Proceedings of the 2nd International Workshop on Learning to Quantify (LQ 2022)

Juan José del Coz, Pablo González,
Alejandro Moreo, and Fabrizio Sebastiani (eds.)

Preface

The 2nd International Workshop on Learning to Quantify (LQ 2022 – `https://lq-2022.github.io/`) was held in Grenoble, FR, on September 23, 2022, as a satellite workshop of the European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML/PKDD 2022). While the 1st edition of the workshop (LQ 2021 – `https://cikmlq2021.github.io/`, which was instead co-located with the 30th ACM International Conference on Information and Knowledge Management (CIKM 2021)) had to be an entirely online event, LQ 2022 was a hybrid event, with presentations given in-presence and both in-presence attendees and remote attendees.

The workshop was a half-day event, and consisted of a keynote talk by Marco Saerens (Université Catholique de Louvain), presentations of four contributed papers, and a final collective discussion on the open problems of learning to quantify and on future initiatives.

The present volume contains the four contributed papers that were accepted for presentation at the workshop. Each of these papers was submitted as a response to the call for papers, was reviewed by at least three members of the international program committee, and was revised by the authors so as to take into account the feedback provided by the reviewers. We hope that the availability of the present volume will increase the interest in the subject of quantification on the part of researchers and practitioners alike, and will contribute to making quantification better known to potential users of this technology and to researchers interested in advancing the field.

Juan José del Coz
Pablo González
Alejandro Moreo
Fabrizio Sebastiani

Table Of Contents

LQ 2022 Program Committee

Juan José del Coz, University of Oviedo, ES (co-Chair)
Pablo González, University of Oviedo, ES (co-Chair)
Alejandro Moreo, Consiglio Nazionale delle Ricerche, IT (co-Chair)
Fabrizio Sebastiani, Consiglio Nazionale delle Ricerche, IT (co-Chair)

Rocío Alaíz-Rodríguez, University of León, ES
Gustavo Batista, University of New South Wales, AU
Mirko Bunse, University of Dortmund, DE
Andrea Esuli, Consiglio Nazionale delle Ricerche, IT
Alessandro Fabris, Università di Padova, IT
Cèsar Ferri, Universitat Politècnica de València, ES
George Forman, Amazon Research, US
Wei Gao, Singapore Management University, SG
Eric Gaussier, University of Grenoble, FR
Rafael Izbicki, Federal University of São Carlos, BR
André G. Maletzke, Universidade Estadual do Oeste do Paraná, BR
Marco Saerens, Catholic University of Louvain, BE
Dirk Tasche, Swiss Financial Market Supervisory Authority, CH

Unification of Algorithms for Quantification and Unfolding*

Mirko Bunse[0000−0002−5515−6278] (✉) and Katharina Morik[0000−0003−1153−5986]

Artificial Intelligence Unit, TU Dortmund University, 44227 Dortmund, Germany
{firstname.lastname}@cs.tu-dortmund.de

Abstract. Quantification is the supervised learning task of predicting the prevalence values of classes in a data sample. Physics literature knows the same task under a different name: unfolding. However, the literature on quantification and the literature on unfolding are largely disconnected from each other, likely due to an interdisciplinary gap. We bridge this gap by proposing a common framework that integrates algorithms from both fields in a unified form. Instantiations of our framework differ from each other in terms of the loss functions, the regularizers, and the feature transformations they employ.

Keywords: Quantification · Unfolding · Classification · Experimental physics · Machine learning.

1 Introduction

Many applications of supervised learning require a prediction of the *distribution* of the target quantity, as exhibited by some data sample. In these applications, predictions for individual data instances are only secondary; they are issued as a means from which the distribution can be reconstructed. Examples of such applications are text sentiment analyses [11], technical support log analyses [10], social sciences [15], the reconstruction of energy spectra in astroparticle physics [6], and several other areas.

Supervised learning for the prediction of target distributions is known as *quantification learning* [10,12]. Within experimental physics, however, the same problem is called *unfolding* [2,14,7] or *deconvolution* [6]. As of today, the literature from quantification research and the literature from unfolding research are largely disconnected from each other, despite their substantial similarities in terms of their problem statements and their solutions.

Contributions We propose a common framework for algorithms that stem from quantification literature and from unfolding literature. This framework reveals several similarities between existing methods from the two research fields. Moreover, it paves the way for strengthening interdisciplinary efforts on the subject. Our presentation completes a similar unification attempt by Firat [9] in terms

* This paper is a slightly modified resubmission of a recent publication by us [5]

of i) taking unfolding algorithms into consideration and ii) giving formal proofs about the correctness of our framework. Our reusable implementation of all methods is available online.[1]

Sec. 2 details unfolding algorithms within our unifying framework. In Sec. 3, we integrate algorithms from quantification literature. We summarize our findings in Tab. 1 before Sec. 4 concludes.

2 Unfolding

A frequent objective in experimental physics is to estimate the spectrum of a physical quantity that cannot be measured directly. In this case, the spectrum needs to be reconstructed from correlated quantities which are measured instead.

To this end, assume that we can measure the distribution $q(\vec{x}) = \mathbb{P}(X = \vec{x})$ of some quantity X within a sample. Moreover, let the measurement process be characterized through the conditional probabilities $M(\vec{x} \mid y_c) = \mathbb{P}(X = \vec{x} \mid Y_c = y)$ of measuring some $\vec{x} \in \mathcal{X}$ when the relevant quantity has the (possibly continuous) value $y \in \mathcal{Y}_c$. The objective of any unfolding algorithm is then to reconstruct the relevant distribution $p(y) = \mathbb{P}(Y_c = y)$ from the distributions q and M, according to the integral

$$q(\vec{x}) \;=\; \int_{\mathcal{Y}_c} M(\vec{x} \mid y) \cdot p(y) \; \mathrm{d}y. \tag{1}$$

The estimation of $p(y)$ from data is enabled through the discretization of Eq. 1. In case of a continuous target interval $\mathcal{Y}_c = [a, b)$, we first need to map each continuous label to a discrete class index $\mathcal{Y} = \{1, \ldots, C\}$. For instance, the estimation of an energy spectrum requires a binning of the interval $\mathcal{Y}_c$ into C bins [3,7]. We proceed similarly with the feature space $\mathcal{X} \subseteq \mathbb{R}^d$, in mapping it to a discrete feature representation $f(\vec{x}) \in \{1, \ldots, F\}$, which is still to be defined for each unfolding algorithm in particular.

The discretization of y and $\vec{x}$ gives rise to a straightforward representation of distributions in terms of histograms. Consider a data sample $D = \{(\vec{x}_i, y_i) \in \mathcal{X} \times \mathcal{Y} : 1 \leq i \leq N\}$ in which the classes y_i are not observed. Estimating the quantities from Eq. 1 in terms of histograms

$$\vec{p} \;=\; \frac{1}{N} \sum_{i=1}^{N} \delta_{y_i}, \qquad \vec{q} \;=\; \frac{1}{N} \sum_{i=1}^{N} \delta_{f(\vec{x}_i)}, \qquad [\delta_j]_k \;=\; \begin{cases} 1 & \text{if } j = k, \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

leads to the system of linear equations

$$\vec{q} \;=\; M \cdot \vec{p}, \tag{3}$$

where the transfer matrix $M \in \mathbb{R}^{C \times F}$ is estimated by counting and normalizing the co-occurrences of labels y and transformed features $f(\vec{x})$ in a training set. Advanced algorithms are required to estimate $\vec{p}$ because a direct solution $M^{-1}\vec{q}$ is not guaranteed to exist.

[1] https://github.com/mirkobunse/QUnfold.jl

A Common Framework for Unfolding and Quantification Unfolding algorithms solve Eq. 3 for $\vec{p}$, a histogram estimate of the (continuous) distribution $p(y)$ from Eq. 1. However, M is not invertible in general. A general regularized solution for the unfolding / quantification problem, with a regularization strength $\tau \geq 0$, is

$$\vec{p}^{\,*} \;=\; \underset{\vec{p}\geq 0\,\text{s.t.}\,\mathbb{1}^\top\vec{p}=1}{\arg\min}\; \mathcal{L}(\vec{p};\,\vec{q},M) + \tau\cdot r(\vec{p}), \tag{4}$$

where the loss function $\mathcal{L}:\mathbb{R}^C\to\mathbb{R}$ and the regularization function $r:\mathbb{R}^C\to\mathbb{R}$ are still to be defined for each particular unfolding / quantification method. The constraints in Eq. 4 ensure that $\vec{p}^{\,*}$ represents a valid probability density. Our framework extends the one by Firat [9] with regularization functions $r(\vec{p})$.

Adhering to this framework are the most important unfolding algorithms, namely

Regularized Unfolding (RUN) [3,2] RUN models the likelihood of solutions in terms of Poisson-distributed counts. Namely, we observe a histogram of counts $\bar{q} = N\cdot\vec{q}\in\mathbb{N}^F$, each element of which is modelled as being Poisson-distributed with the rate $\lambda_i = [M\vec{p}]_i$. This modelling gives rise to the negative log-likelihood function

$$\mathcal{L}^{\mathrm{RUN}}(\vec{p};\,\vec{q},M) \;=\; \sum_{i=1}^{F}[M\vec{p}]_i - \bar{q}_i\ln[M\vec{p}]_i, \tag{5}$$

which RUN minimizes.

To ensure smooth solutions, RUN employs Tikhonov regularization. The Tikhonov matrix $T\in\mathbb{R}^{C\times C}$ is defined such that

$$r^{\mathrm{RUN}}(\vec{p}) \;=\; \frac{1}{2}\,(T\vec{p})^2 \;=\; \frac{1}{2}\sum_{i=2}^{C-1}\left([\vec{p}]_{i-1} - 2[\vec{p}]_i + [\vec{p}]_{i+1}\right)^2. \tag{6}$$

Unfolding via Singular Value Decomposition (SVD) [14] This method employs the regularizer from Eq. 6 with a least squares loss

$$\mathcal{L}^{\mathrm{SVD}}(\vec{p};\,\vec{q},M) \;=\; \left\|\frac{\vec{q}-M\vec{p}}{\vec{w}}\right\|_2^2, \tag{7}$$

which is weighted by a vector $\vec{w}\in\mathbb{R}^F$. For instance, a Poisson model can be realized through Poisson variances $\vec{w}=\sqrt{\bar{q}}$.

Iterative Bayesian Unfolding (IBU) [8,7] IBU revolves around an expectation maximization approach. Starting from a prior $\vec{p}^{\,(0)}$, it repeatedly updates the estimate $\vec{p}^{\,(k)}$ according to Bayes' theorem

$$[\vec{p}^{\,(k)}]_i \;=\; \sum_{j=1}^{F}\frac{[M]_{ij}[\vec{p}^{\,(k-1)}]_i}{\sum_{i'=1}^{C}[M]_{i'j}[\vec{p}^{\,(k-1)}]_{i'}}[\vec{q}]_j. \tag{8}$$

IBU implements regularization in two ways. First, through early stopping in combination with a smooth prior. For instance, starting from $\vec{p}^{(0)} = \frac{1}{C}$ and stopping before Eq. 8 converges will maintain the smoothness of $\vec{p}^{(0)}$ to some degree. Second, the intermediate estimates $\vec{p}^{(k)}$ are smoothed with a low-order polynomial.

The above algorithms do not specify the feature transformation $f(\vec{x}) \in \{1, \dots, F\}$ through which $\vec{q}$ and M are defined; they solely focus on the estimation of $\vec{p}$ from any given $\vec{q}$ and M. In this sense, these algorithms are open to any feature transformation. Physicists have proposed

- to bin a single feature that is well correlated with the target quantity [2],
- to cluster the features in order to map instances to cluster indices [6],
- or to optimally partition the feature space by means of decision trees [4]

in order to obtain histograms $\vec{q}$ which represent the data sample.

3 Quantification

In the following, we show that several algorithms from quantification literature are indeed instances of the unified framework we have presented above. A summary of these findings is displayed in Tab. 1. We prove the correctness of our unifying notation in the Appendix.

Table 1. Algorithms for unfolding and quantification within the framework of Eq. 4.

	loss function $\mathcal{L}$	regularizer r	feature transformation f
RUN [3,2]	$\sum_{i=1}^{d} [M\vec{p}]_i - \bar{q}_i \ln[M\vec{p}]_i$	$\frac{1}{2}(T\vec{p})^2$	not specified / any
SVD [14]	$\left\| \frac{\vec{q} - M\vec{p}}{\vec{w}} \right\|_2^2$	$\frac{1}{2}(T\vec{p})^2$	not specified / any
IBU [8,7]	expectation maximization	smoothing	not specified / any
ACC [10,15]	$\|\vec{q} - M\vec{p}\|_2^2$	none	$\delta_{\arg\max_i [h(\vec{x})]_i}$
PACC [1]	$\|\vec{q} - M\vec{p}\|_2^2$	none	$h(\vec{x})$
ReadMe [15]	$\|\vec{q} - M\vec{p}\|_2^2$	none	$\delta_{\vec{x}=(X_1,\dots,X_{2^d})}$
HDx [13]	$\frac{1}{d}\sum_{i=1}^{d} \mathrm{HD}_i(\vec{q}, M\vec{p})$	none	$(\delta_{b(\vec{x};1)}, \dots, \delta_{b(\vec{x};d)})$
HDy [13]	$\frac{1}{d}\sum_{i=1}^{d} \mathrm{HD}_i(\vec{q}, M\vec{p})$	none	$(\delta_{b(h(\vec{x});1)}, \dots, \delta_{b(h(\vec{x});C)})$
CC [10]	none (assume $M = \mathbb{I}$)	none	$\delta_{\arg\max_i [h(\vec{x})]_i}$
PCC [1]	none (assume $M = \mathbb{I}$)	none	$h(\vec{x})$

Namely, our framework from Eq. 4 accommodates the following algorithms:

Adjusted Classify and Count (ACC) [10,15] Hopkins and King [15] present a method that extends the binary adjustment by Forman [10] to multi-class settings. Their extension represents a data sample as the counts of classification outcomes $\arg\max_i [h(\vec{x})]_i$, as returned by a multi-class classifier $h : \mathcal{X} \to \mathbb{R}^C$. In this case, M is simply the normalized confusion matrix of h, as estimated on held-out training data. Hopkins and King [15] propose to solve Eq. 3 via constrained least squares regression, hence

$$\mathcal{L}^{\text{ACC}}(\vec{p};\, \vec{q}, M) \;=\; \|\vec{q} - M\vec{p}\|_2^2 \tag{9}$$

and no regularization is employed.

Others [17,16] have proposed to solve Eq. (3) through matrix inversion,

$$\vec{p}^{\,\text{inv}} = M^{-1}\vec{q}.$$

However, there is no guarantee that M is indeed invertible. Therefore, $\vec{p}^{\,\text{inv}}$ might be undefined and the method by Hopkins and King [15] should be the prefered multi-class version of ACC.

Probabilistic ACC (PACC) [1] This method employs the same adjustment as ACC, hence the same loss. However, PACC averages soft classifications $h(\vec{x}) \in \mathbb{R}^C$ instead of counting the crisp outcomes $\arg\max_i [h(\vec{x})]_i$.

ReadMe [15] Building on the multi-class version of ACC, ReadMe employs the loss function from Eq. 9. However, ReadMe transforms the features in a unique way that is motivated in text mining. In this application area, instances $\vec{x}$ are often represented as bags of words, i.e. by sparse indicator vectors $\{0,1\}^d$ for a vocabulary of size d. In ReadMe, $\vec{q}$ is a histogram over all 2^d possible incarnations X_i of these indicator vectors, i.e.

$$\begin{aligned} f^{\text{ReadMe}}(\vec{x}) \;&=\; \delta_{\vec{x}=(X_1,\ldots,X_{2^d})}, \\ \text{where} \quad [\delta_{a=(a_1,\ldots,a_n)}]_i \;&=\; \begin{cases} 1 & \text{if } a = a_i, \\ 0 & \text{otherwise} \end{cases}. \end{aligned} \tag{10}$$

Since such a representation is only feasible with small d, ReadMe produces multiple estimates, each of which employs a different and small random selection of words. Finally, all of these estimates are averaged.

HDx [13] In this method, each feature is separately binned and a data sample is represented as a concatenation of all feature-wise histograms

$$f(\vec{x}) \;=\; (\delta_{b(\vec{x};1)}, \ldots, \delta_{b(\vec{x};d)}), \tag{11}$$

where $b(\vec{x}; i)$ is a binning function which maps the feature value $[\vec{x}]_i$ to the corresponding bin index $\{1, \ldots, B_i\}$.

The loss is measured as the average of feature-wise Hellinger distances,

$$\mathcal{L}(\vec{p};\, M, \vec{q}) \;=\; \frac{1}{d}\sum_{i=1}^{d} \mathrm{HD}_i(\vec{q},\, M\vec{p}), \tag{12}$$

$$\text{where} \quad \mathrm{HD}_i(\vec{q},\, M\vec{p}) \;=\; \sqrt{\sum_{j=1+\sum_{k=1}^{i-1} B_k}^{\sum_{k=1}^{i} B_k} \left(\sqrt{[\vec{q}]_j} - \sqrt{[M\vec{p}]_j}\right)^2}. \tag{13}$$

HDy [13] Originally, HDy has been proposed for binary quantification only. However, we can easily extend the method to the multi-class setting. In this setting, HDy replaces the separated binning of features $b(\vec{x}, i)$ in HDx with a separated binning of class-wise classifier outputs $b(h(\vec{x}), i)$. All other aspects of HDx are maintained.

(Probabilistic) Classify and Count (PCC/CC) [10,1] We also conceive these non-adjusted methods, which simply return $\vec{q}$ as their estimates for $\vec{p}$, as instances of our framework. Strictly speaking, CC and PCC do not require the minimization of a loss function. More loosely speaking, however, their disregard of M can be understood as the assumption of a perfect classifier, so that $M = \mathbb{I}$ is the identity matrix. Under this assumption, the least squares loss from Eq. 9 leads to the estimate $\vec{p}^{\,\mathrm{CC}} = \vec{q}$ and we can understand this estimate as an instance of Eq. 4.

