# Ordinal Quantification through Regularization

**Abstract.** Quantification, i.e., the task of training predictors of the class prevalence in sets of unlabelled data items, has received increased attention in recent years. However, most quantification research has concentrated on developing algorithms for binary and multi-class problems in which the classes are not ordered. We here study the ordinal case, i.e., the case in which a total order is defined on the set of classes. We give three main contributions to this field. First, we create and make available two datasets for ordinal quantification (OQ) research that overcome the inadequacies of the previously available ones. Second, we experimentally compare, on the above datasets, the most important OQ algorithms proposed in the literature so far. To this end, we consider algorithms that have been proposed by authors from different research fields, who were unaware of each other's developments. Third, we propose three OQ algorithms, based on the idea of preventing ordinally implausible estimates through regularization. We show experimentally that these algorithms outperform the existing ones.

**Keywords:** Quantification · Ordinal classification · Supervised prevalence estimation

## 1 Introduction

*Quantification* (a.k.a. *learning to quantify*, or *supervised prevalence estimation*, or *class prior estimation*) is a supervised learning task which consists of training (on a set $L$ of labelled data items) a predictor that returns estimates $\hat{p}_\sigma(y_i)$ of the relative frequencies (a.k.a. *prevalence values*, or *prior probabilities*) $p_\sigma(y_i)$ of the classes of interest $\mathcal{Y} = \{y_1, ..., y_n\}$ in a sample $\sigma$ of unlabelled data items (González et al., 2017). Another way of saying this is that a trained *quantifier* (i.e., an estimator of class prevalence values) must return a *predicted distribution* $\hat{p}$ of the unlabelled data items across the classes in $\mathcal{Y}$, where this predicted distribution must diverge as little as possible from the true (unknown) distribution $p$.

Quantification is important in many disciplines (such as e.g., market research, political science, the social sciences, epidemiology) which, by their very own nature, are only interested in aggregate (as opposed to individual) data. In these contexts, classifying individual unlabelled instances is usually not a primary goal, while estimating the prevalence $p(y_i)$ of the classes of interest $\mathcal{Y} = \{y_1, ..., y_n\}$ in the data is. For instance, when classifying the tweets about a certain entity (e.g., a political candidate) as displaying either a Positive or a Negative stance towards the entity, we are usually not much interested in the class of a specific tweet, and we want instead to know the fraction of these tweets that belong to the class (Gao and Sebastiani, 2016).

Generating a predicted distribution $\hat{p}$ could in principle be achieved by the "classify and count" method (CC), i.e., by training a standard classifier, classifying all the unlabelled data items in the sample $\sigma$, counting how many data items have been attributed to each class in $\mathcal{Y}$, and normalising. However, it has been shown that CC delivers poor prevalence estimates, and especially so when the application scenario suffers from *distribution shift* (Moreno-Torres et al., 2012), the (ubiquitous) phenomenon according to which the distribution $p_U(y_i)$ of the unlabelled test documents $U$ across the classes is different from the distribution $p_L(y_i)$ of the labelled training documents $L$. As a result, a plethora of quantification methods have been proposed in the literature (see (González et al., 2017)) that attempt to return accurate class prevalence estimations even in the presence of distribution shift.

However, the vast majority of the methods proposed deal with the "categorical" quantification task in which $\mathcal{Y}$ is a plain, unordered set; this essentially means the standard binary ($n = 2$) or multiclass ($n > 2$) quantification tasks. Very few methods, instead, deal with *ordinal quantification* (OQ), the (much less standard) task of performing quantification on a set of $n > 2$ classes on which a total order "$\prec$" is defined. Ordinal quantification is important, though, because ordinal scales arise in many applications, especially ones involving human judgments. For instance, in a customer satisfaction endeavour one may want to estimate how a set of reviews of a certain product distribute across the set of classes $\mathcal{Y}$ ={1Star, 2Stars, 3Stars, 4Stars, 5Stars}, while a social scientist might want to find out how inhabitants of a certain region are distributed in terms of their happiness with health services in the area ($\mathcal{Y}$ ={VeryUnhappy, Unhappy, Happy, VeryHappy}).

In this paper we contribute to the field of OQ in a number of ways.

First, we develop and make publicly available two datasets for evaluating OQ algorithms, one consisting of textual product reviews and one consisting of telescope observations. Both datasets are from scenarios in which OQ arises naturally, and are generated according to a strong, well-tested protocol for the generation of datasets oriented to the evaluation of quantifiers. This contribution fills a gap, because datasets previously used for the evaluation of OQ were not adequate, for reasons that we discuss in Sec. 2.

Second, we perform an extensive experimental comparison (using the two previously mentioned datasets) among all the OQ algorithms that (to the best of our knowledge) have previously been proposed in the literature; this is important, since some of these algorithms (e.g., the ones of Sec. A.1 and A.2) had been compared with each other on a testbed that was likely inadequate, while some other algorithms (e.g., the ones of Sec. 3.2.1 to 3.2.2) had been developed independently (i.e., in the unawareness) of the previous ones, and had thus never been compared with them.

Third, we propose new OQ algorithms, which introduce regularization into existing quantification methods. We experimentally compare our proposals with the existing state of the art and make the corresponding code publicly available[1].

Experimental physics often has the objective to estimate the distribution of a physical quantity that is measured only indirectly, through correlated quantities. This objective corresponds to a quantification problem because i) the relevant quantity needs to be predicted from the measurements; and ii) the distribution of this quantity, as exhibited by a sample, is the central item of interest. Moreover, this quantification problem is of an ordinal nature because the relevant quantity typically obeys a total order. Early on, physicists have termed this problem "unfolding" (Blobel, 1985; D'Agostini, 1995), which prevented researchers from drawing connections between algorithms that have been proposed in the quantification literature and algorithms that have been proposed in the physics literature. In the following, we provide these connections to find that regularization techniques from physics are able to improve well-known quantification methods in ordinal settings.

Physicists are typically interested in the distribution of continuous quantities, rather than ordered classes. However, a histogram approximation of a continuous distribution is sufficient for many physics analyses (Blobel, 2002). Accordingly, all the unfolding algorithms we consider here evolve around histograms instead of continuous distributions. This conventional simplification essentially maps the values of a continuous target quantity to a set of bins with a total order. Since the values of this quantity are not known, but must be predicted, it is appropriate to consider these bins as totally ordered classes $\mathcal{Y}$ in a classification task. From this consideration, it happens that many unfolding algorithms in fact approach the general OQ problem—quite successfully, as our experiments of Sec. 4 show.

The paper is organized as follows. In Sec. 2 we review past work on OQ. In Sec. 3 we present all the OQ methods discussed in this paper, starting with previously proposed ones (Sec. A) and carrying on with the novel ones we propose in this work (Sec. 3.3). Sec. 4 is devoted to our experimental evaluation; in particular, Sec. 4.2 presents the two datasets that we here make available and that we use for the experimentation, while Sec. 4.4 presents the results of the experiments. Sec. 5 concludes, discussing avenues for future research.

