiguation. *Proceedings of ACL 2006*.

A Fabris, A Esuli, A Moreo, and F Sebastiani. Measuring Fairness under Unawareness of Sensitive Attributes: A Quantification-Based Approach. *Journal of Artificial Intelligence Research* 2023.

Structure of this tutorial

1. Introduction
2. Applications of quantification in ML, DM, NLP
3. Evaluation measures and evaluation protocols for quantification
4. Supervised learning methods for quantification
5. QuaPy: an Open-source library for quantification
6. Advanced topics
7. Conclusions

Notation and terminology

- Domain $\mathcal{X}$ of items (documents), set $\mathcal{Y}$ of classes
- Different brands of classification :
  - Binary classification: each item has exactly one of $\mathcal{Y} = \{y_1, y_2\}$ (which we often write $\mathcal{Y} = \{\oplus, \ominus\}$)
  - Single-label multi-class classification (SLMC): each item has exactly one of $\mathcal{Y} = \{y_1, ..., y_n\}$, with $n > 2$
  - Multi-label multi-class classification (MLMC): each item may have zero, one, or several among $\mathcal{Y} = \{y_1, ..., y_n\}$, with $n > 1$
    - MLMC is often reduced to binary by solving n independent binary classification problems
  - Ordinal classification (aka "ordinal regression"): each item has exactly one of $\mathcal{Y} = (y_1 \preceq ... \preceq y_n)$, where $\preceq$ is a total order and $n > 2$
  - Metric regression: each item has a real-valued score from the range $[\alpha, \beta]$
- For each such brand of classification we will be interested in its "quantification equivalent"
- Most of our discussion will be framed in terms of SLMC quantification

How do we evaluate quantification methods?

- Evaluating quantification means measuring how well a predicted probabilistic distribution $\hat{p}(y)$ fits a true distribution $p(y)$
- The goodness of fit between two categorical distributions can be computed via divergence functions $D(p, \hat{p})$ which enjoy
  1. $D(p, \hat{p}) = 0$ only if $p = \hat{p}$ (identity of indiscernibles)
  2. $D(p, \hat{p}) \geq 0$ (non-negativity)
- Divergences are less restrictive than distances, which must additionally enjoy
  - $D(p, q)$ implies $D(q, p)$ (symmetry)
  - $D(p, q) + D(q, r) \geq D(p, r)$ (triangle inequality)

  in quantification we use divergences (note distances are also divergences)

F. Sebastiani. Evaluation measures for quantification: An axiomatic approach, *Information Retrieval Journal*, 2019.

How do we evaluate quantification methods?

- Binary case: Let us define $p = (p_1, p_2)$ and $p' = (p'_1, p'_2)$, two binary distributions such that $p_1 < p'_1 \leq p'_2 < p_2$, and p_t the distribution obtained from p such that $p_t = (p_1 + t, p_2 - t)$.

  3. impartiality: $D(p, p_{-t}) = D(p, p_{+\alpha})$
     - Enforces the notion that underestimation and overestimation are equally serious
     - E.g., if D enjoys impartiality, it considers estimating $p(y) = .20$ as $\hat{p}(y) = .10$ or as $\hat{p}(y) = .30$ equally serious mistakes

  4. relativity: $D(p, p_{+t}) > D(p', p'_{+t})$
     - Enforces the notion that estimation errors of the same absolute magnitude are more serious for rare classes;
     - E.g., if D enjoys relativity, it considers predicting $\hat{p}(y) = 0.01$ when $p(y) = 0.02$ more serious than predicting $\hat{p}(y) = 0.49$ when $p(y) = 0.50$

How do we evaluate quantification methods?

- Q: Which evaluation function is more desirable?

How do we evaluate quantification methods?

- Q: Which evaluation function is more desirable?
- A: It depends on the application; arguably, for some applications relativity is desired, while for some others it is not; e.g.
- Application 1: estimating the prevalence of illnesses in a given region / age group. Here, relativity is desired (since, e.g., a .01 estimation error may be tolerable if $p(y) = .40$ but not if $p(y) = .0001$).

How do we evaluate quantification methods?