Regarding $f(\vec{x})$, CC employs the feature transformation of ACC and PCC employs the feature transformation of PACC.

4 Conclusion and Outlook

We have presented the unfolding algorithms RUN, SVD, and IBU and the quantification algorithms ACC, PACC, ReadMe, HDx, HDy, CC, and PCC within a common framework. These algorithms differ in terms of the loss functions, the regularizers, and the feature transformations they employ.

Our unification demonstrates the similarity between the problems that are approached in unfolding and in quantification literature. Due to this similarity, we conceive adaptations of quantification algorithms to physics problems as a valuable endeavor for future work. Likewise, we suggest to adapt unfolding algorithms to problems outside of physics.

References

1. Bella, A., Ferri, C., Hernández-Orallo, J., Ramírez-Quintana, M.J.: Quantification via probability estimators. In: Int. Conf. on Data Mining. pp. 737–742. IEEE (2010). https://doi.org/10.1109/ICDM.2010.75
2. Blobel, V.: Unfolding methods in high-energy physics experiments. Tech. rep., CERN (1985). https://doi.org/10.5170/CERN-1985-009.88
3. Blobel, V.: An unfolding method for high energy physics experiments. In: Adv. Stat. Tech. in Part. Phys. pp. 258–267 (2002)

4. Börner, M., Hoinka, T., Meier, M., Menne, T., Rhode, W., Morik, K.: Measurement/simulation mismatches and multivariate data discretization in the machine learning era. In: Astron. Data Anal. Softw. and Syst. pp. 431–434. ASP Conference Series, Astronomical Society of the Pacific (2017)
5. Bunse, M.: Unification of algorithms for quantification and unfolding. In: Workshop on Mach. Learn. for Astropart. Phys. and Astron. Gesellschaft für Informatik e.V. (2022), to appear
6. Bunse, M., Piatkowski, N., Morik, K., Ruhe, T., Rhode, W.: Unification of deconvolution algorithms for Cherenkov astronomy. In: Int. Conf. on Data Sci. and Adv. Anal. pp. 21–30. IEEE (2018). https://doi.org/10.1109/DSAA.2018.00012
7. D'Agostini, G.: A multidimensional unfolding method based on Bayes' theorem. Nucl. Instr. and Meth. in Phys. Res. Sect. A **362**(2-3), 487–498 (1995)
8. D'Agostini, G.: Improved iterative Bayesian unfolding. arXiv:abs/1010.0632 (2010)
9. Firat, A.: Unified framework for quantification. arXiv:abs/1606.00868 (2016)
10. Forman, G.: Quantifying counts and costs via classification. Data Mining and Knowl. Discov. **17**(2), 164–206 (2008). https://doi.org/10.1007/s10618-008-0097-y
11. Gao, W., Sebastiani, F.: From classification to quantification in tweet sentiment analysis. Soc. Netw. Anal. and Mining **6**(19), 1–22 (2016)
12. González, P., Castaño, A., Chawla, N.V., del Coz, J.J.: A review on quantification learning. ACM Comput. Surv. **50**(5), 74:1–74:40 (2017). https://doi.org/10.1145/3117807
13. González-Castro, V., Alaíz-Rodríguez, R., Alegre, E.: Class distribution estimation based on the Hellinger distance. Inf. Sci. **218**, 146–164 (2013). https://doi.org/10.1016/j.ins.2012.05.028
14. Hoecker, A., Kartvelishvili, V.: SVD approach to data unfolding. Nucl. Instr. and Meth. in Phys. Res. Sect. A **372**(3), 469–481 (1996)
15. Hopkins, D.J., King, G.: A method of automated nonparametric content analysis for social science. Amer. J. of Polit. Sci. **54**(1), 229–247 (2010). https://doi.org/10.1111/j.1540-5907.2009.00428.x
16. McLachlan, G.J.: Discriminant analysis and statistical pattern recognition. Wiley (1992)
17. Vucetic, S., Obradovic, Z.: Classification on data with biased class distribution. In: Eur. Conf. on Mach. Learn. pp. 527–538. Springer (2001). https://doi.org/10.1007/3-540-44795-4_45

A Proofs

We now detail the mapping from the original algorithms to our unified notation, to formally prove that our framework is consistent with the original proposals.

Regularized Unfolding (RUN) The loss function we present in Eq. 5 is a verbatim statement by Blobel [2, Eqs. (2.29), and (2.26)]. The original algorithm treats the elements of $\vec{p}$ as B-spline coefficients; however, a more recent version by the same author [3] employs histograms, which are consistent with our Eq. 2. Due to this change "the second derivative in bin j is proportional to $x_{j-1} - 2x_j + x_{j+1}$" [3], where $x_i = [\vec{p}]_i$. This derivative defines the regularization term from Eq. 6. □

Unfolding via Singular Value Decomposition (SVD) The loss function we present in Eq. 7 and the regularization term from Eq. 6 are verbatim statements by Hoecker and Kartvelishvili [14, Eqs. (29), (37), and (38)]. □

Iterative Bayesian Unfolding (IBU) D'Agostini [7, Eqs. (3), and (4)] estimates $[\vec{p}^{(k)}]_i$ as

$$\frac{1}{\epsilon_i} \sum_{j=1}^{n_E} n(E_j) \cdot \frac{P(E_j \mid C_i) \cdot P_0(C_i)}{\sum_{l=1}^{n_C} P(E_j \mid C_l) \cdot P_0(C_l)},$$

where we identify our notation as $F = n_E$, $C = n_C$, $M_{ij} = P(E_j \mid C_i)$, and $[\vec{p}^{(k-1)}]_i = P_0(C_i)$. In the original algorithm, $n(E_j) \in \mathbb{N}$ is the count observed in the j-th bin, i.e. $n(E_j) = N \cdot [\vec{q}]_j$. Moreover, $\epsilon_i > 0$ is an acceptance factor, which models the probability that an existing instance of class i is indeed part of the sample—and not hidden due to measurement complications. Setting $\epsilon_i = N$, we obtain $[\vec{q}]_j = \frac{n(E_j)}{\epsilon_i}$, which is consistent with our Eq. 8.

For regularization, D'Agostini [7] proposes to "smooth the results of the unfolding before feeding them in the next step", for instance "by a polynomial fit of 3$^{\text{rd}}$ degree" or by another low-order polynomial. □

Adjusted Classify and Count (ACC) Hopkins and King [15, Eq. (4)] marginalize over the true labels $D \in \{1, \ldots, J\}$ to yield the distribution of class predictions $\widehat{D}$,

$$P(\widehat{D} = j) \;=\; \sum_{j'=1}^{J} P(\widehat{D} = j \mid D = j') P(D = j).$$

The authors note that "this expression represents a set of J equations [...] that can be solved for the J elements in $P(D)$". Accordingly, we identify our notation as $\vec{p} = P(D)$, $\vec{q} = P(\widehat{D})$, and $M = P(\widehat{D} \mid D)$ in their presentation. To solve this set of equations, the authors propose a "standard constrained least squares to ensure that elements of $P(D)$ are each in [0,1] and collectively sum up to 1". This proposal defines the least squares loss from Eq. 9 and matches our constraints in Eq. 4. □

Note that Hopkins and King have developed their method independently of Forman's binary ACC. However, the basis of their work is precisely the adjustment by Forman [10, Eq. (1)], as can be seen in Hopkins and King [15, Eq. (3)]. Therefore, we call their method "multi-class ACC".
The other multi-class extension of ACC, $\vec{p}^{\text{inv}}$, is presented in McLachlan [16, Eq. (2.3.4)] and in Vucetic and Obradovic [17, Eq. (3)].

Probabilistic ACC (PACC) The essential proposal by Bella et al. [1] is to replace hard classifications $\arg\max_i [h(\vec{x})]_i$ with probabilistic ones $h(\vec{x}) \in \mathbb{R}^C$; their adjustment is the same as in binary ACC. By applying this proposal to multi-class ACC [15], we obtain a multi-class PACC which employs the loss from Eq. 9. □

ReadMe Building on their multi-class design of ACC, Hopkins and King [15, Eq. (6)] set up a matrix equation $P(\mathbf{S}) = P(\mathbf{S} \mid D)P(D)$, which maps to our notation as $\vec{q} = P(\mathbf{S}) \in \mathbb{R}^{2^d}$, $M = P(\mathbf{S} \mid D) \in \mathbb{R}^{2^d \times C}$, and $\vec{p} = P(D) \in \mathbb{R}^C$. The authors note that "$P(\mathbf{S})$ is the probability of each of the 2^K possible word stem profiles" with $K = d$ being the number of word stems. To estimate this probability, "we merely compute the proportion of documents observed with each pattern of word profiles". This computation leads to a histogram

$$\vec{q} = \frac{1}{N} \sum_{i=1}^{N} \delta_{\vec{x}_i = (X_1, \ldots, X_{2^d})},$$

which is consistent with our Eqs. 2 and 10. □

HDx González-Castro et al. [13, Eq. (9)] minimize the average of feature-wise Hellinger distances, as we have stated in Eq. 12. They present the distance with respect to a single feature j, binned into b bins, as

$$\sqrt{\sum_{i=1}^{b} \left(\sqrt{\frac{|V_{j,i}|}{|V|}} - \sqrt{\frac{|U_{j,i}|}{|U|}} \right)^2},$$

where $|U|$ is the total number of instances and $|U_{j,i}|$ is the number of instances whose feature j is mapped to the i-th bin [13, Eq. (10)]. $|V|$ and $|V_{j,i}|$ are the numbers of instances that are to be expected under class prevalence values $\vec{p}$, hence

$$\frac{|V_{j,i}|}{|V|} = [M\vec{p}]_{i + \sum_{k=1}^{j-1} B_k},$$

where $\sum_{k=1}^{j-1} B_k$ is the offset of the histogram of feature j within our concatenation of feature-wise histograms. Using the product $M\vec{p}$ at this point is consistent with the binary conception that is proposed by González-Castro et al. [13, Eq. (12)]. □

HDy The original HDy [13, Eqs. (13) and (14)] only addresses binary quantification. For this case, however, the only change with respect to HDx is that HDy employs soft classifier outputs $h(\vec{x})$ instead of features $\vec{x}$. A straightforward extension to the multi-class setting is therefore to bin the class-wise outputs $[h(\vec{x})]_i$ separately, as HDx does in case of features and as we propose in our presentation of HDy. □

(Probabilistic) Classify and Count (PCC/CC) Let $M = \mathbb{I}$. Recognize that the global minimum of the least squares loss,

$$\min_{\vec{p}} \|\vec{q} - M\vec{p}\|_2^2 \;=\; 0,$$

is now attained if and only if $\vec{p} = \vec{q}$. Therefore, under the assumption $M = \mathbb{I}$, the unique minimizer of the least squares loss is $\vec{q}$. In this sense, PCC and CC are proper instances of our framework. □

Class Prior Estimation under Covariate Shift: No Problem?

Dirk Tasche[0000−0002−2750−2970]

Independent researcher
dirk.tasche@gmx.net

Abstract. We show that in the context of classification the property of source and target distributions to be related by covariate shift may be lost if the information content captured in the covariates is reduced, for instance by dropping components or mapping into a lower-dimensional or finite space. As a consequence, under covariate shift simple approaches to class prior estimation in the style of classify and count with or without adjustment are infeasible. We prove that transformations of the covariates that preserve the covariate shift property are necessarily sufficient in the statistical sense for the full set of covariates. A probing algorithm as alternative approach to class prior estimation under covariate shift is proposed.

Keywords: Covariate shift · Prior probability shift · Quantification · Class prior estimation · Prevalence estimation · Sufficiency

1 Introduction

Class prior estimation (also known as quantification, class distribution estimation, prevalence estimation etc.) may be considered one of the tasks referred to under the general term domain adaptation.

Domain adaptation means adapting algorithms designed for a source (training) dataset (also distribution or domain) to a target (test) dataset. The source and target distributions may be different, a phenomenon which is called *dataset shift*. In this paper, attention is restricted to 'unsupervised' domain adaptation. This term refers on the one hand to the situation where under the source distribution all events and realisations of random variables – including the target (label) variable – are observable such that in principle the whole distribution can be estimated. On the other hand under the target distribution only the marginal distribution of the covariates (features) can be observed, via realisations of the covariates. The target distribution class labels cannot be observed at all or only with delay.

Moreno-Torres et al. [21] proposed the following popular taxonomy of types of dataset shift:

– Covariate shift: Source and target posterior class probabilities are the same but source and target covariate distributions may be different.

- Prior probability shift (label shift [20], global drift [14]): Source and target class-conditional covariate distributions are the same but source and target prior class probabilities may be different.
- Concept shift: Source and target covariate distributions are the same but source and target posterior class probabilities may be different, or source and target prior class probabilities are the same but source and target class-conditional covariate distributions may be different.
- Other shift: Any dataset shift not captured by the previous types.

Covariate shift and prior probability shift are described in constructive terms. Based on their defining properties source and target distributions are fully specified. For this reason, a host of focussed literature is available for these two types of dataset shift. In contrast, it is hardly possible to make specific statements about the two other types of shift such that the literature on these types is much more diverse and hard to capture.

In this paper we focus on covariate shift and a classification setting. We choose a measure-theoretic approach that is particularly suitable for this context as it facilitates a rigorous joint treatment of continuous and discrete random variables, or covariates and class labels more specifically. We work in the same binary classification setting as Ben-David et al. [3] and Johansson et al. [16]. Like Ben-David et al. and Johansson et al., we focus on the binary case but the results are easily generalised to the multi-class case.

Prior probability shift is robust in the following sense: If the set of covariates is transformed in a way that reduces the information reflected by them (e.g. by dropping components or mapping into a lower-dimensional or finite space) then the resulting source and target joint distributions of covariates and labels are still related by prior probability shift. As a consequence, simple approaches to class prior estimation under prior probability shift can be designed which avoid the need to estimate the full class-conditional covariate distributions.[1] The primary example for such an approach is the 'confusion matrix method' (Gart and Buck [9]; Saerens et al. [23]; 'adjusted count' in Forman [8]).

We show by examples and by theoretical analysis that such robustness is not displayed by covariate shift. Under the condition that the target distribution is absolutely continuous with respect to the source distribution, we prove that a set of covariates passes on the covariate shift property if and only if the transformed set of covariates is 'sufficient' in the sense of Adragni and Cook [1] and Tasche [26] for the untransformed set under the source distribution. The result refines an observation of Johansson et al. [16] who found that covariate shift was inherited "only if" the transformation was invertible.

An important consequence of this finding is that in general for class prior estimation under covariate shift, simplification in the sense of reducing the complexity of the covariate set is not a viable path because the covariate shift property of identical posterior class probabilities between source and target distributions

[1] The simplification may come at a cost of increased variance of the estimator (Tasche [27]).

might get lost. We point to a potential alternative approach, based on the so-called 'probing' method of Langford and Zadrozny [19].

The plan of this paper is as follows: We introduce the assumptions and the notation for this paper in Section 2. In Section 3 we give examples of how loss of information may affect the covariate shift property. The main result (Theorem 1 of this paper) is presented in Section 4 while Section 5 provides some comments on the result. A proposal for applying 'probing' to class prior estimation is made in Section 6. The paper concludes with a short summary in Section 7.

2 Assumptions and Notation

In this paper, we work only at population (distribution) level as this level is appropriate for the design of estimators and predictors as well as the study of their fundamental properties. A detailed treatment of the intricacies of sample properties is not needed.

We follow the example of Scott [24] who introduced consistent concepts and notation for appropriately dealing with the classification setting we need. As the concept of information plays a more important role in this paper than in Scott's, we dive somewhat deeper into the measure-theoretic details of the setting than Scott.

2.1 Setting for Binary Classification in the Presence of Dataset Shift

We introduce a measure-theoretic setting, expanding the setting of Scott [24] and adapting the approach of Holzmann and Eulert [15] and Tasche [26]. Phrasing the context in measure theory terms is particularly efficient when random variables with continuous and discrete distributions are studied together like in the case of binary or multi-class classification. Moreover, the measure-theoretic notion of σ-algebras allows for the convenient description of differences in available information.

We use the following population-level description of the binary classification problem in terms of measure theory. See standard textbooks on probability theory like Billingsley [4] or Klenke [17] for formal definitions and background of the notions introduced in Assumption 1.

Assumption 1 *$(\Omega, \mathcal{A})$ is a measurable space. The* source distribution *P and the* target distribution *Q are probability measures on $(\Omega, \mathcal{A})$. An event $A_1 \in \mathcal{A}$ with $0 < P[A_1] < 1$ and a sub-σ-algebra $\mathcal{H} \subset \mathcal{A}$ with $A_1 \notin \mathcal{H}$ are fixed. $A_0 = \Omega \setminus A_1$ is the complementary event of A_1 in Ω.*

In the literature, P is also called 'training distribution' while Q is also referred to as 'test distribution'.

Interpretation. The elements ω of Ω are objects (or instances) with class (label) and covariate (feature) attributes. $\omega \in A_1$ means that ω belongs to class

1 (or the positive class). $\omega \in A_0$ means that ω belongs to class 0 (or the negative class).

The σ-algebra $\mathcal{A}$ of events $M \in \mathcal{A}$ is a collection of subsets M of Ω with the property that they can be assigned probabilities $P[M]$ and $Q[M]$ in a logically consistent way. In the literature, thanks to their role of reflecting the available information, σ-algebras are sometimes also called *information set* (Holzmann and Eulert [15]). In the following, we use both terms exchangeably.

Binary classification problem. The sub-σ-algebra $\mathcal{H} \subset \mathcal{A}$ contains the events which are observable at the time when the class label of an object ω has to be predicted. Since $A_1 \notin \mathcal{H}$, then the class of an object may not yet be known. It can only be predicted on the basis of the events $H \in \mathcal{H}$ which are assumed to reflect the features of the object.