## 2 Related work

Quantification, as a task of its own, was first proposed by Forman (2005), who observed that some applications of classification methods only require the estimation of class prevalence values, and that better methods than "classify and count" can be devised for this requirement. Since then, many methods for quantification have been proposed; however, most of these methods tackle the categorical case, in its binary and/or in its multiclass incarnations.

[1] A public GitHub link will be provided in the camera-ready version; for now, the code is part of our supplementary material.

Ordinal quantification was first discussed by Esuli and Sebastiani (2010). However, it was not until 2016 that the first true OQ algorithms were developed, the *Ordinal Quantification Tree* (OQT) by Da San Martino et al. (2016) and the *Adjusted Regress and Count* (ARC) algorithm by Esuli (2016). In the same years, the first data analysis competitions that involved OQ were proposed (Higashinaka et al., 2017; Nakov et al., 2016; Rosenthal et al., 2017). However, with the exception of OQT and ARC, the participants in these competitions preferred "classify and count" with highly optimised classifiers over true OQ methods; this preference persisted also in later competitions (Zeng et al., 2019, 2020), likely due to a general lack of awareness in the scientific community that more accurate methods than "classify and count" exist.

Unfortunately, the data analysis competitions in which OQT and ARC were evaluated (Nakov et al., 2016; Rosenthal et al., 2017) have tested each quantification method only on a single sample of unlabeled items. This evaluation protocol is not adequate for OQ because predictions in quantification correspond to samples of data items, and not to individual data items, as in classification. Measuring a quantifier's performance on a single sample is therefore as unreliable as measuring a classifier's performance on a single data item. As a result, our knowledge of the relative merits of OQT and ARC lacks solidity. We address this issue by introducing experimental protocols for a reliable evaluation of OQ methods. Moreover, we follow these protocols to release two data sets for which OQ has practical relevance.

Even before Forman (2005) discussed quantification as a task of its own, other research fields had already addressed what we now call OQ problems. Most notably, the so-called "unfolding" methods from experimental physics (Blobel, 1985; D'Agostini, 1995) are in fact OQ methods, a finding we detail in Sec. 3.2. Their value for OQ in general has remained unexplored until today, largely due to different terminologies of the fields and despite recent developments on both sides (Aad et al., 2021; Nachman et al., 2020). Here, we bridge this interdisciplinary gap by discussing unfolding methods within the general context of OQ.

## 3 Methods

We use the following notation. By $\mathbf{x} \in \mathcal{X}$ we indicate a data item drawn from a domain $\mathcal{X}$ and by $y \in \mathcal{Y}$ we indicate a class drawn from a set of classes $\mathcal{Y} = \{y_1, ..., y_n\}$, also known as a *codeframe*. Since we deal with *ordinal* quantification, there exists a total order upon the classes, i.e., $y_i < y_{i+1}$. The symbol $\sigma \subseteq \mathcal{X}$ denotes a *sample*, i.e., a non-empty set of unlabeled data items, while $L \subseteq \mathcal{X} \times \mathcal{Y}$ denotes a set of labeled data items. Here, we consider $L$ to be set of hold-out data that has not been employed during the training of the classifier.

By $p_\sigma(y)$ we indicate the true prevalence of class $y$ in sample $\sigma$, where $0 \leq p_\sigma(y) \leq 1$ and $\sum_{y \in \mathcal{Y}} p_\sigma(y) = 1$. By a caret $\hat{p}^M_\sigma(y)$, we indicate an estimate of this prevalence, as obtained by a quantification method $M$ that receives $\sigma$ as an input.

### 3.1 Non-ordinal quantification methods

We start by introducing the most important multi-class quantifiers which do not take ordinality into account. These quantifiers provide the foundation for the ordinal extensions thereof, which we propose in Sec. 3.3.

**3.1.1 Classify and Count (CC).** In the most basic quantification method, a hard classifier $h : \mathcal{X} \rightarrow \mathcal{Y}$ generates predictions for all data items $\mathbf{x} \in \sigma$ and the fraction of predictions is used as a prevalence estimate

$$\hat{p}^{\mathrm{CC}}_{\sigma}(y_i) = \frac{1}{|\sigma|} \cdot \left|\{\mathbf{x} \in \sigma : h(\mathbf{x}) = y_i\}\right|. \tag{1}$$

In the "probabilistic classify and count" (PCC) method, the hard classifier is replaced by a soft classifier $s : \mathcal{X} \rightarrow [0,1]^n$. Here, we assume $\sum_{i=1}^{n} [s(\mathbf{x})]_i = 1$, where $[\cdot]_i$ is the indexing operator.

$$\hat{p}^{\mathrm{PCC}}_{\sigma}(y_i) = \frac{1}{|\sigma|} \cdot \sum_{\mathbf{x} \in \sigma} [s(\mathbf{x})]_i. \tag{2}$$

**3.1.2 Adjusted Classify and Count (ACC).** Since CC and PCC are not appropriate under prior probability shift, the "adjusted classify and count" (Forman, 2005, ACC) and the "probabilistic adjusted classify and count" (Bella et al., 2010, PACC) have been proposed. They adjust $\hat{p}^{\mathrm{CC}}_{\sigma}$ and $\hat{p}^{\mathrm{PCC}}_{\sigma}$, i.e., they correct these estimates in spite of prior probability shift.

In the multi-class setting, we want to estimate a vector of prevalences $\mathbf{p} \in \mathbb{R}^n$, where $\mathbf{p}_i = p_{\sigma}(y_i)$. In this case, the adjustment of ACC and PACC amounts to solving, for $\mathbf{p}$, the system of linear equations

$$\mathbf{q} = \mathbf{M}\mathbf{p}, \tag{3}$$

where $\mathbf{q} \in \mathbb{R}^n$ is a vector of un-adjusted prevalence estimates from CC or PCC, i.e., $\mathbf{q}^{\mathrm{ACC}}_i = \hat{p}^{\mathrm{CC}}_{\sigma}(y_i)$ or $\mathbf{q}^{\mathrm{PACC}}_i = \hat{p}^{\mathrm{PCC}}_{\sigma}(y_i)$. Moreover, $\mathbf{M} \in \mathbb{R}^{n \times n}$ is a matrix that relates the ground truth labels to the predictions of the employed classifier. In the case of ACC, $\mathbf{M}$ is the misclassification matrix of $h$, as estimated from $L$ For PACC, $\mathbf{M}$ is the "soft" misclassification matrix of $s$. Namely,

$$\mathbf{M}^{\mathrm{ACC}}_{ij} = \frac{|\{(\mathbf{x}, y) \in L : h(\mathbf{x}) = y_i,\ y = y_j\}|}{|\{(\mathbf{x}, y) \in L : y = y_j\}|} \tag{4}$$

$$\mathbf{M}^{\mathrm{PACC}}_{ij} = \frac{\sum_{(\mathbf{x}, y) \in L : y = y_j} [s(\mathbf{x})]_i}{|\{(\mathbf{x}, y) \in L : y = y_j\}|} \tag{5}$$

ACC and PACC solve Eq. 3 with the Moore-Penrose pseudo-inverse $\mathbf{M}^{\dagger}$, i.e.

$$\hat{\mathbf{p}} = \mathbf{M}^{\dagger}\mathbf{q}, \tag{6}$$

where $\hat{\mathbf{p}}_i = \hat{p}_{\sigma}(y_i)$ is the estimate of ACC when Eq. 1 and Eq. 4 are employed or the estimate of PACC when Eq. 2 and Eq. 5 are employed.