- Q: Which evaluation function is more desirable?
- A: It depends on the application; arguably, for some applications relativity is desired, while for some others it is not; e.g.
- Application 1: estimating the prevalence of illnesses in a given region / age group. Here, relativity is desired (since, e.g., a .01 estimation error may be tolerable if $p(y) = .40$ but not if $p(y) = .0001$).
- Application 2: predicting the prevalence of no-shows on a flight-by-flight basis. Here, relativity is undesired (since, e.g., a .02 estimation error has the same impact if $p(y) = .05$ and if $p(y) = .20$).
- Identity of indiscernibles and non-negativity are arguably always desirable

How do we evaluate quantification methods?

- Divergences frequently used for evaluating (binary, SLMC, and MLMC) quantification are
  - $\mathrm{AE}(p,\hat{p}) = \frac{1}{|\mathcal{Y}|}\sum_{y\in\mathcal{Y}} |\hat{p}(y) - p(y)|$ (Absolute Error)
  - $\mathrm{RAE}(p,\hat{p}) = \frac{1}{|\mathcal{Y}|}\sum_{y\in\mathcal{Y}} \frac{|\hat{p}(y) - p(y)|}{p(y)}$ (Relative Absolute Error)
  - $\mathrm{KLD}(p,\hat{p}) = \sum_{y\in\mathcal{Y}} p(y) \log \frac{p(y)}{\hat{p}(y)}$ (Kullback-Leibler Divergence)

	Impartiality	Relativity
Absolute Error	Yes	No
Relative Absolute Error	Yes	Yes
Kullback-Leibler Divergence	No	Yes

- AE and RAE are indeed the most satisfactory measures of quantification error
- For MLMC quantification, "macroaveraged" versions of these measures, obtained by averaging them across the classes, are used

Fabrizio Sebastiani. Evaluation Measures for Quantification: An Axiomatic Approach. Information Retrieval Journal 23(3):255-288, 2020

How do we evaluate quantification methods?

- RAE and KLD may sometimes be undefined due to the presence of zero denominators.
- To solve this we can smooth $p(y)$ and $\hat{p}(y)$ via additive smoothing and use the smoothed versions in place of the original ones; the smoothed version of $p(y)$ is

$$p_s(y) = \frac{\epsilon + p(y)}{\epsilon|\mathcal{Y}| + \sum_{y \in \mathcal{Y}} p(y)} = \frac{\epsilon + p(y)}{\epsilon|\mathcal{Y}| + 1} \tag{1}$$

- $\epsilon = \frac{1}{2|S|}$ is often used as a smoothing factor
- For example: RAE between two smoothed distributions simplifies to

$$\text{RAE}(p_s, \hat{p}_s) = \frac{1}{|\mathcal{Y}|} \sum_{y \in \mathcal{Y}} \frac{|\hat{p}(y) - p(y)|}{p(y) + \epsilon}$$

Multi-objective loss functions

- The "paradox of quantification":

h_1		actual	
		y	$\overline{y}$
pred	y	0	1000
	$\overline{y}$	1000	0

h_2		actual	
		y	$\overline{y}$
pred	y	990	0
	$\overline{y}$	10	1000

- h_1 yields better AE / RAE / KLD than h_2, but we intuitively prefer h_2 to h_1
- It is difficult to trust an aggregative quantifier if it is not based on a good enough classifier ...

Multi-objective loss functions

- The MOLF multi-objective loss function (Milli et al., 2013) strives to keep both classification and quantification error low

$$\begin{aligned}\mathrm{MOLF}(p,\hat{p}) &= \sum_{y_j\in\mathcal{Y}} |\,\mathrm{FP}_j^2 - \mathrm{FN}_j^2\,| \\ &= \sum_{y_j\in\mathcal{Y}} (\mathrm{FN}_j+\mathrm{FP}_j)\cdot|\,\mathrm{FN}_j-\mathrm{FP}_j\,|\end{aligned}$$

  since
  - $|\,\mathrm{FN}_j - \mathrm{FP}_j\,|$ is a measure of quantification error
  - $(\mathrm{FN}_j + \mathrm{FP}_j)$ is a measure of classification error
- It makes sense to use MOLF as a loss function to minimize, but not as a measure for evaluating quantification accuracy
- It applies to "aggregative" quantifiers only

L. Milli, A. Monreale, G. Rossetti, F. Giannotti, D. Pedreschi, F. Sebastiani. Quantification trees. ICDM 2013, pp. 528–536.

J. Barranquero, J. Díez, and J. del Coz. Quantification-oriented learning based on reliable classifiers. *Pattern Recognition* 48(2):591–604, 2015.