Dataset shift. We denote by $\mathcal{H}_A$ the minimal sub-σ-algebra of $\mathcal{A}$ containing both $\mathcal{H}$ and $\sigma(\{A_1\}) = \{\emptyset, A_1, A_0, \Omega\}$, i.e. $\mathcal{H}_A = \sigma\big(\mathcal{H} \cup \sigma(\{A_1\})\big)$. The σ-algebra $\mathcal{H}_A$ can be represented as

$$\mathcal{H}_A \;=\; \big\{(A_1 \cap H_1) \cup (A_0 \cap H_0) : H_1, H_0 \in \mathcal{H}\big\}. \tag{1}$$

A standard assumption in machine learning is that source and target distribution are the same, i.e. $P = Q$. The situation where $P[M] \neq Q[M]$ holds for at least one $M \in \mathcal{H}_A$ is called *dataset shift* (Moreno-Torres et al. [21], Definition 1).

Class prior estimation. Under dataset shift as defined above, typically the prior probabilities $P[A_1]$ of the positive class in the source distribution (assumed to be observable) and $Q[A_1]$ in the target distribution (assumed to be unknown or known with delay only) are different. Class prior estimation in the binary classification context of Assumption 1 is the task to estimate $Q[A_1]$, based on observations from P (the entire source distribution) and from[2] $Q|\mathcal{H}$ (the target distribution of the covariates, also called features).

Notation. Denote by $\mathbf{1}_M$ the indicator function of an event M, i.e. $\mathbf{1}_M(\omega) = 1$ if $\omega \in M$ and $\mathbf{1}_M(\omega) = 0$ if $\omega \notin M$.

If X is a real-valued random variable on a probability space $(\Omega, \mathcal{F}, P)$ and $\mathcal{G} \subset \mathcal{F}$ is a sub-σ-algebra of $\mathcal{F}$, then a random variable Ψ is called *expectation of X conditional on $\mathcal{G}$* (see, e.g., Definition 8.11 of Klenke [17]) if it has the following two properties:
(i) Ψ is $\mathcal{G}$-measurable.
(ii) For all events $G \in \mathcal{G}$ it holds that $E_P[\mathbf{1}_G\, X] = E_P[\mathbf{1}_G\, \Psi]$.

In the following, we use the usual shorthand notation $\Psi = E_P[X \,|\, \mathcal{G}]$. In the case of an indicator function of an event $F \in \mathcal{F}$, the conditional expectation $E_P[\mathbf{1}_F \,|\, \mathcal{G}]$ is called *probability of F conditional on $\mathcal{G}$* and denoted by $P[F \,|\, \mathcal{G}]$.

2.2 Reconciliation of Machine Learning and Measure Theory Settings

The setting of Assumption 1 is similar to a standard setting for binary classification in the machine learning and pattern recognition literature (see e.g. Scott [24] or Devroye et al. [7]):

[2] $Q|\mathcal{H}$ stands for the measure Q with domain restricted to $\mathcal{H}$.

Typically a random vector (X, Y) is studied, where X stands for the covariates of an object and Y stands for its class. X is assumed to take values in a feature space $\mathfrak{X}$ (often $\mathfrak{X} = \mathbb{R}^d$) while Y takes either the value 0 (or -1) or the value 1 (for the positive class).

Standard formulation of the binary classification problem: Predict the value of Y from X or make an informed decision on the occurence or non-occurrence of the event $Y = 1$ despite only being able to observe the values of X.

This is captured by the measure-theoretic setting of Assumption 1: Assume that X and Y map Ω into $\mathfrak{X}$ and $\{0, 1\}$ respectively. Choose $\mathcal{H} = \sigma(X)$ (the smallest sub-σ-algebra of $\mathcal{A}$ such that X is measurable) and $A_1 = \{Y = 1\} = \{\omega \in \Omega : Y(\omega) = 1\}$.

In many machine learning papers, the image (or pushforward) measure of P (or Q if it refers to the target distribution) under the mapping (X, Y) (see Definition 1.98 of Klenke [17]) is denoted by $p(x, y)$.

Often no probability space is specified but only samples $(x_1, y_1), \ldots, (x_n, y_n)$ of realisations of (X, Y) from the source distribution and $x_{n+1}, \ldots, x_{n+m}$ of realisations of X from the target distribution are assumed to be given. This context is sometimes called 'unsupervised domain adaptation'. Usually the samples are assumed to have been generated through i.i.d. drawings from some population distributions which may be identified with $(\Omega, \mathcal{A}, P)$ and $(\Omega, \mathcal{A}, Q)$ as described above.

2.3 More on Dataset Shift

Arguably, the two most important special cases of dataset shift are the following, in the terms introduced in Assumption 1:

- *Covariate shift* (Moreno-Torres et al. [21], Definition 3; Storkey [25], Section 5):
  In this case, $P|\mathcal{H} \neq Q|\mathcal{H}$ holds but $P[A_1 \,|\, \mathcal{H}] = Q[A_1 \,|\, \mathcal{H}]$, i.e. the posterior probabilities under P and Q are the same but the covariate distributions may be different.
- *Prior probability shift* (Moreno-Torres et al. [21], Definition 2; Storkey [25], Section 6):
  In this case, we have $P[A_1] \neq Q[A_1]$ but $P[H \,|\, A_i] = Q[H \,|\, A_i]$, $i \in \{0, 1\}$, for all $H \in \mathcal{H}$, i.e. the class-conditional covariate source and target distributions are the same but the unconditional class prior probabilities may be different.

Covariate shift and prior probability shift are similar in the sense that in both cases one of the conditional distributions (of A_1 conditional on $\mathcal{H}$ and of $\mathcal{H}$ conditional on $\sigma(\{A_1\})$ respectively) are invariant between P and Q, and at least one pair of the marginal distributions (of $\mathcal{H}$ and $\sigma(\{A_1\})$ respectively) are different.

Thanks to the invariance assumptions on the conditional distributions in prior probability shift and in covariate shift, these two types of dataset shift are relatively easily amenable to mathematical treatment and, therefore, have

received considerable attention by researchers. See e.g. Quiñonero-Candela et al. [22] for covariate shift and Caelen [5] for prior probability shift, as well as the references therein.

Note that the definition of dataset shift in Section 2.1 explicitly mentions an associated set of covariates (features, represented through the sub-σ-algebra $\mathcal{H}$). In Section 3, we are going to look closer at the question whether or not the covariate shift property is preserved in the relationship between source and target distribution if the amount of information reflected by the set of covariates is reduced. Formally, the question is phrased as follows:

Under Assumption 1, if $\mathcal{G} \subset \mathcal{H}$ is another sub-σ-algebra of $\mathcal{A}$, does then $P[A_1 \,|\, \mathcal{H}] = Q[A_1 \,|\, \mathcal{H}]$ imply $P[A_1 \,|\, \mathcal{G}] = Q[A_1 \,|\, \mathcal{G}]$?

3 Covariate Shift is Fragile

In theory, class prior estimation under covariate shift is straightforward. Assume that the source distribution P and the target distribution Q are related through covariate shift as defined in Section 2.3. Then by the law of total probability and the fact that $P[A_1 \,|\, \mathcal{H}] = Q[A_1 \,|\, \mathcal{H}]$, the prior class probability $Q[A_1]$ of the positive class can be represented as

$$Q[A_1] \;=\; E_Q\big[P[A_1 \,|\, \mathcal{H}]\big]. \tag{2}$$

As mentioned in Section 2.1, both $P[A_1 \,|\, \mathcal{H}]$ and $Q[A_1 \,|\, \mathcal{H}]$ typically are observable at the time when $Q[A_1]$ is to be estimated such that $Q[A_1]$ in principle can be calculated by means of (2). In the literature on class prior estimation, the approach based on (2) is known as 'probability estimation & average (P & A)' (Bella et al. [2]) or 'probabilistic classify & count (PCC)' (González et al. [11]).

Unfortunately, (2) may not work well in practice:

- Card and Smith [6] observed that poor calibration of the estimates of the posterior class probabilities would entail poor results for the PCC prior probability estimates.
- At a more fundamental level, Storkey ([25], Section 5.1) pointed out that the probability masses of the covariates might be quite differently located under the source and target distributions. As a consequence, an estimate of $P[A_1 \,|\, \mathcal{H}]$ made under the source distribution P might turn out to be rather biased in those regions of the covariate space to which the target distribution Q attributes most mass. This problem can be mitigated by 'importance weighting' which, however, may significantly complicate the estimation procedure.

Due to these issues, it is tempting to try to avoid the potentially difficult estimation of the posterior class probability $P[A_1 \,|\, \mathcal{H}]$ which is conditioned on the full covariate information set $\mathcal{H}$, by mimicking the simplification achieved through the confusion matrix method (Saerens et al. [23]; also called 'adjusted count' in Forman [8]) under prior probability shift.

Adapting the confusion matrix method to covariate shift would work as follows: Fix some hard (i.e. taking either the value 0 or the value 1) classifier which is a function of the covariates and therefore $\mathcal{H}$-measurable. We can identify the classifier with an event $H \in \mathcal{H}$ which specifies the range of the covariates on which a positive class label is predicted. If the source distribution P and the target distribution Q are related by covariate shift for the simple information set $\mathcal{G} = \{\emptyset, H, \Omega \setminus H, \Omega\} \subset \mathcal{H}$ then the following special case of (2) applies:

$$Q[A_1] \stackrel{?}{=} Q[H]\, P[A_1 \,|\, H] + (1 - Q[H])\, P[A_1 \,|\, (\Omega \setminus H)]. \tag{3}$$

Eq. (3) appears to suggest a simple and efficient approach to class prior estimation under covariate shift which avoids the potentially difficult problem to estimate $P[A_1 \,|\, \mathcal{H}]$ for the more complex information set $\mathcal{H}$.

But can we always find a classifier (observable event) H such that the following condition for covariate shift with respect to $\mathcal{G} = \{\emptyset, H, \Omega \setminus H, \Omega\}$ and, as a consequence, also (3) hold true?

$$Q[A_1 \,|\, H] = P[A_1 \,|\, H] \quad \text{and} \quad Q[A_1 \,|\, (\Omega \setminus H)] = P[A_1 \,|\, (\Omega \setminus H)]. \tag{4}$$

The question mark in (3) is meant to suggest that the answer is 'no'. This is illustrated with the following example.

Example 1. We revisit the binormal model with equal variances as an example that fits into the setting of Assumption 1. The source distribution P is defined by specifying the marginal distribution of $Y = \begin{cases} 1, & \text{on } A_1 \\ 0, & \text{on } A_0 \end{cases}$, with $\mathrm{P}[A_1] = p \in (0,1)$, and defining the class-conditional distributions of the covariate X given Y as normal distributions with equal variances:

$$P[X \in \cdot \,|\, A_1] = \mathcal{N}(\nu, \sigma^2) \quad \text{and} \quad P[X \in \cdot \,|\, A_0] = \mathcal{N}(\mu, \sigma^2). \tag{5}$$

In (5), we assume that $\mu < \nu$ and $\sigma > 0$. The unconditional distribution of X then is a mixture with weight p of the two normal distributions.

The posterior class probability $P[A_1 \,|\, \mathcal{H}]$ for $\mathcal{H} = \sigma(X)$ in this setting is given by $P[A_1 \,|\, \mathcal{H}] = \big(1 + \exp(a\,X + b)\big)^{-1}$, with $a = \frac{\mu - \nu}{\sigma^2} < 0$ and $b = \frac{\nu^2 - \mu^2}{2\,\sigma^2} + \log\left(\frac{1-p}{p}\right)$. For the target distribution Q, we only specify the marginal distribution of the covariate X as another normal distribution with mean $E_Q[X] = \tau$ and variance $\mathrm{var}_Q[X] = \sigma^2 + p\,(1-p)\,(\mu - \nu)^2$ such that the variance of X under Q matches the variance of X under P.
Under covariate shift, then by (2) it holds for the target prior class probability $Q[A_1]$ that

$$Q[A_1] \;=\; E_Q\left[\big(1 + \exp(a\,X + b)\big)^{-1}\right]. \tag{6}$$

To illustrate the effect of simplification as suggested by (3), we define a family of classifiers $H_x \;=\; \{X > x\}$ for thresholds $x \in \mathbb{R}$.

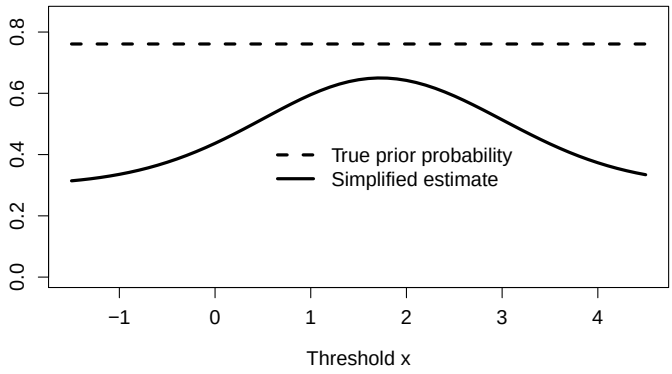


Fig. 1. Illustration of Example 1, with parameters $\mu = 0$, $\nu = 1.5$, $\sigma = 1$, $p = 0.3$ and $\tau = 2.5$ for the normal distributions and the mixture parameter p of the source distribution. Simplification of the covariate information set causes the covariate shift property to be lost as demonstrated by the failure to hit the true target prior $Q[A_1]$ with the simplified estimates according to (3).

Figure 1 shows the true target prior class probability $Q[A_1]$ according to (6) (constant, dashed line) and, for moving threshold x, 'pseudo' priors according to (3) (solid curve). As the pseudo priors do not match the true prior, the covariate shift property (4) must be violated for all information sets

$$\mathcal{G}_x = \{\emptyset, H_x, \Omega \setminus H_x, \Omega\} \subset \mathcal{H} = \sigma(X).$$

This is due to the loss of information compared to the full information set $\mathcal{H}$ associated with the covariate X. □

On the basis of Example 1, we can conclude that under Assumption 1, if $\mathcal{G} \subset \mathcal{H}$ is another sub-σ-algebra of $\mathcal{A}$, then $P[A_1 \,|\, \mathcal{H}] = Q[A_1 \,|\, \mathcal{H}]$ does not always imply $P[A_1 \,|\, \mathcal{G}] = Q[A_1 \,|\, \mathcal{G}]$, i.e. the covariate property may get lost if the amount of information represented by the covariates is reduced.

Information loss and subsequent loss of the covariate shift property can also be the consequence of deploying 'domain-invariant representations' (Johansson et al. [16], Section 4.1).

4 Covariate Shift and Statistical Sufficiency

In the following we will identify sufficient and necessary conditions for simplifications of covariate shift like (4) to hold. We will see that indeed it is almost impossible for (4) to be true if the information set $\mathcal{H}$ of Assumption 1 is large compared to the information set $\mathcal{G}$ on which (4) is based.

Definition 1. *Under Assumption 1, denote by $C_A(P, \mathcal{H})$ the set of all probability measures Q on $(\Omega, \mathcal{A})$ such that P and Q are related by covariate shift, i.e.*

$$C_A(P, \mathcal{H}) \;=\; \big\{Q \text{ probability measure on } (\Omega, \mathcal{A}) :\; P[A_1 \,|\, \mathcal{H}] = Q[A_1 \,|\, \mathcal{H}]\big\}.$$

Denote by $C_A^*(P, \mathcal{H})$ the set of all probability measures Q on $(\Omega, \mathcal{A})$ such that P and Q are related by covariate shift and Q is absolutely continuous[3] with respect to P on $\mathcal{H}$, i.e.

$$C_A^*(P, \mathcal{H}) \;=\; \big\{Q \in C_A(P, \mathcal{H}) : Q|\mathcal{H} \ll P|\mathcal{H}\big\}.$$

The following example illustrates the definition of $C_A(P, \mathcal{H})$ in some simple special cases.

Example 2. Consider the following three special cases for $\mathcal{H}$:

(i) If $A_1 \in \mathcal{H}$ then we have that $\mathcal{H} \supset \sigma(\{A\})$ and

$$C_A(P, \mathcal{H}) \;=\; \{\text{All probability measures on } (\Omega, \mathcal{A})\},$$

because in this case it holds that $P[A_1 \,|\, \mathcal{H}] = \mathbf{1}_{A_1} = Q[A_1 \,|\, \mathcal{H}]$.

(ii) If A_1 and $\mathcal{H}$ are independent under P and Q, it follows that

$$P[A_1 \,|\, \mathcal{H}] \;=\; P[A_1] \quad \text{and} \quad Q[A_1 \,|\, \mathcal{H}] \;=\; Q[A_1].$$

Hence we have $Q \in C_A(P, \mathcal{H})$ if and only if $P[A_1] = Q[A_1]$.

(iii) If $\mathcal{H} = \{\emptyset, \Omega\}$ we are in a special case of (ii). This implies

$$C_A(P, \mathcal{H}) \;=\; \big\{Q \text{ probability measure on } (\Omega, \mathcal{A}) :\; P[A_1] = Q[A_1]\big\}. \qquad \square$$

Note that in case (i) of Example 2, $P[A_1] \neq Q[A_1]$ is possible.

Remark 1. The set $C_A(P, \mathcal{H})$ contains all probability measures Q with the property that there is an event $\Omega_Q \in \mathcal{H}$ such that $Q[\Omega_Q] = 1$ and $P[\Omega_Q] = 0$, i.e. Q and P are mutually singular. Although in this case there is an $\mathcal{H}$-measurable random variable Ψ that is both a version of $P[A_1 \,|\, \mathcal{H}]$ and of $Q[A_1 \,|\, \mathcal{H}]$, it is impossible to completely learn Ψ from the source distribution P because no instances $\omega \in \Omega_Q$ can be sampled under P due to $P[\Omega_Q] = 0$. Hence the distributions Q which are singular to P are not of great theoretical interest. $C_A(P, \mathcal{H})$ may also contain probability measures Q with both absolutely continuous and singular components (with respect to P). In this case, it is not possible to completely learn Ψ from P either. Therefore, in the following the focus is on the distributions Q that are absolutely continuous with respect to P. $\square$

[3] Q is absolutely continuous with respect to P on $\mathcal{H}$ (expressed symbolically as $Q|\mathcal{H} \ll P|\mathcal{H}$) if $P[N] = 0$ for $N \in \mathcal{H}$ implies $Q[N] = 0$. By the Radon-Nikodym theorem (Klenke [17], Corollary 7.34) then there exists an $\mathcal{H}$-measurable non-negative function h such that $Q[H] = E_P[h\,\mathbf{1}_H]$ for all $H \in \mathcal{H}$. The function h is called density of Q with respect to P.