Unlike the true inverse $\mathbf{M}^{-1}$, the pseudo-inverse always exists. If the true inverse exists, the two matrices are identical; if it does not exist, the solution from Eq. 6 amounts to a minimum-norm least-square estimate of $\mathbf{p}$ (Mueller and Siltanen, 2012, Theorem 4.1).

#### 3.1.3 EM-based Quantification (SLD).

The method by Saerens, Latinne and Decaestecker (2002) follows an expectation maximization approach, which leverages Bayes' theorem in the E-step and updates the prevalence estimates in the M-step. Both of these steps can be combined in a single update rule

$$\hat{p}_\sigma^{(k)}(y_i) = \frac{1}{|\sigma|} \sum_{\mathbf{x} \in \sigma} \frac{\frac{\hat{p}_\sigma^{(k-1)}(y_i)}{\hat{p}_\sigma^{(0)}(y_i)} \cdot [s(\mathbf{x})]_i}{\sum_{j=1}^{n} \frac{\hat{p}_\sigma^{(k-1)}(y_j)}{\hat{p}_\sigma^{(0)}(y_j)} \cdot [s(\mathbf{x})]_j}, \tag{7}$$

where $p_\sigma^{(0)}(y)$ is initialized with the class prevalence values of the training set. Ideally , the soft classifier $s : \mathcal{X} \to [0,1]^n$ approximates posterior probabilities, i.e., $[s(\mathbf{x})]_i \approx \Pr(y_i \mid \mathbf{x})$. SLD continues to apply the update rule from Eq. 7 until the estimates converge.

### 3.2 Existing OQ methods from the physics literature

Similar to the adjustment of ACC, experimental physicists have proposed adjustments that solve the system of linear equations from Eq. 3 for $\mathbf{p}$. However, these "unfolding" quantifiers differ from ACC in two regards.

First, the hard classifier $h$ from Eq. 1 and Eq. 4 is often (although not always) replaced by a partition $c : \mathcal{X} \to \{1, \dots, d\}$ of the feature space, so that

$$\begin{aligned} \mathbf{q}_i &= \frac{1}{|\sigma|} \cdot \left| \{ \mathbf{x} \in \sigma : c(\mathbf{x}) = i \} \right|, \\ \mathbf{M}_{ij} &= \frac{|\{(\mathbf{x}, y) \in L : c(\mathbf{x}) = i,\ y = y_j\}|}{|\{(\mathbf{x}, y) \in L : y = y_j\}|}. \end{aligned} \tag{8}$$

and $\mathbf{M} \in \mathbb{R}^{d \times n}$. Note that by choosing $c = h$, we obtain exactly Eq. 1 and Eq. 4. Another proven choice for $c$ is to partition the feature space by the means of a decision tree; in this case, $d > n$ and $c(\mathbf{x})$ represents the index of a leaf node (Börner et al., 2017).

The second difference between ACC and physics-spawned quantifiers is the aspect of regularization. In being designed for OQ tasks, quantifiers from physics regularize their estimates in order to promote solutions that are the most plausible solutions in OQ. Specifically, these methods employ the assumption that neighbouring classes are similar in terms of their prevalences. Depending on the algorithm, this assumption is leveraged in different ways.

**3.2.1 Regularized Unfolding (RUN).** The early RUN method by Blobel (1985, 2002) is used by physicists for decades, until now (Aartsen et al., 2017; Nöthe et al., 2018). It estimates the vector $\mathbf{p}$ of class prevalences by minimizing a loss function $\mathcal{L} : \mathbb{R}^n \to \mathbb{R}$ over the estimate $\hat{\mathbf{p}}$. This loss function consists of two terms, i.e., a negative log-likelihood term to model the error of $\hat{\mathbf{p}}$ and a regularization term to model the plausibility of $\hat{\mathbf{p}}$.

The likelihood term in $\mathcal{L}$ builds on a Poisson assumption about the distribution of the data. Namely, this term models the counts $\bar{\mathbf{q}}_i = |\sigma| \cdot \mathbf{q}_i$, which are observed in the sample $\sigma$, as being Poisson-distributed with the rates $\lambda_i = \mathbf{M}_i^\top \bar{\mathbf{p}}$. Here, $\mathbf{M}_i$ is the $i$-th column vector of $\mathbf{M}$ and $\bar{\mathbf{p}}_i = |\sigma| \cdot \hat{\mathbf{p}}_i$ are the class counts that would be observed under a prevalence estimate $\hat{\mathbf{p}}$.

The second term of $\mathcal{L}$ is a Tikhonov regularization term $\frac{1}{2}\left(\mathbf{C}\mathbf{p}\right)^2$. This term introduces an inductive bias towards solutions which are plausible with respect to ordinality. The Tikhonov matrix $\mathbf{C}$ is chosen such that differences between neighbouring prevalence estimates are penalized, i.e., such that

$$\frac{1}{2}\left(\mathbf{C}\mathbf{p}\right)^2 = \frac{1}{2}\sum_{i=2}^{n-1}\left(-\mathbf{p}_{i-1} + 2\mathbf{p}_i - \mathbf{p}_{i+1}\right)^2 \tag{9}$$

Combining the likelihood term and the regularization term, the loss function of RUN is given by

$$\mathcal{L}(\hat{\mathbf{p}};\ \mathbf{M}, \mathbf{q}, \tau, \mathbf{C}) = \sum_{i=1}^{d}\left(\mathbf{M}_i^\top \bar{\mathbf{p}} - \bar{\mathbf{q}}_i \cdot \ln(\mathbf{M}_i^\top \bar{\mathbf{p}})\right) + \frac{\tau}{2}\left(\mathbf{C}\hat{\mathbf{p}}\right)^2 \tag{10}$$

and an estimate $\hat{\mathbf{p}}$ is chosen by minimizing $\mathcal{L}$ numerically over $\hat{\mathbf{p}}$. Here, $\tau \geq 0$ is a hyperparameter which controls the impact of the regularization.