Measures for evaluating ordinal quantification

- Ordinal classification $\equiv$ SLMC classification when there is a total order on the n classes
- Important in the social sciences, where ordinal scales are often used to elicit human evaluations (e.g., product reviews)
- Important also in astrophysics, where ordinal scales are used to "bin" energy levels of astroparticles
- The most frequently used measure for ordinal quantification is the (normalized) Earth Mover's Distance (aka "Wasserstein metric")

$$\mathrm{EMD}(p, \hat{p}) = \frac{1}{|\mathcal{Y}|-1} \sum_{j=1}^{|\mathcal{Y}|-1} \left| \sum_{i=1}^{j} \hat{p}(y_i) - \sum_{i=1}^{j} p(y_i) \right| \tag{2}$$

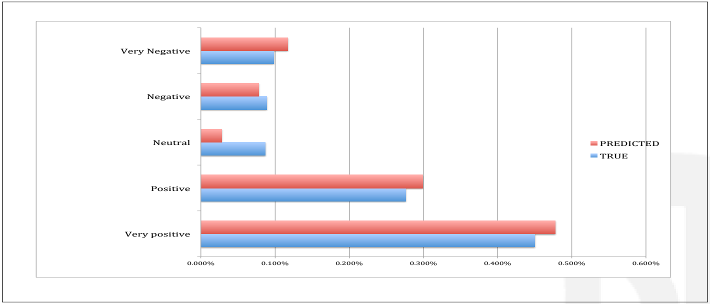
- The EMD is the "ordinal analogue" of absolute error

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

Mirko Bunse, Alejandro Moreo, Fabrizio Sebastiani, and Martin Senz. Ordinal quantification through regularization. ECML/PKDD 2022.

Measures for evaluating ordinal quantification

- The EMD may be seen as measuring the "minimum effort" to turn the predicted distribution into the true distribution, where the effort is measured by
  - the probability masses that need to be moved between one class and another;
  - the "distance" travelled by these probability masses



Experimental protocols for evaluating quantification

- Any test set used for testing the accuracy of classification can obviously be used as a sample σ also for evaluating quantification
- However, while for classification a set of k unlabelled datapoints provides k test datapoints, for quantification a set of k unlabelled datapoints provides only 1 test datapoint
- An experimental protocol for quantification is an algorithm for extracting, from a test set of labelled datapoints, a set $U = \{\sigma_1, \sigma_2, ...\}$ of samples on which quantifiers should be tested
- Different protocols must be chosen for different quantification tasks (binary, multiclass, multilabel, ordinal)

G. Forman. Counting positives accurately despite inaccurate classification. ECML 2005.

Andrea Esuli, Alejandro Moreo, and Fabrizio Sebastiani. LeQua@CLEF2022: Learning to Quantify. ECIR 2022.

Alejandro Moreo, Manuel Francisco, and Fabrizio Sebastiani. Multi-Label Quantification. arXiv:2211.08063 [cs.LG].

Mirko Bunse, Alejandro Moreo, Fabrizio Sebastiani, and Martin Senz. Ordinal quantification through regularization. ECML/PKDD 2022.

Experimental protocols for evaluating quantification

- Two main protocols are used in the literature:
  - The artificial-prevalence protocol (APP): take a standard dataset split into L and U, and extract a set of samples that exhibits the highest possible diversity in terms of class distribution
    - **Pros**: challenging, since some samples exhibit high amounts of PPS
    - **Cons**: samples with unrealistically high amounts of PPS may influence the results too much + only deals with PPS
  - The natural-prevalence protocol (NPP): pick one or more standard datasets that represent a wide array of class prevalence values
    - **Pros**: experimental setting is realistic
    - **Cons**: class prevalence values and shift values may not be varied at will
- The NPP has almost been abandoned now, due to the difficulty of finding datasets that are challenging enough, i.e., displaying substantial amounts of PPS, and sizeable enough
- Research is ongoing for defining protocols that simulate types of dataset shift other than PPS