At first glance, one might guess that the tower property of conditional expectations (Klenke [17], Theorem 8.14) implies $C_A(P,\mathcal{H}) \subset C_A(P,\mathcal{G})$ if $\mathcal{G}$ is a sub-σ-algebra of $\mathcal{H}$. However, the following example shows that this is not true in general.

Example 3. Assume that $\mathcal{H} = \sigma(\mathcal{F} \cup \mathcal{G})$ for sub-σ-algebras $\mathcal{F}$, $\mathcal{G}$ of $\mathcal{A}$, with $A_1 \notin \mathcal{H}$. Assume further that $\mathcal{G}$ and $\sigma\big(\sigma(\{A_1\}) \cup \mathcal{F}\big)$ are independent under P and Q. Then it follows that $P[A_1 \,|\, \mathcal{H}] = P[A_1 \,|\, \mathcal{F}]$ and $Q[A_1 \,|\, \mathcal{H}] = Q[A_1 \,|\, \mathcal{F}]$.

Hence we have $Q \in C_A(P,\mathcal{H})$ if and only if $Q \in C_A(P,\mathcal{F})$. By case (ii) of Example 2, we have $Q \in C_A(P,\mathcal{G})$ if and only if $P[A_1] = Q[A_1]$. Hence, if there is a $Q \in C_A(P,\mathcal{F})$ with $P[A_1] \neq Q[A_1]$, we have an example showing that $C_A(P,\mathcal{H}) \not\subset C_A(P,\mathcal{G})$ may happen despite $\mathcal{G} \subset \mathcal{H}$. □

Example 3 demonstrates that the covariate shift property may get lost if components of the covariates are dropped. We continue with presenting sufficient criteria for covariate shift (Lemma 1) and inheritance of covariate shift (Proposition 1 below).

Lemma 1. *Under Assumption 1, assume further that Q is absolutely continuous with respect to P on $\mathcal{H}_A$ and that there is an $\mathcal{H}$-measurable density h of $Q|\mathcal{H}_A$ with respect to $P|\mathcal{H}_A$. Then it follows that $Q \in C_A^*(P,\mathcal{H})$.*

Proof. Fix any $H \in \mathcal{H}$. Then we obtain that

$$\begin{aligned} E_Q\big[\mathbf{1}_H\, P[A_1 \,|\, \mathcal{H}]\big] &= E_P\big[h\,\mathbf{1}_H\, P[A_1 \,|\, \mathcal{H}]\big] \\ &= E_P\big[E_P[h\,\mathbf{1}_{A_1 \cap H} \,|\, \mathcal{H}]\big] = E_P[h\,\mathbf{1}_{A_1 \cap H}] = Q[A_1 \cap H]. \end{aligned}$$

This implies $P[A_1 \,|\, \mathcal{H}] = Q[A_1 \,|\, \mathcal{H}]$. □

We are now going to point out connections between the notion of covariate shift and the following two concepts that have been considered in the literature in other contexts:

- Covariate shift with posterior drift (Scott [24]): Under Assumption 1, dataset shift between P and Q is more specifically called *covariate shift with posterior drift* if there is an increasing function $f : [0,1] \to \mathbb{R}$ such that it holds that

$$Q[A_1 \,|\, \mathcal{H}] \;=\; f\big(P[A_1 \,|\, \mathcal{H}]\big). \tag{7}$$

- Sufficiency [7,1,26]: Under Assumption 1, if $\mathcal{G} \subset \mathcal{H}$ is another sub-σ-algebra of $\mathcal{A}$ then $\mathcal{G}$ is called (statistically) *sufficient* for $\mathcal{H}$ with respect to A_1 if $P[A_1 \,|\, \mathcal{G}] = P[A_1 \,|\, \mathcal{H}]$ holds true.

Proposition 1. *Under Assumption 1, let P and Q be related by covariate shift with posterior drift such that (7) holds for an increasing function f. Assume that $\mathcal{G}$ is sufficient for $\mathcal{H}$ with respect to A_1 under the source distribution P and that Q is absolutely continuous with respect to P on $\mathcal{H}$. Then $Q[A_1 \,|\, \mathcal{G}] = f\big(P[A_1 \,|\, \mathcal{G}]\big)$ follows.*

Proof. Let $h \geq 0$ be a density of Q with respect to P on $\mathcal{H}$. Then, in particular, h is $\mathcal{H}$-measurable. For any $G \in \mathcal{G}$, we therefore obtain

$$\begin{aligned} E_Q\big[\mathbf{1}_G\, f\big(P[A_1 \,|\, \mathcal{G}]\big)\big] &= E_P\big[h\, \mathbf{1}_G\, f\big(P[A_1 \,|\, \mathcal{G}]\big)\big] = E_P\big[h\, \mathbf{1}_G\, f\big(P[A_1 \,|\, \mathcal{H}]\big)\big] \\ &= E_Q\big[\mathbf{1}_G\, f\big(P[A_1 \,|\, \mathcal{H}]\big)\big] = E_Q\big[\mathbf{1}_G\, Q[A_1 \,|\, \mathcal{H}]\big] = Q[A_1 \cap G]. \end{aligned}$$

This implies the assertion. □

Based on Lemma 1 and Proposition 1, we are in a position to prove the main result of this paper. It states that an information subset inherits the covariate shift property from its information superset for all absolutely continuous target distributions if and only if the subset is statistically sufficient for the superset with respect to the positive class label under the source distribution.

Theorem 1. *Under Assumption 1, let $\mathcal{G} \subset \mathcal{H}$ be another sub-σ-algebra of $\mathcal{A}$. Then $\mathcal{G}$ is sufficient for $\mathcal{H}$ with respect to A_1 under the source distribution P if and only if $C^*_A(P, \mathcal{H}) \subset C^*_A(P, \mathcal{G})$ holds true.*

Proof. The 'only if' part of the assertion is implied by Proposition 1. By the definition of conditional probability, for the 'if' part we have to show that for each $H \in \mathcal{H}$ it holds that $P[A_1 \cap H] = E_P\big[\mathbf{1}_H\, P[A_1 \,|\, \mathcal{G}]\big]$. This is obvious for H with $P[H] = 0$. Hence fix an event $H \in \mathcal{H}$ and assume $P[H] > 0$.

Define the probability measure Q_H on $(\Omega, \mathcal{A})$ as P conditional on H, i.e.

$$Q_H[M] \;=\; P[M \,|\, H] \;=\; \frac{P[M \cap H]}{P[H]}, \quad \text{for all } M \in \mathcal{A}.$$

This Q_H is absolutely continuous with respect to P on $\mathcal{A} \supset \mathcal{H}_{\mathcal{A}}$, with $\mathcal{H}$-measurable density $\frac{\mathbf{1}_H}{P[H]}$. Hence, by Lemma 1 we obtain $Q_H \in C^*_A(P, \mathcal{H})$. By assumption, this implies $Q_H \in C^*_A(P, \mathcal{G})$, and in particular $P[A_1 \,|\, \mathcal{G}] = Q_H[A_1 \,|\, \mathcal{G}]$. From this, it follows that

$$\begin{aligned} E_P\big[\mathbf{1}_H\, P[A_1 \,|\, \mathcal{G}]\big] &= P[H]\, E_{Q_H}\big[P[A_1 \,|\, \mathcal{G}]\big] \\ &= P[H]\, E_{Q_H}\big[Q_H[A_1 \,|\, \mathcal{G}]\big] = P[H]\, Q_H[A_1] = P[A_1 \cap H]. \end{aligned}$$

This completes the proof. □

5 Discussion of Theorem 1

Can sufficiency of $\mathcal{G}$ for $\mathcal{H}$ with respect to A_1 be characterised in other ways than just requiring $P[A_1 \,|\, \mathcal{G}] = P[A_1 \,|\, \mathcal{H}]$?

- As observed by Devroye et al. [7] (Section 32), if $\mathcal{G} = \sigma(T)$ is generated by some random variable T, then $\mathcal{G}$ is sufficient for $\mathcal{H}$ if and only if there exists a measurable function g such that $P[A_1 \,|\, \mathcal{H}] = g(T)$.

– Primary examples for such T are transformations $T = f(P[A_1 \,|\, \mathcal{H}])$ of the posterior class probability which may emerge as scoring classifiers optimising the area under the Receiver Operating Characteristic (ROC) or the area under the Brier curve (Tasche [26], Section 5.3). The process to reengineer $P[A_1 \,|\, \mathcal{H}]$ from T is called 'calibration' (see Kull et al. [18] and the references therein).

Johansson et al. [16] wrote in Section 4.1: "One interpretation ... is that covariate shift ([their] Assumption 1) need not hold with respect to the representation $Z = \phi(X)$, even if it does with respect to X. With $\phi^{-1}(z) = \{x : \phi(x) = z\}$,

$$p_t(Y \,|\, z) \;=\; \frac{\int_{x \in \phi^{-1}(z)} p_t(Y \,|\, x)\, p_t(x)\, dx}{\int_{x \in \phi^{-1}(z)} p_t(x)\, dx} \;\neq\; p_s(Y \,|\, z). \tag{8}$$

Equality holds for general p_s, p_t only if ϕ is invertible." According to Section 2 of Johansson et al., p_s and p_t stand for the densities of the covariate X on the 'source domain' and 'target domain' respectively. By Theorem 1, with $\mathcal{G} = \sigma(Z)$, actually covariate shift holds under the transformation ϕ if $\mathcal{G}$ is sufficient for $\mathcal{H} = \sigma(X)$ (in the setting of Johansson et al.). Sufficiency of $\mathcal{G}$ is implied by invertibility of ϕ. Hence, Theorem 1 is a more general statement than the one by Johansson et al. [16].[4]

Under Assumption 1, a mapping (representation) $T : (\Omega, \mathcal{H}) \to (\Omega_T, \mathcal{H}_T)$ which is $\mathcal{H}_T$-$\mathcal{H}$-measurable is said to have 'invariant components' (Gong et al. [10]) if its distributions under the source and target distributions are the same, i.e. if

$$P[T \in M] \;=\; Q[T \in M], \qquad \text{for all } M \in \mathcal{H}_T. \tag{9}$$

As $\mathcal{H}$ reflects the covariates, T can be interpreted as a transformation of the covariates that makes their distributions undistinguishable under the source and target distributions. As Gong et al. [10] noted, (9) alone does not imply that the posterior probabilities under source and target distributions are the same or at least similar. He et al. [12] therefore defined the notion of 'domain invariance' by

$$P[A_1 \,|\, \sigma(T)] = P[A_1 \,|\, \mathcal{H}], \quad Q[A_1 \,|\, \sigma(T)] = Q[A_1 \,|\, \mathcal{H}], \quad \text{and} \tag{10a}$$

$$P[A_i \cap \{T \in M\}] = Q[A_i \cap \{T \in M\}], \quad i \in \{0, 1\}, M \in \mathcal{H}_T. \tag{10b}$$

He et al. [12] then observed that (10a) and (10b) together imply covariate shift with respect to the information set $\mathcal{H}$, i.e. $P[A_1 \,|\, \mathcal{H}] = Q[A_1 \,|\, \mathcal{H}]$.[5] In a sense, the observation by He et al. can be considered complementary to Theorem 1 because

[4] The derivation of (8) in [16] is somewhat sloppy. In Section 2.3 of [16], the assumption is made for $Z = \phi(X)$ that '$p(Z)$' is a density. This implies $\int_{x \in \phi^{-1}(z)} p_t(x)\, dx = 0$ which means that the denominator of the fraction in (8) is zero.

[5] Actually, (10b) implies covariate shift with respect to $\sigma(T)$. From this, together with (10a), follows covariate shift with respect to $\mathcal{H}$. Hence the assumption of (10b) could be replaced by the weaker assumption of having covariate shift with respect to $\sigma(T)$.

Theorem 1 is about passing on covariate shift from a larger information set to a smaller one while the observation by He et al. is a statement about covariate shift on a smaller information set implying covariate shift on a larger one.

In unsupervised domain adaptation, the case of source and target distributions where part or all of the support of the target distribution is not covered by the support of the source distribution is of great interest [3,16]. In that case, the target distribution is at least partially singular to the source distribution. Has Theorem 1 any relevance for this situation? Arguably, representations of the covariates which do not work even in the plain-vanilla environment of target distributions which are absolutely continuous with respect to the source distribution, are rather questionable. Hence Theorem 1 may be considered useful for providing a kind of 'fatal flaw' test for representations.

There are situations when covariate shift for a given sub-σ-algebra $\mathcal{G}$ can be forced. The most important example of such a situation is sample selection (Hein [13], 'Class-Conditional Independent Selection'). Theorem 1 may not be relevant then.

However, if the rationale for the assumption of covariate shift is based on causality considerations (like e.g. in Storkey [25]), the set of covariates associated to the information set $\mathcal{H}$ in the definition of covariate shift might turn out to be quite large, rendering tedious the task of estimating the posterior $P[A_1 \mid \mathcal{H}]$. Theorem 1 provides the condition under which the size (or dimension) of the set of covariates may be reduced without destroying the invariance of the posterior class probabilities between the source and arbitrary target distributions. This condition does not require any special properties of the target distributions Q but the harmless requirement of being absolutely continuous with respect to the source distribution P. Note however that Theorem 1 leaves open the possibility that the covariate shift property is inherited by a non-sufficient sub-σ-algebra for some (but not all) specific target distributions.

If the set of covariates generating $\mathcal{H}$ contains at least one real-valued covariate which has a Lebesgue-density and is not independent of A_1, then there is no sufficient four-elements sub-σ-algebra $\mathcal{G}$ such that (4) holds. For sufficieny would imply that the range of the posterior class probability $P[A_1 \mid \mathcal{H}]$ consists of two values only – which is wrong for probabilities conditional on continuous random variables. Hence by Theorem 1 no radically simple approach to class prior estimation like (3) that would be applicable under all possible shifts of the covariate distribution is available in this case.

6 Probing for class prior estimation under covariate shift

To the author's best knowledge, there is basically one approach to class prior estimation on the target dataset under covariate shift: Estimate the posterior probability of the positive class as a function of the covariates on the source dataset and then calculate its average on the target dataset, see (2). Card and Smith [6] discuss two variants of this approach, one of them with and the other

without proper calibration of the posterior probabilities – hence the concept in principle is the same in both variants.

Under prior probability shift, the simple 'confusion matrix method' can be deployed to achieve consistent class prior estimates [9,23]. As seen in Sections 3 and 4, no similarly simple approach based on merely making use of one classifier's output works under covariate shift. However, averaging the counting results of a large ensemble of classifiers trained for a variety of cost-sensitive classification problems would work ('probing': Langford and Zadrozny [19]; Tasche [26]).

Sketch of class prior estimation with probing. Define the cost-sensitive (weighted) classification loss (with $0 \leq t \leq 1$) in the setting of Assumption 1:

$$L(H,t) \;=\; (1-t)\,P[A_1 \cap (\Omega \setminus H)] + t\,P[A_0 \cap H], \quad H \in \mathcal{H}.$$

The probing algorithm adapted to class prior estimation then can be described as follows:

1) Choose an appropriately 'dense' set $0 = t_0 < t_1 < t_2 < \ldots < t_n < 1$.
2) For each t_i, $i = 1, \ldots, n$, find – with possibly different approaches – a nearly optimal minimising classifier[6] $H(t_i)$ of $L(H, t_i)$, $H \in \mathcal{H}$.
3) Let $Z = \sum_{i=1}^{n} (t_i - t_{i-1})\,\mathbf{1}_{H(t_i)}$.
4) For all j with $L(\{Z > t_j\}, t_j) < L(H(t_j), t_j)$, replace $H(t_j)$ with $\{Z > t_j\}$.
5) Repeat steps 3) and 4) until $L(\{Z > t_j\}, t_j) \geq L(H(t_j), t_j)$ for all j.
6) Calculate $\widehat{q} = \sum_{i=1}^{n} (t_i - t_{i-1})\,Q[H(t_i)]$ as estimate of the positive class prior probability $Q[A_1]$ under the target distribution.

7 Conclusions

We have shown that covariate shift is a fragile notion, in the sense that the invariance of the posterior class probabilities between source and target distributions may be lost if the set of covariates on which the posterior probabilities are conditioned is diminished. This observation implies that under covariate shift simple estimators of the target prior class probabilities are infeasible if they are designed in the style of the confusion matrix method (adjusted count) which is a popular quantifier under prior probability shift.

Valid methods for class prior estimation under covariate shift are the careful estimation of the posterior class probabilities conditioned on the full set or a sufficient subset of the covariates, combined with subsequently averaging them on the target dataset (probabilistic classify & count). The application of probing as described in Section 6 could also prove useful for class prior estimation under covariate shift. So far, probing for class prior estimation has not yet been thoroughly tested. This could be a subject for future research.

[6] As before, we identify a set with its indicator function that gives the value 1 on the set and the value 0 on its complement.

Acknowledgements. The author is grateful to Juan José del Coz and Pablo González for drawing his attention to the subject of class prior estimation under covariate shift and to four anonymous reviewers whose comments redounded to significant improvements of the paper.