**3.2.2 Iterative Bayesian Unfolding (IBU).** The IBU method, proposed by D'Agostini (1995, 2010) and still popular today (Aad et al., 2021; Nachman et al., 2020), revolves around an expectation maximisation approach with Bayes' theorem. It therefore shares a common foundation with the SLD method. The E-step and the M-step of IBU can be written as a single, combined update rule

$$\hat{p}_\sigma^{(k)}(y_i) = \sum_{j=1}^{d} \frac{\mathbf{M}_{ij} \cdot \hat{p}_\sigma^{(k-1)}(y_i)}{\sum_{l=1}^{n} \mathbf{M}_{lj} \cdot \hat{p}_\sigma^{(k-1)}(y_l)}\, \mathbf{q}_i. \tag{11}$$

One difference between IBU and SLD is that $\mathbf{q}$ and $\mathbf{M}$ are defined via counts of hard assignments to partitions $c(\mathbf{x})$, see Eq. 8, while SLD is defined over individual soft predictions $s(\mathbf{x})$, see Eq. 7.

Another difference between IBU and SLD is regularization. In order to promote solutions which are plausible in ordinal quantification, IBU smooths each intermediate estimate $\hat{p}_\sigma^{(k)}(y)$ by fitting a low-order polynomial to $\hat{p}_\sigma^{(k)}(y)$. A linear interpolation between $\hat{p}_\sigma^{(k)}(y)$ and this polynomial is then used as the prior of the next iteration, to reduce the differences between neighbouring prevalence estimates. The interpolation factor is a hyperparameter of IBU through which the degree of regularization is controlled.

**3.2.3 Other methods from the physics literature.** RUN and IBU are two examples for a collection of algorithms that goes under the name of "unfolding". We focus on these two methods due to their long-standing popularity within physics research. In fact, they are among the first methods that have been proposed in this field and they are still widely adopted today, in astroparticle physics (Aartsen et al., 2017; Nöthe et al., 2018), high-energy physics (Aad et al., 2021), and more recently in quantum computing (Nachman et al., 2020). Moreover, RUN and IBU already cover the most important aspects of unfolding methods with respect to ordinal quantification.

Several other unfolding methods share similarities with RUN. For instance, the method by Hoecker and Kartvelishvili (1996) employs the same regularization as RUN, but assumes different Poisson rates, which are simplifications of the rates that RUN uses. In preliminary experiments, here omitted for the sake of conciseness, we have found this simplification to typically deliver less accurate results than RUN. Two other methods, by Schmelling (1994) and by Schmitt (2012), employ the same simplification as Hoecker and Kartvelishvili (1996), but regularize differently. To this end, Schmelling (1994) regularizes with respect to the deviation from a prior, instead of regularizing with respect to ordinal plausibility; therefore, we do not perceive this method to be a true OQ method. Schmitt (2012) adds a second term to the RUN regularization, which enforces prevalence estimates that sum up to one. We use a RUN implementation which already resolves this issue through a positivity constraint and normalization.

Another line of work evolves around the algorithm by Ruhe et al. (2013) and its extensions (Bunse et al., 2018). We perceive this algorithm to lie out of the scope of OQ because it does not address the order of classes, like the other methods from the physics literature do. Moreover, the algorithm was shown to exhibit a performance that is comparable to RUN and IBU, but not better (Bunse et al., 2018).

### 3.3 New ordinal variants of ACC, PACC, and SLD

RUN, IBU, and other OQ methods from the physics literature address ordinality through regularization. Each of their regularization techniques prevents implausible estimates of class prevalence values, i.e., each technique prevents estimates in which the prevalences of neighbouring classes deviate too much from each other. The strength of the regularization is controlled via hyperparameters, which can be tuned to the type of problem at hand. Well-known categorical methods from the quantification literature, such as ACC, PACC, and SLD, do not employ any regularization of this kind. Therefore, they are not ideal choices for OQ tasks.

In the following, we develop algorithms which extend ACC, PACC, and SLD with the regularizers from RUN and IBU. Through this extension, we obtain o-ACC, o-PACC, and o-SLD, the OQ counterparts of these well-known categorical quantification algorithms. Since we only employ the regularizers, but not any other aspect of RUN and IBU, we preserve the general characteristics of ACC, PACC, and SLD. In particular, our methods continue to work with classifier predictions, i.e., we do not employ the categorical feature representation from Eq. 8, which RUN and IBU employ. We also do not use the Poisson assumption of RUN. Therefore, our extensions are "minimal" in the sense that they directly address ordinality, without introducing any undesired side effects.

**3.3.1 o-ACC and o-PACC.** Our ordinal extensions to ACC and PACC build on the finding by Mueller and Siltanen (2012, Theorem 4.1), which states that the solution from Eq. 6 corresponds to a minimum-norm least-squares solution. Namely, among all least-squares solutions $\hat{\mathbf{p}}^{\mathrm{LSq}} = \arg\min_{\mathbf{p}} \|\mathbf{q} - \mathbf{M}\mathbf{p}\|_2^2$, which by themselves do not need to be unique, Eq. 6 is the unique solution that also minimizes the quadratic norm $\|\mathbf{p}\|_2^2$. Therefore, Eq. 6 is conceptually similar, although not necessarily equal, to a regularized estimate

$$\hat{\mathbf{p}}' = \arg\min_{\mathbf{p}} \|\mathbf{q} - \mathbf{M}\mathbf{p}\|_2^2 + \frac{\tau}{2} \|\mathbf{p}\|_2^2 \qquad (12)$$

which employs the quadratic norm for regularization. In particular, both Eq. 6 and Eq. 12 simultaneously minimize a least-squares objective and the norm of their solution candidates. Note that the regularization function herein is, unlike the regularization from RUN, unrelated to the ordinal nature of the classes.

To obtain the true OQ methods o-ACC and o-PACC, we replace the minimum-norm regularization in Eq. 12 with the regularization term of RUN, see Eq. 9. Through this replacement, we minimize the same objective function as ACC and PACC, i.e., a least-squares objective, but regularize towards solutions that we deem more plausible for OQ. The prevalence estimate is

$$\hat{\mathbf{p}}^{\mathrm{o}} = \arg\min_{\mathbf{p}} \|\mathbf{q} - \mathbf{M}\mathbf{p}\|_2^2 + \frac{\tau}{2} \left(\mathbf{C}\mathbf{p}\right)^2 , \qquad (13)$$

the minimizer of which is found through numerical optimization, e.g. through the BFGS optimization technique (Nocedal and Wright, 2006). The o-ACC variant emerges from plugging in Eq. 1 and Eq. 4 for $\mathbf{q}$ and $\mathbf{M}$, while the o-PACC variant emerges from plugging in Eq. 2 and Eq. 5.