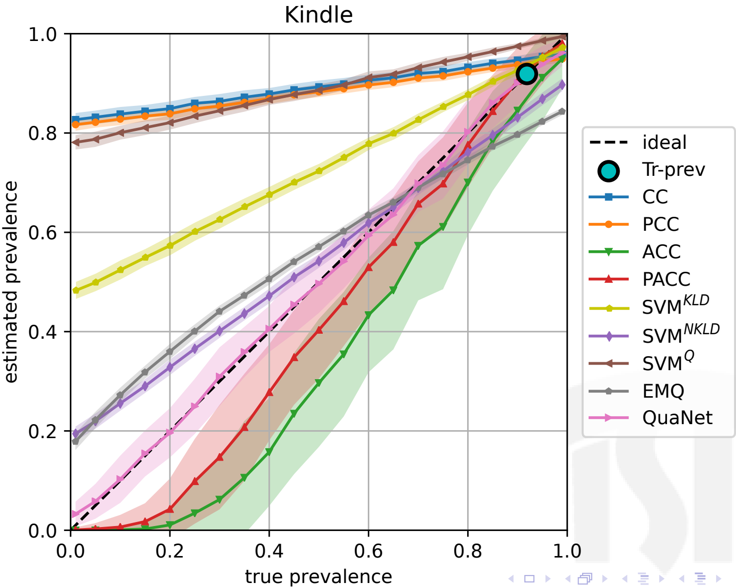
G. Forman. Counting positives accurately despite inaccurate classification. ECML 2005.

A. Esuli and F. Sebastiani. Optimizing text quantifiers for multivariate loss functions. *ACM Transactions on Knowledge Discovery and Data*, 9(4):Article 27, 2015.

The artificial-prevalence protocol in the multiclass case

- The APP in the binary case consists of
  1. establishing a grid of values in the [0,1] interval, e.g., $G = \{0.00, 0.05, ..., 0.95, 1.00\}$
  2. for each value $\alpha \in G$ extract, by random sampling with replacement, m samples of k datapoints each such that the prevalence $p_\sigma(y_1)$ of the positive class in the sample is α;
  3. use the set of $|G| \times m$ extracted samples as the test set for evaluating quantifiers.

The APP for the binary case: An example



The artificial-prevalence protocol in the multiclass case

- Using the APP in the multiclass case can be problematic since, given a grid G, the number of samples that can be extracted via the above method is $O(g^n)$
- We can then resort to extracting samples whose distribution is extracted uniformly at random from the unit $(n-1)$-simplex

$$\Delta^{n-1} = \{p_1, \ldots, p_n : p_i \geq 0, \sum_{i=1}^{n} p_i = 1\}$$

- However, in order to guarantee randomness we need to avoid naive extraction algorithms ...

Sampling uniformly at random from the unit simplex

- The naive algorithm (NA):
  1. Given a set of classes $\mathcal{Y}$, generate a vector $A = \langle a_1, \ldots, a_{(|\mathcal{Y}|-1)} \rangle$ of datapoints sampled uniformly at random from [0,1]
  2. Obtain a vector $P = \langle p_1, \ldots, p_{|\mathcal{Y}|} \rangle$ by defining

$$p_i = \begin{cases} a_i \prod_{j=1}^{i-1}(1-a_j) & \text{if } i < |\mathcal{Y}| \\ (1 - \sum_{j=1}^{|\mathcal{Y}|-1} p_j) & \text{if } i = |\mathcal{Y}| \end{cases}$$

  3. Use P as the distribution of class prevalence values for generating sample σ

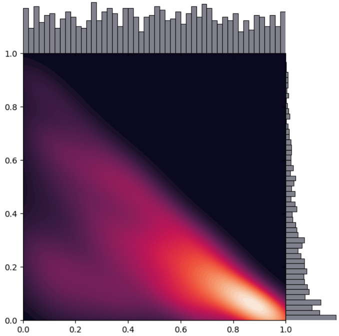


Figure: Distribution of datapoints $\langle p_1, p_2, p_3 \rangle$ sampled via the NA on the unit 2-simplex.

Sampling uniformly at random from the unit simplex

- The IID algorithm (IIDA):
  1. Given a set of classes $\mathcal{Y}$, generate a vector $A = \langle a_1, ..., a_{|\mathcal{Y}|} \rangle$ of datapoints sampled uniformly at random from [0,1]
  2. Obtain a vector $P = \langle p_1, ..., p_{|\mathcal{Y}|} \rangle$ by normalizing A to unit length
  3. Use P as the distribution of class prevalence values for generating sample σ

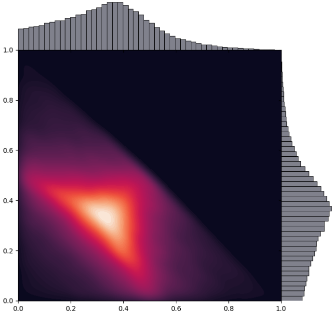


Figure: Distribution of datapoints $\langle p_1, p_2, p_3 \rangle$ sampled via the IIDA on the unit 2-simplex.