References

1. Adragni, K., Cook, R.: Sufficient dimension reduction and prediction in regression. Philosophical Transactions of the Royal Society A **367**, 4385–4405 (2009)
2. Bella, A., Ferri, C., Hernandez-Orallo, J., Ramírez-Quintana, M.: Quantification via probability estimators. In: Data Mining (ICDM), 2010 IEEE 10th International Conference on. pp. 737–742. IEEE (2010)
3. Ben-David, S., Blitzer, J., Crammer, K., Pereira, F.: Analysis of Representations for Domain Adaptation. In: Schölkopf, B., Platt, J., Hoffman, T. (eds.) Advances in Neural Information Processing Systems. vol. 19, pp. 137–144. MIT Press (2006)
4. Billingsley, P.: Probability and measure. John Wiley & Sons, second edn. (1986)
5. Caelen, O.: Quantification and learning algorithms to manage prior probability shift. Master thesis, Institut de Statistique, Biostatistique et Sciences Actuarielles, Université catholique de Louvain (2017)
6. Card, D., Smith, N.: The Importance of Calibration for Estimating Proportions from Annotations. In: Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers). pp. 1636–1646 (2018)
7. Devroye, L., Györfi, L., Lugosi, G.: A Probabilistic Theory of Pattern Recognition. Springer (1996)
8. Forman, G.: Counting Positives Accurately Despite Inaccurate Classification. In: European Conference on Machine Learning (ECML 2005). pp. 564–575. Springer (2005)
9. Gart, J., Buck, A.: Comparison of a screening test and a reference test in epidemiologic studies. II. A probabilistic model for the comparison of diagnostic tests. American Journal of Epidemiology **83**(3), 593–602 (1966)
10. Gong, M., Zhang, K., Liu, T., Tao, D., Glymour, C., Schölkopf, B.: Domain Adaptation with Conditional Transferable Components. In: Balcan, M., Weinberger, K. (eds.) Proceedings of The 33rd International Conference on Machine Learning. Proceedings of Machine Learning Research, vol. 48, pp. 2839–2848. PMLR, New York, New York, USA (20–22 Jun 2016)
11. González, P., Castaño, A., Chawla, N., Coz, J.D.: A Review on Quantification Learning. ACM Comput. Surv. **50**(5), 74:1–74:40 (2017)
12. He, H., Yang, Y., Wang, H.: Domain Adaptation with Factorizable Joint Shift. arXiv preprint arXiv:2203.02902 (2022)
13. Hein, M.: Binary Classification under Sample Selection Bias. In: Quiñonero-Candela, J., Sugiyama, M., Schwaighofer, A., Lawrence, N. (eds.) Dataset Shift in Machine Learning, chap. 3, pp. 41–64. The MIT Press, Cambridge, Massachusetts (2009)
14. Hofer, V., Krempl, G.: Drift mining in data: A framework for addressing drift in classification. Computational Statistics & Data Analysis **57**(1), 377–391 (2013)
15. Holzmann, H., Eulert, M.: The role of the information set for forecasting – with applications to risk management. The Annals of Applied Statistics **8**(1), 595–621 (2014)

16. Johansson, F., Sontag, D., Ranganath, R.: Support and Invertibility in Domain-Invariant Representations. In: Chaudhuri, K., Sugiyama, M. (eds.) Proceedings of the Twenty-Second International Conference on Artificial Intelligence and Statistics. Proceedings of Machine Learning Research, vol. 89, pp. 527–536. PMLR (16–18 Apr 2019)
17. Klenke, A.: Probability Theory: A Comprehensive Course. Springer Science & Business Media (2013)
18. Kull, M., Silva Filho, T., Flach, P.: Beyond sigmoids: How to obtain well-calibrated probabilities from binary classifiers with beta calibration. Electron. J. Statist. **11**(2), 5052–5080 (2017)
19. Langford, J., Zadrozny, B.: Estimating Class Membership Probabilities using Classifier Learners. In: Cowell, R., Ghahramani, Z. (eds.) AISTATS 2005 – Proceedings of the Tenth International Workshop on Artificial Intelligence and Statistics. pp. 198–205. The Society for Artificial Intelligence and Statistics (2005)
20. Lipton, Z., Wang, Y.X., Smola, A.: Detecting and Correcting for Label Shift with Black Box Predictors. In: Dy, J., Krause, A. (eds.) Proceedings of the 35th International Conference on Machine Learning. Proceedings of Machine Learning Research, vol. 80, pp. 3122–3130. PMLR (10–15 Jul 2018)
21. Moreno-Torres, J., Raeder, T., Alaiz-Rodriguez, R., Chawla, N., Herrera, F.: A unifying view on dataset shift in classification. Pattern Recognition **45**(1), 521–530 (2012)
22. Quiñonero-Candela, J., Sugiyama, M., Schwaighofer, A., Lawrence, N. (eds.): Dataset Shift in Machine Learning. MIT Press (2008)
23. Saerens, M., Latinne, P., Decaestecker, C.: Adjusting the Outputs of a Classifier to New a Priori Probabilities: A Simple Procedure. Neural Computation **14**(1), 21–41 (2001)
24. Scott, C.: A Generalized Neyman-Pearson Criterion for Optimal Domain Adaptation. In: Proceedings of Machine Learning Research, 30th International Conference on Algorithmic Learning Theory. vol. 98, pp. 1–24 (2019)
25. Storkey, A.: When Training and Test Sets Are Different: Characterizing Learning Transfer. In: Quiñonero-Candela, J., Sugiyama, M., Schwaighofer, A., Lawrence, N. (eds.) Dataset Shift in Machine Learning, chap. 1, pp. 3–28. The MIT Press, Cambridge, Massachusetts (2009)
26. Tasche, D.: Calibrating sufficiently. Statistics **55**(6), 1356–1386 (2021)
27. Tasche, D.: Minimising quantifier variance under prior probability shift. In: Cong, G., Ramanath, M. (eds.) Proceedings of the CIKM 2021 Workshops (2021), first International Workshop on Learning to Quantify: Methods and Applications (LQ 2021)

Semi-Automated Estimation of Weighted Rates for E-commerce Catalog Quality Monitoring

Mauricio Sadinle, Karim Bouyarmane, Grant Galloway, Shioulin Sam, Changhe Yuan, and Ismail Tutar

Amazon.com, Inc.
{sadinlem,bouykari,gggallow,shioulin,ychanghe,ismailt}@amazon.com

Abstract. Product catalogs represent the backbone of e-commerce websites. Given these catalogs' constant evolution, we need to closely monitor the quality of their product information. Identifying defective product information, however, often requires human auditing, which makes catalog monitoring expensive. In this article, we investigate approaches for tracking weighted rates over time, here defined as the fraction of customer attention that goes to items with a particular defect. We focus on these metrics, given that to improve customer trust we need to minimize their exposure to listings with defective information. We assume that the gold standard for detecting defects comes from human auditors, but to avoid collecting audits at each point in time, we leverage existing machine learning classifiers. However, simply replacing human auditor decisions with automated predictions generally leads to large biases in the estimated weighted rates. We instead leverage classifiers while obtaining approximately unbiased and low variance estimators of the weighted rate of interest. We rely on being able to evaluate the quality of the classifier using audits at a baseline time, and then extrapolate its performance to the target times. We perform extensive simulation studies to stress-test our proposed estimation approaches under a variety of scenarios representative of our use cases. Our proposed estimation approach is related to the task of *quantification* in machine learning, and so we draw connections throughout the document.

Keywords: Quantification · Prior probability shift · Label shift.

1 Introduction

Product catalogs are the backbone of e-commerce websites, as they provide the information that is presented to customers. Maintaining customer trust requires identifying defects in product information, which usually needs human inspection for detection. For instance, product information on Amazon.com is consolidated from contributions by individual sellers [1]. These consolidated product attributes frequently contain defects, such as inconsistencies or erroneous values due to honest mistakes by sellers, system errors, and bad actors who intentionally introduce corrupted information. This causes detrimental performance of a

variety of customer-facing applications; for instance, displaying such imperfect information to customers erodes their trust.

Given the scale of e-commerce product catalogs, it is nearly impossible to manually inspect all of their information. An important task in ensuring high catalog quality involves monitoring quality metrics. This monitoring is often done through careful human inspection of random samples of product entries collected periodically. Even with carefully designed samples, when business goals require tight control of these metrics at high frequency, monitoring through human auditing becomes extremely expensive. This creates the need for automated procedures that allow to monitor quality metrics while maintaining strong guarantees on their accuracy.

In this article, we investigate a methodology for estimating weighted rates in a semi-automated way. The reason for using weighted rates in our use cases is that not all products are equally important for customers. Given a signal of customer engagement with each product, we are interested in monitoring the fraction of such signal among products with a defect. For instance, we might want to track the fraction of customer visits to product pages that contain erroneous information; from a customer-centric point of view, this is a more important metric to monitor and aim to reduce than the simple fraction of catalog products with erroneous information.

For our use cases there typically exist machine learning classifiers in production for detecting defects. These classifiers tend to be complex and are trained on audited data collected over time. Given that retraining such classifiers for the sake of metric measurement is burdensome and not cost-efficient, we propose to use them in their existing form to predict defects. However, it is well known that simply replacing human auditor decisions with classifier predictions generally leads to large biases in the estimated metrics [9, 12, 6, 7]. To leverage existing classifiers while obtaining approximately unbiased and low variance estimators, we rely on being able to evaluate the quality of the classifier using audits at a baseline time. We then assume that the performance of the classifier in terms of its true and false positive rates can be extrapolated to the target time. Our methodology constitutes an extension of techniques proposed for the machine learning task of *quantification*, reviewed next.

1.1 Quantification

Forman [7] introduced the *quantification* task to address the following problem: how can we use labeled training data from a baseline population to estimate the proportion of a class in a target population where we only have unlabeled data. This task is related to the fundamental problem of estimating a proportion using an imperfect diagnostic tool, studied earlier in epidemiology [9, 12], in the context of mechanical sorting devices [11], among others [10]. A seemingly obvious solution to the quantification task is to train a classifier on the labeled data, use it to predict the class for the unlabeled data, and then simply summarize the proportion of class predictions. This approach, known as *classify and count* [6, 7], is known to perform poorly, as it is generally guaranteed to be biased, except

for a few restrictive conditions [8, 10]. Forman [6, 7] recognized this, and proposed alternative approaches for estimation, including an *adjusted classify and count* (ACC) approach that is guaranteed to work well under certain conditions; we will refer back to this method later in this article. Interestingly, the ACC approach had also been derived earlier by other authors [9, 12, 11], which shows the ubiquity of the quantification problem.

Forman [7, 8] also introduced *cost-quantification*, as the task of estimating total costs for each class using class predictions by imperfect classifiers. This task seems to have received less attention in the literature; for instance, a 2017 review [10] only included the proposed solutions by Forman [7] in 2006, and to the best of our knowledge no further advances have been proposed for cost-quantification since then, despite many advances for the simpler quantification task [13–15]. The automated estimation of weighted rates, as in this paper, is closely related to cost-quantification, given that if we can estimate the total of a class, we can also estimate the fraction it represents with respect to the total cost across classes. The methods that we propose in this article therefore also contribute to cost-quantification solutions. Our contribution consists in showing that an analog of the ACC approach is valid for estimating weighted rates under two assumptions that allow to extrapolate and simplify the true and false positive rates of the classifier. We also investigate approaches for dealing with the classifiers' thresholds that lead to weighted rate estimators with low variance, and compare approaches for constructing confidence intervals.

2 Methodology

We shall think of a product catalog at a time t as a collection of features on N_t products. A product i has a *known* non-negative measure of importance for customers, or weight, at time t denoted W_{it}. Let Y_{it} denote the defect indicator for product i at time t, 1 if defective, 0 otherwise. This indicator Y_{it} is unknown and determining its true value requires human auditing.

Estimation target: Formally, our goal is to estimate the *weighted rate* R_t at a time t:

$$R_t = \sum_{i=1}^{N_t} W_{it} Y_{it} \Big/ \sum_{i=1}^{N_t} W_{it}. \tag{1}$$

We assume that we do not have audited data from the catalog at time t, and so we rely on the existence of a classifier to predict the status Y_{it} of a product i. Let $h(\cdot)$ denote a generic classifier that takes in a feature vector X_{it} of product i at time t, and outputs a predicted status $\hat{Y}_{it}$ for product i, that is, $\hat{Y}_{it} = h(X_{it}) \in \{0, 1\}$. The classifier $h(\cdot)$, for instance, can be obtained from thresholding a score at a cutpoint c, say $h(X_{it}) = I[g(X_{it}) > c]$, where $I(\cdot)$ is the indicator function and $g(\cdot)$ may represent a score obtained from a model or from some complicated procedure. For now we assume that $h(\cdot)$ is fixed, but later we compare approaches to handle classification thresholds.

We also rely on having an audited sample at a baseline time, which we use to estimate the true and false positive rates of the classifier at that baseline time. In

practice, we implement *measurement cycles* that start with collection of audits to evaluate the performance of the classifier, and then use that information to produce automated estimates for the remainder of the cycle; see Appendix A at `https://bit.ly/3wJK5Mj` for a more detailed description.

2.1 The Proposed Weighted Rate Estimator

To derive the proposed estimator of the weighted rate, we first do a slight rewriting of the estimation target. To this end, let $(W, Y, \hat{Y})$ be a random vector that takes with probability $1/N_t$ each of the catalog values at time t, $\{(W_{it}, Y_{it}, \hat{Y}_{it})\}_{i=1}^{N_t}$. With this formulation, our estimation target can be equivalently written as

$$R_t = \frac{\sum_{i=1}^{N_t} W_{it} Y_{it}}{\sum_{i=1}^{N_t} W_{it}} = \frac{(1/N_t)\sum_{i=1}^{N_t} W_{it} Y_{it}}{(1/N_t)\sum_{i=1}^{N_t} W_{it}} = \frac{E_t(WY)}{E_t(W)},$$

where $E_t(\cdot)$ denotes the expected value using the values of the catalog at time t.

The quantity that we would obtain from simply using the predictions $\hat{Y}_{it}$ instead of the true values Y_{it} is here denoted as R_t^{raw}, and it is given by

$$R_t^{raw} = \frac{\sum_{i=1}^{N_t} W_{it} \hat{Y}_{it}}{\sum_{i=1}^{N_t} W_{it}} = \frac{E_t(W\hat{Y})}{E_t(W)},$$

which generally will differ from the target R_t. Our strategy to derive the proposed weighted rate estimator requires connecting R_t and R_t^{raw} through the classification performance of $h(\cdot)$. First, note that we assume the weights W_{it} to be known, and therefore $E_t(W)$ to be known, allowing us to focus on connecting $E_t(WY)$ with $E_t(W\hat{Y})$. Note that, by the law of total expectation, we can write

$$\begin{aligned} E_t(WY) &= E_t[W\ P_t(Y = 1 \mid W)], \\ E_t(W\hat{Y}) &= E_t[W\ P_t(\hat{Y} = 1 \mid W)]. \end{aligned} \tag{2}$$

Also, by the law of total probability,

$$P_t(\hat{Y} = 1 \mid W) = p_{1|1,t}(W)\ P_t(Y = 1 \mid W) + p_{1|0,t}(W)\ [1 - P_t(Y = 1 \mid W)], \tag{3}$$

where $p_{1|a,t}(W) = P_t(\hat{Y} = 1 \mid Y = a, W)$ denotes the true positive rate (TPR) for $a = 1$, and the false positive rate (FPR) for $a = 0$, as a function of the weights at time t. From equation (3), we can establish the relationship

$$P_t(Y = 1 \mid W) = \frac{P_t(\hat{Y} = 1 \mid W) - p_{1|0,t}(W)}{p_{1|1,t}(W) - p_{1|0,t}(W)}, \tag{4}$$

which resembles the basis for the ACC estimator of simple proportions [6], although here it appears conditional on a value W of the weights. Replacing equation (4) into (2) above, we obtain the identity

$$E_t(WY) = E_t\left[W\ \frac{P_t(\hat{Y} = 1 \mid W) - p_{1|0,t}(W)}{p_{1|1,t}(W) - p_{1|0,t}(W)}\right]. \tag{5}$$

Creating an estimator based on this expression is not straightforward. Firstly, estimating the TPR and FPR functions, $p_{1|1,t}(W)$ and $p_{1|0,t}(W)$, for the catalog at time t would require collecting audited data at time t, which defeats the purpose of automating the estimation approach. The validity of our proposed estimator therefore relies on being able to extrapolate the performance of the classifier from a baseline time to the target time t.

Extrapolation assumption (EA): The TPR and FPR at the time of interest t are the same as at the baseline time.

Additionally, although not strictly required, we also work under an extra assumption to favor a simple estimator.

Simplifying assumption (SA): The TPR and FPR are constant as a function of the weights.

We discuss the plausibility of these assumptions in detail in Section 2.2. The EA can be written as $P_0(\hat{Y} = 1 \mid Y = a, W) = P_t(\hat{Y} = 1 \mid Y = a, W)$, for $a = 0, 1$. Under the EA, we can ignore the time subindex and simply write $p_{1|a}(W) = P(\hat{Y} = 1 \mid Y = a, W)$, for $a = 0, 1$. Then, the SA can be written as $p_{1|a}(W) = p_{1|a}(W')$ for any two values of the weights W and W', where $a = 0, 1$. Under the SA we can simplify the notation and write $p_{1|1} = p_{1|1}(W)$ and $p_{1|0} = p_{1|0}(W)$.

Given the EA and SA, expression (5) simplifies as

$$E_t(WY) = E_t\left[W\ \frac{P_t(\hat{Y} = 1 \mid W) - p_{1|0}}{p_{1|1} - p_{1|0}}\right] = \frac{E_t(W\hat{Y}) - p_{1|0}E_t(W)}{p_{1|1} - p_{1|0}},$$

and we obtain

$$R_t = \frac{E_t(WY)}{E_t(W)} = \frac{E_t(W\hat{Y})/E_t(W) - p_{1|0}}{p_{1|1} - p_{1|0}} = \frac{R_t^{raw} - p_{1|0}}{p_{1|1} - p_{1|0}}. \tag{6}$$

Interestingly, this has the same form as the ACC estimator for simple proportions [9, 12, 6, 7], except that here R_t and R_t^{raw} are weighted rates.