**3.3.2 o-SLD.** Our ordinal variant o-SLD leverages the ordinal regularization of IBU in SLD. Namely, our method does not use the latest estimate directly as the prior of the next iteration, but a smoothed version of this estimate. To this end, we fit a low-order polynomial to each intermediate estimate $\hat{p}^{(k)}_{\sigma}(y_i)$ and use a linear interpolation between this polynomial and $\hat{p}^{(k)}_{\sigma}(y_i)$ as the prior of the next iteration. Like in IBU, we consider the interpolation factor as a hyperparameter through which the strength of this regularization is controlled.

## 4 Experiments

The goal of our experiments is to uncover the relative merits of OQ methods that come from different fields. We pursue this goal through a thorough comparison of these methods, on representative OQ data sets.

### 4.1 Evaluation measures

The main evaluation measure we use in this paper is the *Normalized Match Distance* (NMD), defined by Sakai (2018) as

$$\mathrm{NMD}(p, \hat{p}) = \frac{1}{n-1} \mathrm{MD}(p, \hat{p}) \tag{14}$$

where $\frac{1}{n-1}$ is just a normalisation factor that allows NMD to range between 0 (best) and 1 (worst). Here, MD is the *Match Distance* by Werman et al. (1985), which is defined as

$$\mathrm{MD}(p, \hat{p}) = \sum_{i=1}^{n-1} d(y_i, y_{i+1}) \cdot |\hat{P}(y_i) - P(y_i)| \tag{15}$$

where $d(y_i, y_{i+1})$ is the "distance" between consecutive classes $y_i$ and $y_{i+1}$, i.e., the cost we incur in assigning to $y_i$ a probability mass that we should instead assign to $y_{i+1}$, or vice versa; here, we assume $d(y_i, y_{i+1}) = 1$. Moreover, $P(y_i) = \sum_{j=1}^{i} p(y_j)$ is the cumulative distribution of $p$.

MD is a special case of the *Earth Mover's Distance* (EMD) by Rubner et al. (1998), which is a widely acknowledged measure for OQ evaluation (Bunse et al., 2018; Da San Martino et al., 2016; Esuli and Sebastiani, 2010; Nakov et al., 2016; Rosenthal et al., 2017). Since MD and EMD coincide in all of these works, we could as well speak of evaluating OQ methods in terms of EMD, normalized by the constant factor $\frac{1}{n-1}$ from Eq. 14.

Another proposal for measuring the quality of OQ estimates is the *Root Normalised Order-aware Divergence* (RNOD) by Sakai (2018). We include an evaluation in terms of RNOD in the supplementary material, finding that RNOD and NMD consistently lead to the same conclusions.

To obtain an overall score for a quantifier on a data set, we apply this quantifier to each sample $\sigma$. The resulting prevalence estimates are then compared to the ground-truth prevalences, which yields one NMD (or RNOD) value for each sample. The final score of the quantifier is the average of these values, i.e., the average NMD (or RNOD) across all samples of the data set. We test for statistically significant differences between quantification methods in terms of a paired Wilcoxon signed-rank test. Loosely speaking, this test tells us whether one method consistently wins over the other.

### 4.2 Datasets and preprocessing

We conduct our experiments on two large datasets that we have generated for the purpose of this work, and that we make available to the scientific community[2]. The first dataset, named AMAZON-OQ-BK, consists of product reviews labelled according to customer's judgments of quality, i.e., 1Star to 5Stars. The

[2] A public link will be provided in the camera-ready version; for now, our supplementary material includes scripts to extract the data from public sources.

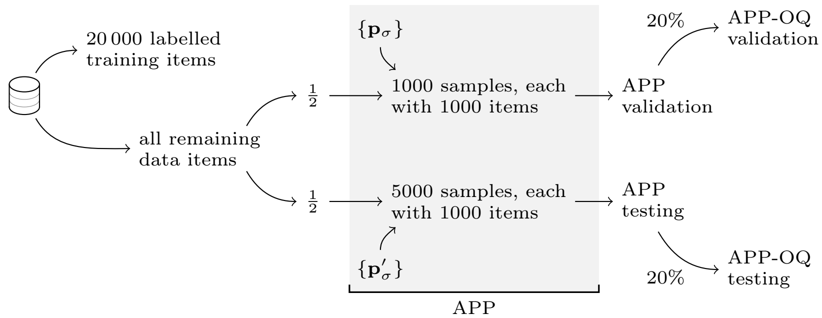


Fig. 1: Sampling of training data, validation data, and testing data through the artificial prevalence protocol (APP). For each sample, a random prevalence vector $\mathbf{p}_\sigma$ or $\mathbf{p}'_\sigma$ is drawn uniformly from the unit simplex and data items are drawn according to this vector. For the Amazon data, a data item corresponds to a single product review. For the telescope data, a data item corresponds to a single telescope recording.

second dataset, Fact-OQ, consists of telescope observations labelled by one of 12 totally ordered classes. Hence, these data sets originate in practically relevant and diverse applications of OQ. From each of these data sets, we subsample a training set, multiple validation samples, and multiple test samples according to two protocols that are well suited for OQ in particular.

**4.2.1 The data sampling protocol.** We start by dividing a set of labelled data items into a training set $L$, a pool of validation items, and a pool of test items, see Fig. 1 . All of these sets are disjoint from each other and each of them is obtained through stratified sampling. From each of the pools, we separately extract samples for quantification.

The extraction of samples follows the *Artificial Prevalence Protocol* (APP), which is by now a standard protocol in quantifier evaluation. This protocol generates each sample in two steps. First, APP generates a random vector $\mathbf{p}_\sigma$ of class prevalence values. This random vector is drawn uniformly at random, from the set of all legitimate prevalence vectors. Namely, we follow Esuli et al. (2022) in using the Kraemer algorithm (Smith and Tromble, 2004), which ensures that all prevalences in the unit $(n-1)$ simplex are picked with equal probability. The second step of APP is to draw from the pool of data, be it our validation pool or our test pool, a subset of a fixed size which realizes the pre-determined class prevalence values of the current sample. The result is a set of samples, each consisting of a set of items with ground-truth prevalence values that are uniformly distributed. We obtain one set of samples from the validation pool and another set of samples from the test pool.

In our experiments, we set size of each sample to 1000, i.e., each sample consists of 1000 data items which realize a random class prevalence vector. The validation set consists of 1000 such samples, the test set of 5000 samples. We set the size of the training set to 20 000.

All items in the pool are replaced after the generation of each sample, so that no sample contains duplicate items but samples from the same pool are not necessarily disjoint. Note, however, that our initial split into a training set, a validation pool and a test pool ensures that each validation sample is disjoint from each test sample and that the training set is disjoint from all other samples.