Sampling uniformly at random from the unit simplex

- The Kraemer algorithm (KA):
  1. Given a set of classes $\mathcal{Y}$, generate a vector $A = \langle a_1, ..., a_{(|\mathcal{Y}|-1)} \rangle$ of datapoints sampled uniformly at random from [0,1]
  2. Sort the a_i's to obtain $B = \langle b_1 \leq ... \leq b_{(|\mathcal{Y}|-1)} \rangle$, and define $b_0 = 0$ and $b_{|\mathcal{Y}|} = 1$
  3. Obtain a vector $P = \langle p_1, ..., p_{|\mathcal{Y}|} \rangle$ by defining $p_i = b_i - b_{(i-1)}$ for all $i \in \{1, ..., |\mathcal{Y}|\}$
  4. Use P as the distribution of class prevalence values for generating sample σ

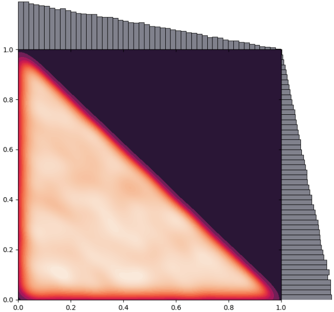


Figure: Distribution of datapoints $\langle p_1, p_2, p_3 \rangle$ sampled via the KA on the unit 2-simplex.

Smith, Noah A. and Tromble, Roy W., Sampling uniformly from the unit simplex, Technical report, Johns Hopkins University, 2004. https://www.cs.cmu.edu/~nasmith/papers/smith+tromble.tr04.pdf

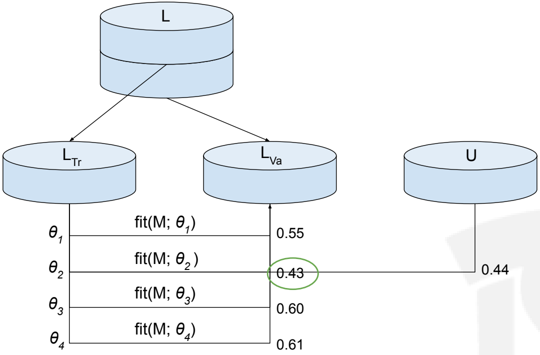
Model Selection in Quantification

- The performance of machine learning algorithms typically depends on how their hyperparameters are set.
- The process of hyperparameter optimisation is known as model selection, and consists of testing how well the model fares with different combinations of hyperparameters on held-out validation data.
- Model selection is inherently related to evaluation.
- Since quantification has specific evaluation measures and specific evaluation protocols, model selection should be in agreement with these.

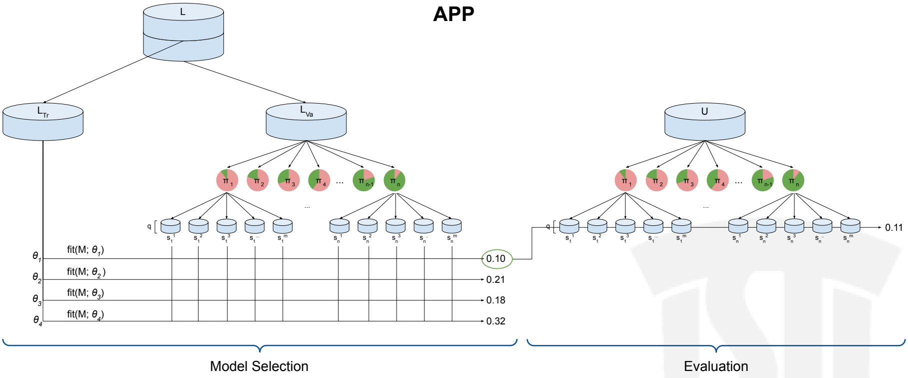
Moreo & Sebastiani. Re-assessing the "classify and count" quantification method. ECIR 2021.