Given expression (6), we propose to estimate the weighted rate as

$$\hat{R}_t = \frac{\hat{R}_t^{raw} - \hat{p}_{1|0}}{\hat{p}_{1|1} - \hat{p}_{1|0}}, \tag{7}$$

where $\hat{R}_t^{raw}$ is estimated from a very large random sample from the catalog at time t, or preferably $\hat{R}_t^{raw}$ is taken exactly as R_t^{raw}, if computational resources allow. The estimated TPR and FPR, $\hat{p}_{1|1}$ and $\hat{p}_{1|0}$, are obtained from the audited data from baseline. The appropriate estimators for each of these quantities depend on the sampling scheme [16], but as long as they are consistent, the consistency of $\hat{R}_t$ is guaranteed by the continuous mapping theorem [19] because the true value $R_t = (R_t^{raw} - p_{1|0})/(p_{1|1} - p_{1|0})$ is a continuous function of R_t^{raw}, $p_{1|1}$, and $p_{1|0}$. This argument serves as the proof of the following result.

Theorem 1 (statistical consistency). *Under EA and SA, assume that $\hat{R}_t^{raw}$, $\hat{p}_{1|1}$, and $\hat{p}_{1|0}$ are statistically consistent estimators for R_t^{raw}, the TPR, and the*

FPR, respectively. Then, the proposed estimator $\hat{R}_t$ is statistically consistent for the target rate R_t.

Statistical consistency of our estimator is an important property, as it guarantees that as the sample sizes increase, the estimator converges in probability to the true value that we want to estimate [19]. In particular, it implies that our estimator is approximately unbiased for large sample sizes. Working with statistically consistent estimators $\hat{R}_t^{raw}$, $\hat{p}_{1|1}$, and $\hat{p}_{1|0}$ is relatively standard; for instance, with simple random samples $\mathcal{S}_0$ of size n_0 at baseline, and $\mathcal{S}_t$ of size $n_t \gg n_0$ at time t, the following estimators are consistent:

$$\hat{R}_t^{raw} = \sum_{i \in \mathcal{S}_t} W_{it} \hat{Y}_{it} \Big/ \sum_{i \in \mathcal{S}_t} W_{it}; \quad \hat{p}_{1|a} = \sum_{i \in \mathcal{S}_0} \hat{Y}_{i0} I(Y_{i0} = a) \Big/ \sum_{i \in \mathcal{S}_0} I(Y_{i0} = a),\ a = 0, 1.$$

More intricate estimators will be needed under more complex sampling schemes, but those details go beyond the scope of this paper. The proposed estimator $\hat{R}_t$ heavily relies on the assumptions EA and SA, which we discuss next.

2.2 Discussion of Assumptions

To examine the plausibility of the assumptions EA and SA, let us expand the TPR and FPR in terms of the classifier $h(\cdot)$ and the product's features X,

$$P_t(\hat{Y} = 1 \mid Y = a, W) = \int P_t(\hat{Y} = 1 \mid x, Y = a, W) f_t(x \mid Y = a, W) dx,$$

where $P_t(\hat{Y} = 1 \mid x, Y = a, W) = I[h(x) = 1]$ since the automated procedure $h(\cdot)$ only uses the features X as input, and $f_t(x \mid Y = a, W)$ represents the distribution of the features X at time t among products with $Y = a$ and weight W. We can see that $P_t(\hat{Y} = 1 \mid Y = a, W)$ might depend on the time t and the product weight W only if the distribution of the features X changes from time 0 to t and for different values of the product's weight W among the two groups of products with and without the characteristic of interest. This leads to sufficient conditions for the assumptions above.

Sufficient condition for extrapolation assumption: The distributions of the features X among products with and without the characteristic of interest, and for the different values of importance, are the same at time 0 and at time t, that is, $f_t(x \mid Y = a, W) = f_0(x \mid Y = a, W)$.

This is a conditional version of what is sometimes referred to as the *prior probability shift* assumption [5, 18]. To examine this sufficient condition, let us say that $Y = 1$ indicates that a product contains a defect in a specific attribute. In such case, this condition says that the distribution of the features used to predict defects, among products that are defective $Y = 1$ and that have a specific importance W, is the same at baseline and at time t. In other words, we expect to see the same indications of defects at baseline and at time t among defective products that have the same importance. A similar interpretation would apply among non-defective products.

Sufficient condition for simplifying assumption: The distributions of the features X among products with and without the characteristic of interest Y are the same regardless of the importance of the products, that is, $f(x \mid Y = a, W) = f(x \mid Y = a)$.

Continuing with the example of defects, this condition says that the distribution of the features used to predict defects among defective products is the same regardless of how popular the product is. Namely, we expect to see the same indicators of defects among defective products, regardless of how important they are. A similar interpretation would apply among non-defective products.

The EA is a fundamental assumption that allows us to borrow information from the audited sample at baseline to obtain an estimate for follow-up times. We need this assumption to extrapolate the performance of the classifier $h(\cdot)$. On the other hand, the SA is not strictly necessary, as in principle we can use the audited data at baseline to build models of the probabilities $P_t(\hat{Y} = 1 \mid Y = a, W)$ and obtain a more flexible estimator; we discuss this further in Section 4. Nevertheless, the SA allows us to obtain an initial simple estimator on which we can build and improve upon.

2.3 Dealing with Classifier Thresholds

The proposed estimator (7) of the weighted rate was derived assuming that the classifier $h(\cdot)$ is fixed, however, the classifier might be obtained as $h(x) = I[g(x) > c]$, that is, it depends on thresholding a score $g(x)$. We study two approaches for handling the cutpoint c, although we assume that the score function $g(x)$ is fixed, as in our use cases where it is already trained at the baseline time.

Variance Minimization Given a threshold c, we can use the classifier $h(x) = I[g(x) > c]$ to obtain an estimate $\hat{R}_t = (\hat{R}_t^{raw} - \hat{p}_{1|0})/(\hat{p}_{1|1} - \hat{p}_{1|0})$, where each of $\hat{R}_t^{raw}$, $\hat{p}_{1|1}$ and $\hat{p}_{1|0}$ are implicitly functions of the threshold c. Given the classifier $h(x)$, we can obtain an analytical approximation of the variance of $\hat{R}_t$, as shown in Appendix B at `https://bit.ly/3wJK5Mj`. We denote the estimated variance given threshold c as V_c. The variance minimization approach simply takes a grid of u threshold values, $c_1, \ldots, c_u$, computes the estimated variance given each threshold, $V_1, \ldots, V_u$, and selects the threshold c^* that minimizes the estimated variance. The final weighted rate estimator is computed from the classifier $h(x) = I[g(x) > c^*]$.

Variance minimization has been implemented for quantification before, for instance [17] used it within a mixture model approach to quantification.

Median Sweep Forman [7] studied different strategies for choosing classification thresholds to obtain reliable estimation of the prevalence of a class, and found that the approach known as *median sweep* was the best in terms of leading to the lowest bias. These results were replicated recently [15], and therefore we implement median sweep along with our proposed weighted rate estimator.

Median sweep consists in computing the estimates $\hat{R}_{t,1}, \ldots, \hat{R}_{t,u}$ according to the threshold values in a grid $c_1, \ldots, c_u$, and returning the median estimate. This approach is theoretically justified, given that each of the estimators $\hat{R}_{t,1}, \ldots, \hat{R}_{t,u}$ corresponding to a fixed grid $c_1, \ldots, c_u$ is guaranteed to be statistically consistent, as shown in Theorem 1, and thereby asymptotically unbiased, as long as $\hat{R}_t^{raw}$, $\hat{p}_{1|0}$ and $\hat{p}_{1|1}$ are estimated in a statistically consistent way. Under such reasonable conditions, the median of these individual estimators inherits the statistical consistency and asymptotic unbiasedness.

We also explore the performance of a *trimmed median sweep* approach by Forman [7], who proposed to use median sweep after discarding estimates from thresholds that lead to $|\hat{p}_{1|1} - \hat{p}_{1|0}| < 0.25$, in order to provide more stability to the ACC estimator.

2.4 Confidence Intervals

We also propose approaches to build confidence intervals, using analytical methods and the bootstrap [4].

Analytic Confidence Interval Given an estimator $\hat{R}_t = (\hat{R}_t^{raw} - \hat{p}_{1|0})/(\hat{p}_{1|1} - \hat{p}_{1|0})$ obtained from a specific classifier $h(\cdot)$, for instance obtained from a specific threshold, we can use the analytical variance formula derived in Appendix B (at `https://bit.ly/3wJK5Mj`) to obtain an estimate of the variance $\widehat{\text{var}}(\hat{R}_t)$, and form a confidence interval based on the asymptotic normality of $\hat{R}_t$. A $100(1-\alpha)\%$ confidence interval, with $\alpha \in (0, 1)$, is given by $\hat{R}_t \pm z_{1-\alpha/2}\sqrt{\widehat{\text{var}}(\hat{R}_t)}$, where $z_{1-\alpha/2}$ is the $1 - \alpha/2$ quantile of a standard normal distribution, say $z_{0.975} = 1.96$ for a 95% confidence interval. Despite the simplicity of this interval, its actual coverage might be lower than 95%, given that the analytic variance formula $\widehat{\text{var}}(\hat{R}_t)$ is obtained from an asymptotic analysis that might be less accurate for small samples. Furthermore, if the threshold to obtain $\hat{R}_t$ comes from a threshold selection procedure subject to randomness from the sampling, such as the variance minimization approach presented above, then the estimated variance $\widehat{\text{var}}(\hat{R}_t)$ might underestimate the true variance of $\hat{R}_t$, and the analytical confidence interval might not actually have the promised coverage.

Using an analytical confidence interval along with the estimators obtained from the median sweep approach is more challenging, given that deriving the analytical variance of the median of correlated estimators is complex. Instead, we turn our attention to the bootstrap [4] as a flexible way of obtaining estimates of variances and confidence intervals.

Bootstrap Confidence Intervals The basis of the bootstrap [4] is to take samples with replacement from the original sample, of the same size as the original sample, and for each of these new samples repeat the estimation procedure. For instance, if we denote $\hat{R}_t^{\dagger(b)}$ the estimate obtained via variance minimization or median sweep from a bootstrap sample b, then we can use the bootstrap estimates obtained from B independent bootstrap samples, $\hat{R}_t^{\dagger(1)}, \ldots, \hat{R}_t^{\dagger(B)}$,

to compute confidence intervals in two ways. First, we can simply find the $\alpha/2$ and $1-\alpha/2$ quantiles of the bootstrap estimates, and take those as the bounds of the $100(1-\alpha)\%$ confidence interval; we refer to this as the *bootstrap quantile* approach. A second approach is to compute the variance of the bootstrap estimates, $\widehat{\text{var}}_{boot}(\hat{R}_t^\dagger)$, and use it to construct confidence intervals as $\hat{R}_t^\dagger \pm z_{1-\alpha/2}\sqrt{\widehat{\text{var}}}_{boot}(\hat{R}_t^\dagger)$, where $\hat{R}_t^\dagger$ is the estimate obtained via variance minimization or median sweep from the original sample; we refer to this as the *bootstrap standard error* approach. Given that in the estimator $\hat{R}_t = (\hat{R}_t^{raw} - \hat{p}_{1|0})/(\hat{p}_{1|1} - \hat{p}_{1|0})$, we assume that the variability from $\hat{R}_t^{raw}$ is negligible in comparison to the variability from $\hat{p}_{1|1}$ and $\hat{p}_{1|0}$, we only apply the bootstrap to the audited sample collected at baseline.

In the next section we compare the actual coverage of the five confidence intervals detailed here: for variance minimization we compute the analytical approach in addition to the two bootstrap approaches, whereas for median sweep we compare the two bootstrap confidence intervals.

3 Performance Comparison

3.1 Existing Estimators

Weighted rates of the form $R_t = \sum_{i=1}^{N_t} W_{it}Y_{it} / \sum_{i=1}^{N_t} W_{it}$ can be estimated using techniques for cost-quantification, as mentioned in Section 1.1: since in our use cases the weights W_{it} are known, we only need to estimate the total $\sum_{i=1}^{N_t} W_{it}Y_{it}$. To the best of our knowledge, the existing approaches for cost-quantification are due to Forman [7, 8, 10]. Here we consider two of those.

First, the *classify and total* (CT) approach simply replaces Y_{it} with $\hat{Y}_{it}$, and so this estimator leads to our $\hat{R}_t^{raw}$; we consider this estimator to show the reader how biased this approach can be. Second, the *grossed-up total* approach takes the CT estimator and multiplies it by the ratio $\hat{r}_t^{acc}/\hat{r}_t^{cc}$, where $\hat{r}_t^{cc} = \sum_{i=1}^{N_t} \hat{Y}_{it}/N_t$ is the classify and count estimator for the simple rate r_t, and $\hat{r}_t^{acc} = (\hat{r}_t^{cc} - \hat{p}_{1|0})/(\hat{p}_{1|1} - \hat{p}_{1|0})$ is its adjusted version. The resulting estimator for the weighted rate is $\hat{R}_t^{gut} = \hat{R}_t^{raw}\hat{r}_t^{acc}/\hat{r}_t^{cc}$. This approach is derived from a rule of three, that is, assuming that these ratios are equal: $\sum_{i=1}^{N_t} W_{it}Y_{it} / \sum_{i=1}^{N_t} W_{it}\hat{Y}_{it} = \sum_{i=1}^{N_t} Y_{it} / \sum_{i=1}^{N_t} \hat{Y}_{it}$.

The remaining approaches proposed by Forman [7, 8] for cost-quantification rely on the following idea. The total weight in the positive class can be written as $\sum_{i=1}^{N_t} W_{it}Y_{it} = \mu_t^+ N_t r_t$, where $r_t = \sum_{i=1}^{N_t} Y_{it}/N_t$ is the simple rate and $\mu_t^+ = \sum_{i=1}^{N_t} W_{it}Y_{it} / \sum_{i=1}^{N_t} Y_{it}$ is the mean weight among the positive class. If we know or have a good estimate of μ_t^+, then we can simply use quantification techniques to estimate r_t, and then estimate the total cost as $\mu_t^+ N_t \hat{r}_t$. In the applications studied by Forman [7, 8], it was reasonable to assume that μ_t^+ did not change over time, and so it could be estimated from the audited data at baseline. However, programs to improve data quality of e-commerce catalogs often target products with the largest weights, which directly impacts the value of

μ_t^+ over time. Because of this, we do not consider these approaches, as assuming that μ_t^+ is constant is unreasonable in our use cases.

3.2 Simulation Design

To compare the performance of the proposed and existing estimation approaches, we opt for conducting extensive simulation studies where we generate synthetic catalogs under a variety of scenarios that reflect characteristics of our use cases. We opt for this approach, given that we want to obtain an estimation strategy that can be reliably deployed across different circumstances, and a simulation study allows us to control the characteristics of the scenarios that we want to explore. Furthermore, given that we are restricted from publishing results obtained on datasets from our organization, creating synthetic scenarios that reflect characteristics of our use cases seems like a good compromise. We generate synthetic catalogs of size $N_t = 10^6$, and each simulation run involves one catalog for a baseline time $t = 0$ and one for a follow-up time $t > 0$. The exact details of their construction are given in Appendix C at `https://bit.ly/3wJK5Mj`, but here we present a brief description.

For baseline, a catalog is generated with a proportion of defective items, $r_0 = 0.1, 0.2, 0.3$. We then generate product weights using distributions obtained from actual numbers of visits to product pages in the Amazon.com website during a fixed time period and for a specific category of products. This is done such that the weighted rate R_0 is a specific fraction d of the proportion of defective products r_0. Given that for many of our use cases we expect defects to be more prevalent among products with lower weights, we expect $R_0 < r_0$. In particular, we take $R_0 = d\, r_0$ for $d = 1/4, 1/2, 3/4$. We generate synthetic product features to predict defects so that we obtain different levels of classification difficulty, here characterized by the true and false positive rates of the classifier; we consider three scenarios by fixing TPR=$0.5, 0.7, 0.9$ and FPR=0.05, which reflect a range of use cases, from cases where classifiers are in their infancy and do no yet reach high accuracy, to cases where mature classifiers have been developed and reach relatively high accuracy.

To generate the catalog at time $t > 0$, we fix different values of the percent change $\Delta = 100(R_t - R_0)/R_0$ of the weighted rate from time 0 to $t > 0$; we take $\Delta = -50\%, -25\%, +25\%, +50\%$ to cover a range of relatively large changes. The different combinations of Δ and R_0 considered here lead to a wide range of scenarios for the weighted rate R_t going from 1.25% to 33.75%, which is representative of the rates that we observe in our use cases.

Given a pair of synthetic catalogs for baseline and for time $t > 0$, we repeat 1000 times the estimation process of the weighted rate R_t with each of the competing estimation approaches. For all approaches, we start with sampling with replacement n_0 products from the baseline catalog, and record their ground truth values Y_{i0} (analog to auditing), along with their weights W_{i0} and model scores $g(X_{i0})$. We explore three sampling scenarios with $n_0 = 500, 1000, 2000$. In this simulation study we do not consider sampling from the catalog at time t, as we use the exact R_t^{raw} in computing the estimator (7), given that R_t^{raw} only

depends on the classifier predictions $\hat{Y}_{it} = I[g(X_{it}) > c]$, which do not involve auditing resources. If this is not tenable in practice, we need to estimate R_t^{raw} using a large sample such that its induced variability is negligible in comparison with the baseline sample.