**4.2.2 Partitioning of samples in terms of their plausibility.** The APP samples all prevalence vectors with the same probability, disregarding of whether these vectors are plausible in the sense of being likely to appear in the practice of OQ. We counteract this shortcoming with *APP-OQ*, a second protocol which is very similar to APP but limited to those samples that that we deem to be the most plausible in the context of OQ. Namely, we select the seemingly most plausible 20% of the previously generated APP samples. We always report the results of APP and APP-OQ side by side, to draw conclusions about the OQ-related merits of the different quantification methods.

We use "smoothness" as a proxy for plausibility. We measure smoothness by invoking Eq. 9 on the true prevalence vector of each sample. In APP-OQ, the hyperparameter optimization is performed on the selected 20% validation samples and the evaluation is performed on the selected 20% test samples.

**4.2.3 The Amazon-OQ-BK dataset.** The first dataset we extract, called Amazon-OQ-BK, is a subset of an existing dataset[3] of 233.1M English-language Amazon product reviews, spanning the period from May 1996 to October 2018, made available by McAuley et al. (2015) . As the labels of the reviews, we use their "stars" scores, and our codeframe is thus $\mathcal{Y}$ ={1Star, 2Stars, 3Stars, 4Stars, 5Stars}, which represents a sentiment quantification task.

We restrict our attention to reviews from the Books domain. We then remove (a) all reviews shorter than 200 characters (since recognising sentiment from shorter reviews may be nearly impossible in some cases), and (b) all reviews that have not been recognized as "useful" by any users (since many reviews never recognised as "useful" may contain comments, say, on Amazon's speed of delivery, and not on the product itself).

We convert the textual representation of the documents into a vector form by using the RoBERTa transformer (Liu et al., 2019) from the Hugging Face hub.[4] To this aim, we fine-tune RoBERTa via prompt learning for a maximum of 5 epochs on our training data, thus taking the model parameters from the epoch which yields the smallest validation loss as monitored on 1000 held-out documents randomly sampled from the training set in a stratified way. For training, we set the learning rate to $2e^{-5}$, the weight decay to 0.01, and the batch size to 16, leaving the other hyperparameters at their default values. For each document,

[3] `http://jmcauley.ucsd.edu/data/amazon/links.html`
[4] `https://huggingface.co/docs/transformers/model_doc/roberta`

we generate features by first applying a forward pass over the fine-tuned network and then averaging the embeddings produced for the special token [CLS] across all the 12 layers of RoBERTa. In our initial experiments, this approach yielded slightly better results than using the [CLS] embedding of the last layer alone. The embedding size of RoBERTa, and hence the number of dimensions of our vectors, amounts to 768.

We make the AMAZON-OQ-BK dataset publicly available,[2] both in its raw textual form and in its processed vector form.

#### 4.2.4 The telescope dataset.

We further evaluate all methods on the open dataset[5] of the FACT telescope (Anderhub et al., 2013). For data of this kind, the physics-spawned OQ methods RUN and IBU are conventional choices among astro-particle physicists (Aartsen et al., 2017; Nöthe et al., 2018). We represent this data in terms of the 20 dense features that are extracted by the standard processing pipeline[6] of the telescope. Each of the 1,851,297 recordings is labelled with the energy of the corresponding particle and our goal is to estimate the distribution of these energy labels through quantification.

While the energy labels are originally continuous, astro-particle physicists have established a common practice of dividing the range of energy values into ordinal classes, as argued in Sec. 3.2. Based on discussions with astro-particle physicists, we divide the range of continuous energy values into 12 ordinal classes.

In order to fit and evaluate quantification methods, we employ simulated telescope data in our experiments. Using simulated data for this purpose is common practice among astro-particle physicists (Aartsen et al., 2017; Nöthe et al., 2018). Indeed, the simulation comprises all aspects of the telescope, from particle interactions inside the atmosphere, over light propagation, up to electrical artefacts inside the telescope camera, so that the simulated data is representative of the real telescope.

### 4.3 Results with ordinal classifiers

In our first experiment, we investigate whether ordinal quantification is solved by non-ordinal quantifiers that embed ordinal classifiers. To this end, we compare a standard multi-class logistic regression (LR) to several ordinal variants of LR. In general, we have found that LR models, trained on the deep RoBERTa embedding of the AMAZON-OQ-BK data set, are extremely powerful models in terms of quantification performance. Therefore, approaching OQ with ordinal LR variants, which are embedded in non-ordinal quantifiers, could be a straightforward solution that is worth the investigation.

The ordinal LR variants we try are the "All Threshold" variant (OLR-AT) and the "Immediate-Threshold variant" (OLR-IT) by Rennie and Srebro (2005). In addition, we try two classifiers which are based on discretising the outputs that are generated by regression models. These methods include an ordinal classifier

[5] https://factdata.app.tu-dortmund.de/

[6] https://github.com/fact-project/open_crab_sample_analysis/

that is based on Ridge Regression (ORidge) and one that is based on linear support vector machines, named Least Absolute Deviation (LAD).

Table 1: Performance of classifiers in terms of the average NMD (lower is better) in the Amazon-OQ-BK dataset. Boldface indicates the best classifier variant for each quantification method, or a variant that is not significantly different from the best one in terms of a paired Wilcoxon signed-rank test at a confidence level of $p = 0.01$. For LR we present standard deviations, while for all other classifiers we show the average deterioration in NMD with respect to LR. PCC, PACC, and SLD require a soft classifier, so that ORidge and LAD cannot be embedded in these methods.

| | CC | PCC | ACC | PACC | SLD |
|---|---|---|---|---|---|
| LR | **.0526** ±.0190 | **.0629** ±.0215 | .0247 ±.0096 | **.0206** ±.0080 | **.0174** ±.0068 |
| OLR-AT | .0527 (+0.2%) | .0657 (+4.4%) | **.0237** (−4.4%) | .0219 (+6.5%) | .0210 (+20.5%) |
| OLR-IT | **.0526** (+0.0%) | .0695 (+10.4%) | .0256 (+3.6%) | .0215 (+4.5%) | .0648 (+271.8%) |
| ORidge | .0550 (+4.5%) | — | .0244 (−1.6%) | — | — |
| LAD | **.0527** (+0.3%) | — | **.0240** (−3.1%) | — | — |

Tab. 1 reports the results we obtain from this experiment, using several well-known non-ordinal quantifiers. These results reveal that, in order to deliver accurate estimates of class prevalence values in the ordinal case, it is not sufficient to equip a multi-class quantifier with an ordinal classifier of this kind. Moreover, the results of SLD, PCC, and PACC suggests that the quality of the posterior probabilities suffers from the adoption of ordinal classifiers. We thus conclude that ordinality in quantification has to involve the quantification level.

### 4.4 Results of the quantifier comparison

In our main experiment, we compare our proposaled methods o-ACC, o-PACC, and o-SLD with several baselines. First, we consider the existing OQ methods OQT (Da San Martino et al., 2016) and ARC (Esuli, 2016), which we further detail in the supplementary material. Second, we consider the "unfolding" OQ methods IBU and RUN from Sec. 3.2. Third, we consider the well-known non-ordinal methods CC, PCC, ACC, PACC, and SLD. We compare these methods on both data sets and with both protocols, as introduced in Sec. 4.2.