Model Selection in Quantification

- Many papers have instead carried out model selection mimicking the classification approach, i.e.:



- This is theoretically flawed: model selection has to be carried out following a quantification-oriented evaluation protocol:



Structure of this tutorial

1. Introduction
2. Applications of quantification in ML, DM, NLP
3. Evaluation measures and evaluation protocols for quantification
4. Supervised learning methods for quantification
5. QuaPy: an Open-source library for quantification
6. Advanced topics
7. Conclusions

Advanced topics (hints)

- Cross-lingual quantification
- Multi-label quantification
- Ordinal quantification
- Regression quantification
- Quantification for networked data
- Quantification for DS types other than PPS
- ...

Advanced topics (hints)

- Cross-lingual quantification
- Multi-label quantification
- Ordinal quantification
- Regression quantification
- Quantification for networked data
- Quantification for DS types other than PPS
- ...

Cross-lingual quantification

- An instance of transfer learning in which the supervised information available is in one language, but we want to deploy a model in another language.
- Example:
  - training documents are book reviews in English
  - test documents are book reviews in Spanish
- Problem: predict the percentage of positive opinions in books reviews written in Spanish
- Solution:
  - Adopt any aggregative quantifier (e.g., SLD)
  - Use a cross-lingual classifier as the underlying classifier (e.g., DCI)
- Less than 3% of error!

Esuli, A., Moreo, A., & Sebastiani, F. (2020). Cross-lingual sentiment quantification. IEEE Intelligent Systems, 35(3), 106-114.

Multi-label quantification

- Multi-label multi-class (MLMC) quantification: each item may have zero, one, or several among $\mathcal{Y} = \{y_1, ..., y_n\}$, with $n > 1$
- MLMC quantification is often reduced to binary quantification by solving n independent binary quantification problems; this is the baseline that all "truly" MLMC quantification methods are supposed to beat
- This "reduction to binary" does not allow leveraging possible stochastic correlations between classes; e.g., we may notice from training data that many datapoints in class "Technology" are also in class "Startups"
- Work in MLMC classification has shown that leveraging these correlations brings about higher accuracy

Multi-label quantification

- For simplicity, we will deal with aggregative quantification methods only
- The most trivial class of solutions to MLMC quantification is BC+BA, which consists of using n binary classifiers and, on top of them, n instances of a binary aggregative quantification method
- A slightly less trivial class of solutions is MLC+BA, which consists of using a truly multi-label classifier and, on top of it, n instances of a binary aggregative quantification method
- Another less trivial class of solutions is BC+MLA, which consists of using n independent binary classifiers and, on top of them, a truly multi-label aggregative quantification method
- The most interesting class of solutions is MLC+MLA, which consists of using a truly multi-label classifier and, on top of it, a truly multi-label aggregative quantification method

A Moreo, M Francisco, F Sebastiani. ACM Transactions on Knowledge Dicovery and Data, 2023.

Multi-label quantification

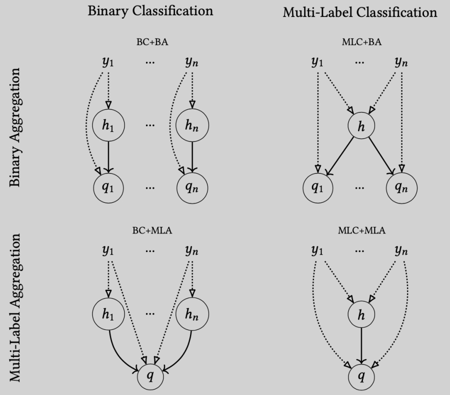


Fig. 2. The four groups of multi-label quantification methods. Dotted lines connecting class labels with a model (classifier or quantifier) indicate that the model learns from (or has access to) the class labels of the training datapoints. Solid lines connecting classifiers with quantifiers indicate a transfer of outputs from the classifier to the quantifier. With a slight deviation from our notation, here h denotes any classifier, hard or soft.

Alejandro Moreo, Manuel Francisco, Fabrizio Sebastiani. Multi-label quantification. arXiv:2211.08063 [cs.LG], 2022.