3.3 Results

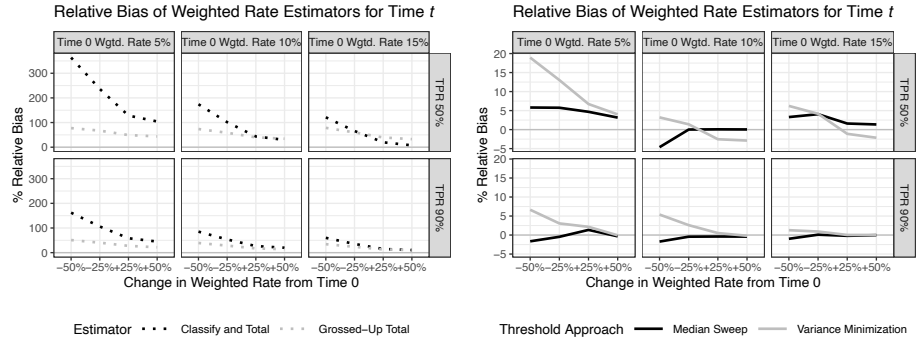
Estimators' Bias and Variance For each of the catalog scenarios described above, we summarize the performance of the different estimation approaches in terms of their bias and standard deviation. In Figures 1a and 1b we present the bias results for sample size $n_0 = 1000$ and for baseline weighted rates such that $R_0 = r_0/2$; the results for other n_0 and relationships between R_0 and r_0 are similar to the results presented here, in terms of leading to the same conclusions on which estimation approach is best. We also omit results for TPR=70%, as the performance is in between that of TPR=50% and TPR=90%. The vertical axis in the panels of Figures 1a and 1b show the estimation bias as a percentage of the true value R_t.

In Figure 1a we present the results for the classify and total, and the grossed-up total approaches [7, 8], which in some scenarios lead to relative bias of up to 350% and 90% respectively. The bias obtained from these approaches is too large to consider them reliable, and so we do not further study them.

In Figure 1b, we present the bias results for our proposed approaches, that is, estimator (7) along with median sweep (MS) or variance minimization (VM) to handle the classification threshold. The performance of the trimmed MS approach is virtually the same as the basic MS, so we omit it. To illustrate the results, consider the top left panel in Figure 1b, which shows a relative bias for the VM approach of almost 20% when the initial (baseline) weighted rate is 5% and the change is -50%, that is, when the weighted rate that we want to estimate at time t is 2.5%. In such case, a 20% relative bias means that the VM approach is on average returning 3% instead of 2.5%. While this is a small bias overall, the MS approach has relative biases of less than around 6% across all scenarios considered here. Undoubtedly, MS leads to a more reliable estimation approach in terms of bias, although the performance of the VM approach comes in close.

We can also see from comparing the rows of panels in Figure 1b that working with a high quality classifier (TPR=90%) generally leads to lower biases, especially when the weighted rates are small. Figure 1b also indicates that it is easier to unbiasedly estimate larger weighted rates. Another striking conclusion from looking at the first row of Figure 1b is that even with a very low quality classifier (TPR=50%) we can still obtain estimation approaches with relatively small bias, an encouraging sign of the reliability of the proposed estimation approaches for different use cases.

A reliable estimator should also have a small variance. In Figure 2 we present the standard deviation of the proposed estimation approaches under the same conditions presented for Figure 1b. We find that in most scenarios both VM and MS lead to nearly the same standard deviation, but VM can sometimes lead to higher estimation variance. This result seems counter-intuitive, given that



(a) Existing methods. (b) Proposed methods.

Fig. 1: Relative bias of classifier-based weighted rate estimation approaches. Note the different scales of the vertical axes.

by design VM should lead to the lowest variance. However, VM here uses an analytic approximation to the actual variance of the estimator based on large samples, which leads to an approach that does not actually reduce the estimation variance for small samples. Additionally, a factor that might contribute to the good performance of MS is that, in a sense it corresponds to an ensemble of classifiers, one per threshold in our grid, which are working together to estimate R_t; ensemble methods are known to both reduce bias and variance of learning algorithms [3].

Confidence Intervals' Coverage and Length We now present the performance of the five methods to build confidence intervals described in Section 2.4. If a procedure to construct confidence intervals truly leads to a confidence level of $100(1-\alpha)\%$, that means that if we were to repeat the measurement process (starting from random sampling) many times, then $100(1-\alpha)\%$ of those times the observed confidence interval would contain the true value of the parameter. Unfortunately, some confidence interval procedures might be misleading if their actual coverage is different from their nominal one. To ensure that a confidence interval procedure is reliable, it is customary to conduct a simulation study where we repeat the measurement process many times under a fixed set of conditions, and compute the actual coverage or confidence of the confidence intervals by computing the proportion of times that the intervals contain the true value of the parameter of interest. A confidence interval procedure is reliable if the actual coverage is around the nominal one.

In Figure 3 we present the actual coverage of the five confidence interval procedures described in Section 2.4. Undoubtedly, the bootstrap quantile confidence interval obtained from the median sweep procedure is the most reliable of these five approaches, given that its actual coverage is nearly the nominal 95%. In fact, the performance of the four bootstrap-based confidence intervals is gener-

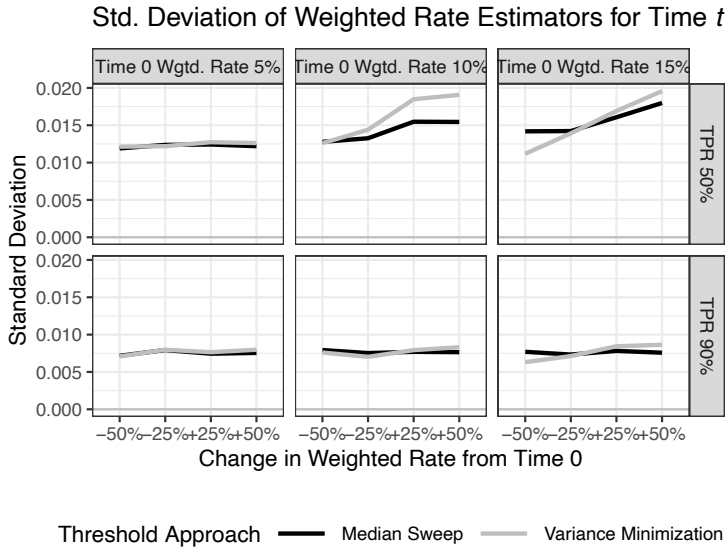


Fig. 2: Standard deviation of proposed weighted rate estimation approaches.

ally reasonable. The worst performance overall is obtained from the confidence interval based on the analytic approximate variance of the estimator obtained from the variance minimization approach. This might occur due to the analytic variance formula not accounting for the variability that comes from the threshold selection, which in turn leads to lower actual coverage of the analytic confidence interval.

Finally, an important property of a good confidence interval procedure is that it does not lead to unnecessarily wide confidence intervals. In this simulation study we also computed the average length of the confidence intervals obtained under each approach, and found that the average lengths are very similar for all approaches across all scenarios. In the interest of space, we do not present plots with these results.

Given these results, our final recommendation is to use median sweep to deal with the thresholds in the classifiers, and to use bootstrap quantile confidence intervals to quantify the uncertainty in the estimation.

4 Discussion and Extensions

Our proposed estimation approach, using median sweep to deal with the thresholds of the classifiers and bootstrap confidence intervals to quantify estimation uncertainty, is currently being implemented in our organization to produce estimates of weighted rates for several types of catalog defects. Our implementation

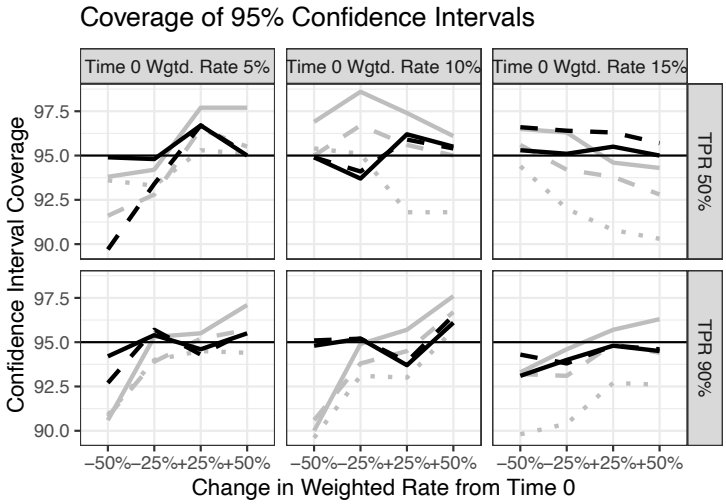


Fig. 3: Actual coverage of nominal 95% confidence intervals (CIs). CIs based on variance minimization: grey dotted lines: analytic confidence interval; grey dashed lines: bootstrap standard error; grey solid lines: boostrap quantiles. CIs based on median sweep: black dashed lines: bootstrap standard error; black solid lines: boostrap quantiles.

consists of measurement cycles, which are marked by baseline times, when we collect audited data, and followed up by automated estimation.

Intuitively, for follow-up times close to the baseline time of the cycles, the proposed estimation approach should be reliable, given that the extrapolation assumption (EA) should approximately hold. As we move farther away from the baseline, the EA might become more questionable. In our use cases, we plan to start from short measurement cycles, say monthly periods, and based on the audited data test the hypothesis of whether the TPR and FPR are the same at the beginning of the cycles. If we repeatedly fail to reject the hypothesis, we expand the measurement cycles, as this indicates that the EA holds for longer in that particular use case.

Regarding the simplifying assumption (SA) used to derive our proposed estimator, it says that the TPR and FPR do not depend on the product weights. This seems initially reasonable, given that the classifiers that we work with use product features exclusively, and not measures of engagement of customers with the products. Nevertheless, the SA can be examined using audited data, for instance by regressing the predicted indicators of defects on the weights, separately for audited products with and without the defect. For use cases when there is evidence of an association, a simple solution is to stratify the estimation domain based on weight intervals, proceed with the estimation as described here separately within each stratum, and aggregate the per-stratum estimates to obtain an

overall estimate of the weighted rate, where the aggregation is done weighting the strata by their relative share of the products' weights. This stratified approach requires the SA to hold within stratum, which is more tenable. Intuitively, in the extreme case where there is one stratum per weight value the assumption holds exactly. However, while estimation based on a very fine stratification will alleviate the bias induced by violating SA, it will lead to a large estimation variance. Selecting the right stratification then involves a bias-variance tradeoff which will change depending on the use case.

Acknowledgements The authors would like to thank George Forman, Bunyamin Sisman, Kee Kiat Koo, Florian Verhein, Wenyi Wu, and Vito Mandorino for their insightful feedback.

References

1. Amazon Seller Central: Suggest changes to your product detail page. `https://sellercentral.amazon.com/gp/help/external/200335450?ref=efph_200335450_cont_201950630&language=en_US`, accessed: 2022-06-13
2. Bishop, Y.M.M., Fienberg, S.E., Holland, P.W.: Discrete Multivariate Analysis: Theory and Practice. Springer (1974)
3. Dietterich, T.G.: Ensemble learning. In: The Handbook of Brain Theory and Neural Networks, p. 405. 2nd edn. (2003)
4. Efron, B., Tibshirani, R.: Bootstrap methods for standard errors, confidence intervals, and other measures of statistical accuracy. Statistical Science pp. 54–75 (1986)
5. Fawcett, T., Flach, P.A.: A response to webb and ting's "on the application of roc analysis to predict classification performance under varying class distributions". Machine Learning **58**(1), 33–38 (2005)
6. Forman, G.: Counting positives accurately despite inaccurate classification. In: Proceedings of the European Conference on Machine Learning (ECML'05). pp. 564–575 (2005)
7. Forman, G.: Quantifying trends accurately despite classifier error and class imbalance. In: Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (SIGKDD'06). pp. 157–166. ACM (2006)
8. Forman, G.: Quantifying counts and costs via classification. Data Min. Knowl. Discov. **17**(2), 164–206 (2008)
9. Gart, J.J., Buck, A.A.: Comparison of a screening test and a reference test in epidemiologic studies ii. a probabilistic model for the comparison of diagnostic tests. Am. J. Epidemiol. **83**(3), 593–602 (1966)
10. González, P., Castaño, A., Chawla, N.V., Coz, J.J.D.: A review on quantification learning. ACM Computing Surveys **50**(5), 74 (2017). https://doi.org/https://doi.org/10.1145/3117807
11. Grassia, A., Sundberg, R.: Statistical precision in the calibration and use of sorting machines and other classifiers. Technometrics **24**(2), 117–121 (1982)
12. Levy, P.S., Kass, E.H.: A three-population model for sequential screening for bacteriuria. Am. J. Epidemiol. **91**(2), 148–154 (1970)

13. Maletzke, A., dos Reis, D., Cherman, E., Batista, G.: Dys: A framework for mixture models in quantification. In: The Thirty-Third AAAI Conference on Artificial Intelligence (AAAI-19). pp. 4552–4560. AAAI (2019)
14. Meertens, Q.A., Diks, C.G.H., van den Herik, H.J., Takes, F.W.: Improving the output quality of official statistics based on machine learning algorithms (2021)
15. Schumacher, T., Strohmaier, M., Lemmerich, F.: A comparative evaluation of quantification methods (2021)
16. Särndal, C.E., Swensson, B., Wretman, J.: Model Assisted Survey Sampling. Springer-Verlag Publishing (1992)
17. Tasche, D.: Minimising quantifier variance under prior probability shift (2021)
18. Vaz, A.F., Izbicki, R., Stern, R.B.: Quantification under prior probability shift: the ratio estimator and its extensions. Journal of Machine Learning Research **20**(79), 1–33 (2019)
19. Wasserman, L.: All of statistics: a concise course in statistical inference. Springer Science & Business Media (2013)

On Multi-Class Extensions of Adjusted Classify and Count

Mirko Bunse[0000−0002−5515−6278]

Artificial Intelligence Unit, TU Dortmund University, 44227 Dortmund, Germany
mirko.bunse@cs.tu-dortmund.de

Abstract. Adjusted Classify and Count (ACC) is one of the most widely acknowledged methods for quantification, the supervised learning task of predicting the class prevalences in a data sample. While ACC stems from binary quantification, where only two classes are considered, several different multi-class extensions have been proposed. In this work, we compare four existing multi-class extensions, both conceptually and empirically. Moreover, we propose a novel multi-class extension that employs an un-constrained least squares optimization with the aid of a soft-max layer. Our empirical results on a recent benchmark data set demonstrate that numerical optimization techniques for multi-class ACC, like our proposed method, outperform analytic solutions.

Keywords: Quantification · Multi-class classification · Constrained optimization · Unconstrained optimization

1 Introduction

Quantification [8] is the task of predicting the prevalence of each class in a data sample. This supervised learning task is in contrast to "standard" classification learning, where predictions for individual data items, and not for a sample of items, are desired to be accurate. Applications of quantification arise in text sentiment analyses [9], in the social sciences [11], in astroparticle physics [4], and in several other areas.

One of the most widely acknowledged methods for quantification is the *Adjusted Classify and Count* (ACC) technique [8], which was initially proposed for binary quantification in particular. For the multi-class setting, there are at least four different extensions to binary ACC: one-versus-all decomposition [8], matrix inversion [12, 17], pseudo-inversion [14], and constrained least squares [3, 7, 11]. ACC has desireable properties, such as Fisher consistency [16] and computational efficiency.

In this work, we discuss the four existing alternatives and we propose a novel multi-class ACC extension. Our proposal employs un-constrained least squares with the aid of a soft-max layer. We compare the five multi-class extensions empirically on a gold-standard benchmark from the LeQua2022 competition [6]. Our reusable implementation is available online.[1]

[1] https://github.com/mirkobunse/QUnfold.jl

Sec. 2 introduces binary ACC and Sec. 3 presents the multi-class extensions. Our experiments are discussed in Sec. 4 before we conclude in Sec. 5.

2 Adjusted Classify and Count in Binary Quantification

In the following, we revisit four fundamental methods for *binary* quantification, where predictions $\hat{y}_i \in \{-1, +1\}$ take one of two values. In the binary setting, the goal is to predict the prevalence of the positive class, $\mathbb{P}(Y = +1)$, in a sample with N data items. We start with the un-adjusted methods before we detail the adjustment rule that yields consistent quantifiers.

First, the (un-adjusted) Classify and Count (CC) method [8] estimates the prevalence $\mathbb{P}(Y = +1)$ from predictions that are issued for each individual data item in a sample, i.e.

$$p^{(\mathrm{CC})} \;=\; \frac{1}{N} \sum_{i=1}^{N} \mathbb{1}_{\hat{y}_i = +1} \,. \tag{1}$$

At this point, the crisp predictions $\hat{y}_i$ can be replaced with estimates of the posterior probabilities [2], which several classification methods return as an indicator of uncertainty. This proposal leads to the (un-adjusted) Probabilistic Classify and Count (PCC) estimate

$$p^{(\mathrm{PCC})} \;=\; \frac{1}{N} \sum_{i=1}^{N} \widehat{\mathbb{P}}\big(\widehat{Y} = +1 \mid X = \vec{x}_i\big) \,. \tag{2}$$

CC and PCC are easily extended to multi-class settings, where each component $[\vec{p}]_i$ of a vector $\vec{p} \in \mathbb{R}^C$ estimates the prevalence of one class $i \in \{1, \ldots, C\}$, as according to Eq. 1 or Eq. 2.

Unfortunately, it is well-acknowledged that CC and PCC are susceptible to prior probability shift [8, 16], due to imperfections of the underlying classifier. In particular, Eqs. 1 and 2 will systematically over- or under-estimate the true class prevalences if these prevalences deviate from the ones that are used during the training of the classifier. In quantification, these prevalences are typically not known a priori, so that prior probability shift must be expected. Therefore, CC and PCC are not appropriate solutions for the quantification problem.