Each of the methods is allowed to tune the hyperparameters of its embedded classifier using the samples of the validation set. To this end, the Amazon-OQ-BK data is always predicted with logistic regression models and the Fact-OQ data is always predicted with probability-calibrated decision trees. This choice of classifiers is motivated by common practice in the fields where these data sets come from and from our own experience that these classifiers work well on the data. After the hyperparameters of the classifier are chosen, we apply each method to the samples of the test set.

Table 2: Average performance in terms of NMD (lower is better). For each data set (Amazon-OQ-BK and FACT-OQ), we present the results of the two protocols APP and APP-OQ. The best performance in each column is highlighted in boldface. According to a Wilcoxon signed rank test with $p = 0.01$, all other methods are significantly different from the best method.

| method | Amazon-OQ-BK | | Fact-OQ | |
|---|---|---|---|---|
| | APP | APP-OQ | APP | APP-OQ |
| CC | .0526 ± .019 | .0344 ± .013 | .0534 ± .012 | .0494 ± .011 |
| PCC | .0629 ± .022 | .0440 ± .017 | .0651 ± .017 | .0621 ± .017 |
| ACC | .0229 ± .009 | .0193 ± .007 | .0582 ± .028 | .0575 ± .028 |
| PACC | .0209 ± .008 | .0176 ± .007 | .0791 ± .048 | .0816 ± .049 |
| SLD | **.0172 ± .007** | .0154 ± .006 | .0373 ± .010 | .0355 ± .009 |
| OQT | .0775 ± .026 | .0587 ± .027 | .0746 ± .019 | .0731 ± .020 |
| ARC | .0641 ± .023 | .0477 ± .015 | .0566 ± .014 | .0568 ± .016 |
| IBU | .0253 ± .010 | .0197 ± .007 | **.0213 ± .005** | .0187 ± .004 |
| RUN | .0252 ± .010 | .0198 ± .007 | .0222 ± .006 | .0194 ± .005 |
| o-ACC | .0229 ± .009 | .0188 ± .007 | .0274 ± .007 | .0230 ± .006 |
| o-PACC | .0209 ± .008 | .0174 ± .007 | .0230 ± .006 | **.0178 ± .004** |
| o-SLD | .0173 ± .007 | **.0152 ± .006** | .0327 ± .008 | .0289 ± .007 |

The results of this experiment, in terms of NMD, are summarized in Tab. 2. We see that our proposals win on both data sets, if the ordinal APP-OQ protocol is employed. More specifically, o-SLD is the best method on the Amazon-OQ-BK data set and o-PACC is the best method on the Fact-OQ data set. Moreover, o-SLD is consistently better or equal to SLD, o-ACC is consistently better or equal to ACC, and o-PACC is consistently better or equal to PACC, also in the standard APP protocol in which smoothness is not imposed.

Additional experiments we have carried out, including further datasets, RNOD as an alternative evaluation measure, and TFIDF as an alternative vectorial representation for text, confirm the conclusions we draw from Tab. 2. We provide these results in the supplementary material.

## 5 Conclusion

We have proposed two evaluation protocols for ordinal quantification, which we have taken out on two OQ data sets that we have released. We have demonstrated that so-called "unfolding" methods from experimental physics are in fact OQ methods and, as such, are also applicable in other OQ applications. We took inspiration from these methods when we devised o-ACC, o-PACC, and o-SLD, our OQ variants of some well-known non-ordinal quantification methods. Namely, our OQ variants successfully employ the regularization techniques from "unfolding" methods to prevent solutions that are less plausible in OQ.

We have provided empirical evidence that OQ has to be tackled at the quantification level, and is not solved by equipping a non-ordinal quantifier with an ordinal classifier. Evaluating our proposed quantifiers against existing OQ methods from different fields and against non-ordinal baselines, we observe that, despite some non-ordinal quantifiers work reasonably well in OQ scenarios, there is a clear tendency that dedicated OQ methods outperform the non-ordinal quantifiers in OQ tasks.

For future work, we conceive the idea of regularization to be fruitful also for other quantification tasks, e.g. multi-label quantification or quantification with priors. Moreover, we recognize a need for more public OQ data sets.

# Bibliography


Aad, G., Abbott, B., Abbott, D. C., et al. (2021). Measurements of the inclusive and differential production cross sections of a top-quark–antiquark pair in association with a Z boson at $\sqrt{s} = 13$ TeV with the ATLAS detector. *Eur. Phys. J. C*, 81(8).

Aartsen, M. G., Ackermann, M., Adams, J., et al. (2017). Measurement of the $\nu_\mu$ energy spectrum with IceCube-79. *Eur. Phys. J. C*, 77(10).

Anderhub, H., Backes, M., Biland, A., et al. (2013). Design and operation of FACT, the first G-APD Cherenkov telescope. *J. Inst.*, 8(06).

Bella, A., Ferri, C., Hernández-Orallo, J., and Ramírez-Quintana, M. J. (2010). Quantification via probability estimators. In *Int. Conf. Data Mining*.

Blobel, V. (1985). Unfolding methods in high-energy physics experiments. Technical Report DESY-84-118, CERN, Geneva, CH.

Blobel, V. (2002). An unfolding method for high-energy physics experiments. In *Adv. Stat. Techn. Part. Phys.*, pages 258–267, Durham, UK.

Börner, M., Hoinka, T., Meier, M., et al. (2017). Measurement/simulation mismatches and multivariate data discretization in the machine learning era. In *Conf. Astron. Data Anal. Softw. Syst.*, pages 431–434, Santiago, CL.

Bunse, M., Piatkowski, N., Morik, K., Ruhe, T., and Rhode, W. (2018). Unification of deconvolution algorithms for Cherenkov astronomy. In *Int. Conf. Data Science and Adv. Anal.*, pages 21–30.

Da San Martino, G., Gao, W., and Sebastiani, F. (2016). Ordinal text quantification. In *Conf. Research Dev. Inf. Retrieval*, pages 937–940.

D'Agostini, G. (1995). A multidimensional unfolding method based on Bayes' theorem. *Nucl. Instr. Meth. Phys. Research: Sect. A*, 362(2-3):487–498.

D'Agostini, G. (2010). Improved iterative Bayesian unfolding. arXiv:1010.0632.

Esuli, A. (2016). ISTI-CNR at SemEval-2016 Task 4: Quantification on an ordinal scale. In *Int. Workshop Semantic Eval.*, pages 92–95.

Esuli, A., Moreo, A., and Sebastiani, F. (2022). LeQua@CLEF2022: Learning to Quantify. In *Eur. Conf. Inf. Retrieval*. Forthcoming.