Multi-label quantification

- Best-performing system so far: the regression-based MLC+MLA quantification method (Moreo et al. 2023):
  1. Take a multi-label quantifier q trained via a MLC+BA quantification method
  2. Put a regressor $r : \mathbb{R}^n \rightarrow \mathbb{R}^n$ on top of it that takes as input a vector of n "uncorrected" prevalence values and returns a vector of n "corrected" prevalence values
  3. Train the regressor with a set of pairs $(\hat{\mathbf{p}}^q_{\sigma_i}, \mathbf{p}_{\sigma_i})$, where
     - $\hat{\mathbf{p}}^q_{\sigma_i}$ is the vector of the n prevalence values estimated by q
     - $\mathbf{p}_{\sigma_i}$ is the vector of the n true prevalence values
- The regressor is thus trained to leverage the stochastic dependencies among the classes
- This method can be used also if the MLC+BA underlying method is non-aggregative
- (Moreo et al. 2023) provide an experimental protocol specific to multi-label quantification, that can be used for evaluation and also for generating the σ_i's to be used in Step 3 above

A Moreo, M Francisco, F Sebastiani. ACM Transactions on Knowledge Dicovery and Data, 2023.

Ordinal quantification

- Ordinal quantification is SLMC quantification when there is a total order $\mathcal{Y} = (y_1 \preceq ... \preceq y_n)$ on the classes
- Mis-assigning probability mass to a neighbouring class is less serious than mis-assigning it to a faraway class; EMD is thus a good evaluation measure;
- Few research works conducted on this task
  - Early OQ algorithms are (da San Martino et al., 2016) and (Esuli, 2016)
  - "Unfolding" algorithms in the astrophysics literature (Bunse, 2018)
  - More recent algorithms are (Bunse et al., 2022) and (Castaño et al, 2022)

Ordinal quantification

- Class of OQ methods based on regularization (Bunse et al., 2022, 2024)
- Basic idea: take an algorithm for SLMC quantification, and introduce a "regularization" that penalizes "unlikely" assignments of probability mass
- "Likely $\approx$ Smooth", i.e., sharp differences between $p_\sigma(y_i)$ and $p_\sigma(y_{i+1})$ are considered unlikely
- Several algorithms proposed along these lines, including o-ACC, o-PACC, o-SLD

Ordinal quantification

- E.g., o-ACC, an ordinal version of ACC:
- ACC amounts to solving for $\mathbf{p}$ the system of linear equations $\mathbf{q} = \mathbf{M}\mathbf{p}$, where $\mathbf{q} \in \mathbb{R}^n$ are the prevalence estimates obtained via CC and $\mathbf{M}$ is the misclassification matrix.
- Least-squares solutions to this system are found by computing

$$\mathrm{argmin}_{\mathbf{p}} \|\mathbf{q} - \mathbf{M}\mathbf{p}\|_2^2$$

- A regularization term is introduced that penalizes non-smooth solutions

$$\mathrm{argmin}_{\mathbf{p}} \|\mathbf{q} - \mathbf{M}\mathbf{p}\|_2^2 + \frac{\tau}{2}\left(\mathbf{C}\mathbf{p}\right)^2 \quad (3)$$

where the Tikhonov matrix $\mathbf{C}$ is such that

$$\frac{1}{2}\left(\mathbf{C}\mathbf{p}\right)^2 = \frac{1}{2}\sum_{i=2}^{n-1}\left(-[\mathbf{p}]_{i-1} + 2[\mathbf{p}]_i - [\mathbf{p}]_{i+1}\right)^2 \quad (4)$$

Mirko Bunse, Alejandro Moreo, Fabrizio Sebastiani, Martin Senz: Ordinal Quantification Through Regularization. ECML/PKDD (5) 2022: 36-52

Open challenges for quantification

- Quantification has not received the same attention as classification; therefore, many open problems still remain; there is a need, e.g., to

  1. Investigate non-aggregative quantification methods more extensively, since they are the true realization of "Vapnik's principle"
  2. Investigate transductive quantification methods, to take advantage of the fact that transductive contexts are "easier"
  3. Devise methods for exploiting the full potential of deep learning for quantification
  4. Better investigate the relationships between quantification and types of dataset shift other than PPS

Structure of this tutorial

1. Introduction
2. Applications of quantification in ML, DM, NLP
3. Evaluation measures and evaluation protocols for quantification
4. Supervised learning methods for quantification
5. QuaPy: an Open-source library for quantification
6. Advanced topics
7. Conclusions

Conclusion

- Growing awareness that quantification is going to be more and more important; given the advent of big data, application contexts will spring up in which we will simply be happy with analysing data at the aggregate (rather than at the individual) level
- Takeaway message to users of supervised learning: when
  - You are using classification
  - Your only goal is to obtain aggregate results, i.e., class prevalence estimates

  your work would probably benefit from using quantification technology instead of classification technology

Thank you!

Questions?