In binary quantification, we can correct this deficiency through an adjustment rule. This rule leads to the *Adjusted Classify and Count* (ACC) method [8], one of the most widely acknowledged techniques for handling prior probability shift in quantification. Binary ACC estimates $\mathbb{P}(Y = +1)$ as

$$p^{(\mathrm{ACC})} \;=\; \frac{p^{(\mathrm{CC})} - \mathrm{FPR}}{\mathrm{TPR} - \mathrm{FPR}}, \tag{3}$$

where $\mathrm{FPR} = \mathbb{P}(\widehat{Y} = +1 \mid Y = -1)$ is the false positive rate of the underlying classifier and $\mathrm{TPR} = \mathbb{P}(\widehat{Y} = +1 \mid Y = +1)$ is the true positive rate. Both of these rates need to be estimated on hold-out data that is not used during the

training of the classifier; otherwise, overfitting of $p^{(\mathrm{ACC})}$ is likely. If the number of falsely positive predicted instances and the number of falsely negative predicted instances are equal, one can return $p^{(\mathrm{CC})}$ without making an adjustment.

The adjustment rule from Eq. 3 can also be applied to $p^{(\mathrm{PCC})}$ instead of $p^{(\mathrm{CC})}$. In this case, the adjustment rule yields the *Probabilistic Adjusted Classify and Count* (PACC) method [2].

Binary ACC and PACC have desireable properties. Most importantly, they are Fisher consistent estimators of $\mathbb{P}(Y = +1)$ even under prior probability shift [16]. Moreover, they are computationally efficient: a prediction requires only a single pass over the data sample to compute $p^{(\mathrm{CC})}$ or $p^{(\mathrm{PCC})}$; so does the computation of FPR and TPR during training. Hence, multi-class extensions to the binary ACC and PACC are promising topics for quantification research.

3 Multi-Class Extensions of Adjusted Classify and Count

In multi-class quantification with $C > 2$ classes, the goal is to estimate a vector $\vec{p} \in \mathcal{P}$ of class prevalences, where the set of feasible solutions

$$\mathcal{P} \;=\; \left\{ \vec{p} \in \mathbb{R}^C \;:\; [\vec{p}]_i \geq 0 \;\forall\; 1 \leq i \leq C \;\;\wedge\;\; 1 = \sum_{i=1}^{C} [\vec{p}]_i \right\} \tag{4}$$

is the unit simplex. All solutions within this set are valid probability densities.

In the following, we detail four existing multi-class extensions of ACC. We further propose one additional extension, an un-constrained least squares estimate which employs a soft-max layer.

Tab. 1 displays a summary of the conceptual properties of these extensions. In this table, we emphasize that each row respectively extends its preceeding row only in terms of a single aspect. Therefore, we recognize all of these methods as being "true" ACC extensions, rather than being independent methods.

Table 1. Adjustments in multi-class ACC extensions.

adjustment	basis	loss function	constraints	optimization
one-vs-rest (Eq. 5)	TPR_i, FPR_i	—	—	—
inverse (Eq. 8)	M	—	—	—
pseudo-inverse (Eq. 9)	M	least squares	min. norm	—
constrained (Eq. 10)	M	least squares	$\mathcal{P}$	constrained
soft-max (Eq. 11)	M	least squares	$\mathcal{P}$	unconstrained

3.1 One Versus Rest Decomposition

The most straightforward extension of binary ACC decomposes the multi-class quantification problem into C one-versus-rest tasks [8]. Each of these tasks requires a binary quantification of one class versus all others. Hence, we can use the

binary adjustment rule from Eq. 3 in each of the tasks separately. The resulting estimate is $\vec{p}^{\,(\text{one-vs-rest})} \in \mathbb{R}^C$, where

$$[\vec{p}^{\,(\text{one-vs-rest})}]_i \;=\; \frac{[\vec{p}^{\,(\text{CC})}]_i - \text{FPR}_i}{\text{TPR}_i - \text{FPR}_i} \tag{5}$$

is the i-th component of $\vec{p}^{\,(\text{one-vs-rest})}$. Here, $[\vec{p}^{\,(\text{CC})}]_i$ is the CC estimate for the i-th class. Moreover, TPR_i and FPR_i are the true positive rate and the false positive rate when class i is classified against all other classes.

Like in binary ACC, the estimate from Eq. 5 requires clipping to ensure that each component is between 0 and 1. Moreover, this estimate requires normalization to ensure that the sum of all components is one. Unfortunately, these ad-hoc corrections can lead to estimation errors if the data sets are not sufficiently large to accurately estimate $[\vec{p}^{\,(\text{CC})}]_i$, TPR_i, and FPR_i.

3.2 Matrix Inversion

A multi-class classifier can confuse each pair of classes with a non-zero probability. The confusion matrix $M \in \mathbb{R}^{C\times C}$ of a classifier comprises all of these probabilities in the matrix cells

$$[M]_{ij} \;=\; \mathbb{P}\big(\widehat{Y} = i \mid Y = j\big). \tag{6}$$

The matrix of ground-truth confusion probabilities, which are typically unknown, defines the CC outcome

$$\vec{p}^{\,(\text{CC})} \;=\; M \cdot \vec{p}, \tag{7}$$

from the ground-truth prevalence vector $\vec{p} \in \mathcal{P}$. Consequently, we can recover an estimate of the true $\vec{p}$ with an estimate of the confusion matrix M.

The most straightforward attempt in this direction [12, 17] is to invert an estimate of M to yield the prevalence estimate

$$\vec{p}^{\,(\text{inverse})} \;=\; M^{-1} \cdot \vec{p}^{\,(\text{CC})}. \tag{8}$$

For instance, this matrix inversion estimate is implemented in the current release[2] of QuaPy [13]. Since QuaPy is likely the most complete and usable software package for quantification, this choice has established $\vec{p}^{\,(\text{inverse})}$ as the "quasi-standard" multi-class extension of ACC and PACC.

However, the inverse of an estimated M is not guaranteed to exist. In this case, the estimator is undefined. QuaPy deals with this issue by falling back to the un-adjusted $\vec{p}^{\,(\text{CC})}$ and $\vec{p}^{\,(\text{PCC})}$ if M is not invertible.

3.3 Pseudo-Inversion

A robust alternative to matrix inversion replaces the actual inverse M^{-1} with the Moore-Penrose pseudo-inverse $M^{\dagger}$. This replacement leads to the estimate

$$\vec{p}^{\,(\text{pseudo-inverse})} \;=\; M^{\dagger} \cdot \vec{p}^{\,(\text{CC})}, \tag{9}$$

[2] QuaPy, v0.1.6: https://github.com/HLT-ISTI/QuaPy/releases/tag/0.1.6

which is always defined because $M^\dagger$ is always guaranteed to exist. Moreover, $M^\dagger$ is equal to M^{-1} if M^{-1} exists. Hence, the replacement does not reduce the quality of the estimate. It gains robustness because no fallback to an un-adjusted $\vec{p}^{\,(\mathrm{CC})}$ or $\vec{p}^{\,(\mathrm{PCC})}$ is necessary if M is not invertible.

The pseudo-inverse estimator is proven to be a least-squares estimate of the true $\vec{p}$, which is constrained to the minimum norm estimate [14, Th. 4.1]. This constraint has the advantage that $\vec{p}^{\,(\mathrm{pseudo-inverse})}$ is unique. However, a minimum norm constraint lacks motivation from a practical perspective; in fact, the constraint is unrelated to the actual feasible set $\mathcal{P}$ from Eq. 4.

3.4 Constrained Least Squares

Both inversion techniques $\vec{p}^{\,(\mathrm{inverse})}$ and $\vec{p}^{\,(\mathrm{pseudo-inverse})}$ suffer from not being constrained to the feasible set $\mathcal{P}$ from Eq. 4. In fact, both techniques tend to produce estimates that i) do not sum to one and ii) have components that are less than zero. This deficiency is typically addressed through clipping and normalization, an ad-hoc correction that can lead to estimation errors.

A more appropriate approach is presented by Hopkins and King [11]. They propose a constrained optimization task

$$\vec{p}^{\,(\mathrm{constrained})} \;=\; \underset{\vec{p}\in\mathcal{P}}{\arg\min} \left\| \vec{p}^{\,(\mathrm{CC})} - M \cdot \vec{p} \right\|_2^2, \tag{10}$$

which explicitly constrains the estimate to the space $\mathcal{P}$ of valid probabilities. Within this space, the most accurate estimate according to the L_2 norm is searched for. Hence, $\vec{p}^{\,(\mathrm{constrained})}$ employs the same loss function as the estimate $\vec{p}^{\,(\mathrm{pseudo-inverse})}$, but uses a more appropriate set of constraints.

Unfortunately, Hopkins and King [11] do not propose a specific algorithm to solve Eq. 10. While an analytical solution exists for the unit sum constraint $1 = \sum_{i=1}^{C} [\vec{p}]_i$ [1, Chap. 1.4], we are not aware of an analytic solution that considers the inequality constraints $[\vec{p}]_i \geq 0 \;\forall\; 1 \leq i \leq C$ from Eq. 4.

Consequently, the optimization of Eq. 10 requires numerical optimization techniques. In our implementation, we employ a primal-dual interior-point algorithm with a filter line search [18]. However, other numerical methods are conceivable at this point. For instance, Firat [7] employs a sequential quadratic programming technique [19, Chap. 18] to solve Eq. 10. We leave a comparison of numerical optimization techniques in quantification to future work.

3.5 Unconstrained Least Squares with a Soft-Max Layer

We now propose a novel multi-class extension of binary ACC. The goal of our proposal is to rephrase the optimization task from Eq. 10 to achieve an unconstrained optimization task which produces valid probability densities despite being unconstrained. The desire to optimize without constraints is rooted in our subjective perception that unconstrained optimization is an easier problem than constrained optimization. Hence, we hope for less noise in the gradients that are computed during the optimization process.

We obtain an unconstrained optimization task through a soft-max layer, which is a derivable operation that transforms latent variables into probability densities. We maintain the least squares loss function from Eq. 10. Our multi-class ACC is defined over latent variables $\vec{l} \in \mathbb{R}^C$, as

$$\begin{aligned} \vec{p}^{\,(\mathrm{soft-max})} &= \mathrm{softmax}(\vec{l}^{\,*}), \\ \vec{l}^{\,*} &= \operatorname*{arg\,min}_{\vec{l} \in \mathbb{R}^C} \left\| \vec{p}^{\,(\mathrm{CC})} - M \cdot \mathrm{softmax}(\vec{l}) \right\|_2^2 + \lambda \cdot \left\| \vec{l} \right\|_2^2, \\ [\mathrm{softmax}(\vec{l})]_i &= \frac{\exp([\vec{l}]_i)}{\sum_{j=1}^{C} \exp([\vec{l}]_j)}, \end{aligned} \tag{11}$$

where $\lambda \cdot \|\vec{l}\|_2^2$ is a regularization term that ensures all $\exp([\vec{l}]_i)$ to be finite within floating point precision. This regularization term is only a technical detail: it affects the latent variables $\vec{l}$, but not the estimate $\vec{p}^{\,(\mathrm{soft-max})}$, which is always in $\mathcal{P}$ due to the soft-max layer. In our experiments, we fix $\lambda = 10^{-6}$.

4 Experiments

In the following, we intend to uncover the merits and the disadvantages of the above multi-class extensions of ACC and PACC. To this end, we evaluate their performance on the public data set [5] of the LeQua2022 competition [6]. Our reusable Julia implementation of methods and experiments is available online.[1]

The LeQua2022 dataset is designed to constitute a gold-standard benchmark, both for binary text quantification and for multi-class text quantification. The multi-class problem in this competition features 28 classes, 20 000 training items and 1 000 validation samples. Each of the validation samples consists of 1 000 data items that are drawn according to varying class prevalences. We employ the vectorial representation of the data and a logistic regression classifier, which obtained the highest performance on this representation during the competition [15]. We optimize the regularization parameter of this classifier on the validation set and over the grid $\{10^{-3}, 10^{-2}, 10^{-1}, 10^{0}, 10^{1}\}$, to obtain the best performance for each quantification method. The selection of the best regularization parameter is either in terms of the absolute error (AE) or in terms of the relative absolute error (RAE). We report the results, in terms of both metrics, on the test set.

All multi-class extensions of ACC require the estimation of the confusion matrix M (or at least the rates TPR_i and FPR_i) on hold-out data. In order to use all labeled data for classifier training and for the adjustments, we use a bagging ensemble of size 100. We estimate M, TPR_i, and FPR_i on the out-of-bag predictions of this ensemble.

During the hyper-parameter optimization on the validation set, almost all methods succeeded in producing estimates for the class prevalences. An exception to this outcome is the matrix inversion from Eq. 8 in ACC. This method failed to produce prevalence estimates for the values 10^{-3} and 10^{-2} of the classifier's

regularization parameter because these values led to confusion matrices M that were *not* invertible.

Table 2. Test set performance of the different multi-class adjustments, for ACC and PACC and in terms of AE and RAE. The performance of the best adjustment in each setting is printed in boldface.

adjustment	ACC		PACC	
	AE	RAE	AE	RAE
un-adjusted (Eq. 1 / Eq. 2)	0.0254	2.5532	0.0246	2.6771
one-vs-rest (Eq. 5)	0.0262	4.1484	0.0262	4.1484
inverse (Eq. 8)	0.0222	1.7224	0.0195	1.5288
pseudo-inverse (Eq. 9)	0.0177	1.7224	0.0195	1.5288
constrained (Eq. 10)	0.0158	1.2826	0.0123	**0.9908**
soft-max (Eq. 11)	**0.0130**	**1.2633**	**0.0106**	1.0886

Discussion The results from Tab. 2 demonstrate that the different multi-class adjustments exhibit quite different performances, in general. The lowest errors are achieved by the constrained estimator from Eq. 10 (in terms of RAE in PACC) and by our unconstrained soft-max estimator from Eq. 11 (in terms of all other configurations). The margins of improvement over all other adjustments are considerable: for instance, the constrained PACC achieves an RAE that is 38% smaller than the RAE of the pseudo-inverse PACC (last column, 0.9669 vs 1.5536); our soft-max PACC achieves an AE that is 46% smaller than the AE of the pseudo-inverse PACC (third column, 0.0106 vs 0.0197).

5 Conclusions and Outlook

We have discussed five different multi-class extensions of the binary adjustment that is employed in ACC and PACC. One of these extensions is an original proposal by us; this proposal employs a soft-max layer to circumvent the constraints that are otherwise required to obtain valid solutions in a numerical optimization process. Our proposal and an existing constrained least squares adjustment [3, 7, 11] deliver the most competitive performances.

Future work should compare different optimization techniques [19] to solve the constrained optimization task and our unconstrained soft-max proposal. Our "trick" of using a soft-max layer in quantification is also applicable to other methods, like ReadMe [11] and HDx / HDy [10], where it should be evaluated.

References

1. Amemiya, T.: Advanced econometrics. Blackwell (1985)

2. Bella, A., Ferri, C., Hernández-Orallo, J., Ramírez-Quintana, M.J.: Quantification via probability estimators. In: Int. Conf. on Data Mining. pp. 737–742. IEEE (2010). https://doi.org/10.1109/ICDM.2010.75
3. Bunse, M., Morik, K.: Unification of algorithms for quantification and unfolding. In: Workshop on Mach. Learn. for Astropart. Phys. and Astron. Gesellschaft für Informatik e.V. (2022), to appear
4. Bunse, M., Piatkowski, N., Morik, K., Ruhe, T., Rhode, W.: Unification of deconvolution algorithms for Cherenkov astronomy. In: Int. Conf. on Data Sci. and Adv. Anal. pp. 21–30. IEEE (2018). https://doi.org/10.1109/DSAA.2018.00012
5. Esuli, A., Moreo, A., Sebastiani, F.: Learning to quantify: LeQua 2022 datasets (2021). https://doi.org/10.5281/zenodo.6546188
6. Esuli, A., Moreo, A., Sebastiani, F.: A detailed overview of LeQua@CLEF 2022: Learning to quantify. In: Conf. and Labs of the Eval. Forum. pp. 1849–1868. CEUR Workshop Proc. (2022), http://ceur-ws.org/Vol-3180/paper-146.pdf
7. Firat, A.: Unified framework for quantification. arXiv:abs/1606.00868 (2016)
8. Forman, G.: Quantifying counts and costs via classification. Data Mining and Knowl. Discov. **17**(2), 164–206 (2008). https://doi.org/10.1007/s10618-008-0097-y
9. Gao, W., Sebastiani, F.: From classification to quantification in tweet sentiment analysis. Soc. Netw. Anal. and Mining **6**(19), 1–22 (2016)
10. González-Castro, V., Alaíz-Rodríguez, R., Alegre, E.: Class distribution estimation based on the Hellinger distance. Inf. Sci. **218**, 146–164 (2013). https://doi.org/10.1016/j.ins.2012.05.028
11. Hopkins, D.J., King, G.: A method of automated nonparametric content analysis for social science. Amer. J. of Polit. Sci. **54**(1), 229–247 (2010). https://doi.org/10.1111/j.1540-5907.2009.00428.x
12. McLachlan, G.J.: Discriminant analysis and statistical pattern recognition. Wiley (1992)
13. Moreo, A., Esuli, A., Sebastiani, F.: Quapy: A python-based framework for quantification. In: Int. Conf. on Inf. and Knowl. Management. pp. 4534–4543. ACM (2021). https://doi.org/10.1145/3459637.3482015
14. Mueller, J.L., Siltanen, S.: Linear and Nonlinear Inverse Problems with Practical Applications, Computational science and engineering, vol. 10. SIAM (2012). https://doi.org/10.1137/1.9781611972344
15. Senz, M., Bunse, M.: DortmundAI at LeQua 2022: Regularized SLD. In: Conf. and Labs of the Eval. Forum. CEUR Workshop Proc., vol. 3180, pp. 1911–1915 (2022), http://ceur-ws.org/Vol-3180/paper-152.pdf
16. Tasche, D.: Fisher consistency for prior probability shift (2017)
17. Vucetic, S., Obradovic, Z.: Classification on data with biased class distribution. In: Eur. Conf. on Mach. Learn. pp. 527–538. Springer (2001). https://doi.org/10.1007/3-540-44795-4_45
18. Wright, S.J., Nocedal, J.: Numerical optimization. Operations Research and Financial Engineering, Springer, 2 edn. (2006)
19. Wächter, A., Biegler, L.T.: On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. Math. programming **106**(1), 25–57 (2006)