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

Forman, G. (2005). Counting positives accurately despite inaccurate classification. In *Eur. Conf. Mach. Learn.*, pages 564–575.

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

Gao, W. and Sebastiani, F. (2016). From classification to quantification in tweet sentiment analysis. *Social Network Analysis and Mining*, 6(19):1–22.

González, P., Castaño, A., Chawla, N. V., and del Coz, J. J. (2017). A review on quantification learning. *ACM Computing Surveys*, 50(5):74:1–74:40.

Higashinaka, R., Funakoshi, K., Inaba, M., Tsunomori, Y., Takahashi, T., and Kaji, N. (2017). Overview of the 3rd Dialogue Breakdown Detection challenge. In *Dialog System Technology Challenge*.

Hoecker, A. and Kartvelishvili, V. (1996). SVD approach to data unfolding. *Nucl. Instr. Meth. Phys. Research: Sect. A*, 372(3):469–481.

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., and Stoyanov, V. (2019). RoBERTa: A robustly optimized BERT pretraining approach. arXiv:1907.11692.

McAuley, J. J., Targett, C., Shi, Q., and van den Hengel, A. (2015). Image-based recommendations on styles and substitutes. In *Int. Conf. Research Dev. Inf. Retrieval*, pages 43–52.

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

Mueller, J. L. and Siltanen, S. (2012). *Linear and nonlinear inverse problems with practical applications*. Society for Industrial and Applied Mathematics.

Nachman, B., Urbanek, M., de Jong, W. A., and Bauer, C. W. (2020). Unfolding quantum computer readout noise. *npj Quant. Inf.*, 6(1).

Nakov, P., Ritter, A., Rosenthal, S., Sebastiani, F., and Stoyanov, V. (2016). SemEval-2016 Task 4: Sentiment analysis in Twitter. In *Int. Workshop Semantic Eval.*, pages 1–18.

Nocedal, J. and Wright, S. J. (2006). *Numerical optimization*. Springer, Cham, CH, 2nd edition.

Nöthe, M., Adam, J., Ahnen, M. L., et al. (2018). FACT – performance of the first Cherenkov telescope observing with SiPMs. In *Int. Cosmic Ray Conf.*

Rennie, J. D. and Srebro, N. (2005). Loss functions for preference levels: Regression with discrete ordered labels. In *IJCAI 2005 Workshop on Adv. in Pref. Handling*.

Rosenthal, S., Farra, N., and Nakov, P. (2017). SemEval-2017 Task 4: Sentiment analysis in Twitter. In *Int. Workshop Semantic Eval.*, pages 502–518.

Rubner, Y., Tomasi, C., and Guibas, L. J. (1998). A metric for distributions with applications to image databases. In *Int. Conf. Comp. Vision*, pages 59–66.

Ruhe, T., Schmitz, M., Voigt, T., and Wornowizki, M. (2013). DSEA: A data mining approach to unfolding. In *Int. Cosmic Ray Conf.*, pages 3354–3357.

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

Sakai, T. (2018). Comparing two binned probability distributions for information access evaluation. In *Int. Conf. Research Dev. Inf. Retrieval*, pages 1073–1076.

Sakai, T. (2021). A closer look at evaluation measures for ordinal quantification. In *CIKM 2021 Workshop on Learning to Quantify*.

Schmelling, M. (1994). The method of reduced cross-entropy: A general approach to unfold probability distributions. *Nucl. Instr. Meth. Phys. Research: Sect. A*, 340(2):400–412.

Schmitt, S. (2012). TUnfold, an algorithm for correcting migration effects in high energy physics. *J. Inst.*, 7(10).

Smith, N. A. and Tromble, R. W. (2004). Sampling uniformly from the unit simplex. Technical report, Johns Hopkins University.

Werman, M., Peleg, S., and Rosenfeld, A. (1985). A distance metric for multi-dimensional histograms. *Comp. Vis., Graph., Image Proc.*, 32:328–336.

Zeng, Z., Kato, S., and Sakai, T. (2019). Overview of the NTCIR-14 Short Text Conversation task: Dialogue Quality and Nugget Detection subtasks. In *Workshop on NII Testbeds and Community for Inf. access Research.*

Zeng, Z., Kato, S., Sakai, T., and Kang, I. (2020). Overview of the NTCIR-15 Dialogue Evaluation task (DialEval-1). In *Workshop on NII Testbeds and Community for Inf. access Research.*

# A Existing OQ methods from quantification literature

For completeness, we introduce the OQ methods by Da San Martino et al. (2016) and by Esuli (2016), which appear in our main experiment from Sec. 4.4. Both of these methods do not address ordinality through regularization, like we suggest, but through binary decompositions of the codeframe.

## A.1 Ordinal Quantification Tree (OQT)

The algorithm by Da San Martino et al. (2016) trains a quantifier by arranging probabilistic binary classifiers (one for each possible bipartition of the ordered set of classes) into an *ordinal quantification tree* (OQT), which is conceptually similar to a hierarchical classifier. Two characteristic aspects of training an OQT are that (a) the loss function used for splitting a node is a quantification loss (and not a classification loss), e.g., the Kullback-Leibler Divergence, and (b) the splitting criterion is informed by the class order. Given a test document, one generates a posterior probability for each of the classes by having the document descend all branches of the trained tree; after this is done for all documents in the test sample, the probabilistic classify-and-count (PCC – (Bella et al., 2010)) multiclass (i.e., non-ordinal) quantification method is invoked in order to compute the final prevalence estimates.

The OQT method was only tested in the SemEval 2016 "Sentiment analysis in Twitter" shared task (Nakov et al., 2016). While OQT was the best performer in that subtask, its true value still has to be assessed, since the above-mentioned subtask evaluated participating algorithms on one test sample only. In Sec. 4 we have tested OQT in a much more robust way.

## A.2 Adjusted Regress and Count (ARC)

The algorithm by Esuli (2016) is similar to OQT in that it trains a hierarchical classifier where the leaves of the tree are the classes, these leaves are ordered left-to-right, and each internal node partitions an ordered sequence of classes in two such subsequences. One difference between the two algorithms is the criterion used in order to decide where to split a given sequence of classes, which for OQT is based on a quantification loss (KLD), and for ARC is based on the principle of minimizing the imbalance (in terms of the number of training examples) of the two subsequences. A second difference is that, once the tree is trained and used to classify the test documents, OQT uses what is basically a PCC algorithm, while ARC uses the adjusted classify-and-count (ACC) multiclass quantification method (Forman, 2008).

Concerning the quality of ARC, the same considerations made for OQT apply, since ARC, like OQT, has only been tested in the Ordinal Quantification subtask of the SemEval 2016 "Sentiment analysis in Twitter" shared task; despite the fact that it worked well in that context, the experiments that we are presenting in Sec. 4 are more conclusive.

# B Extended results

The following results complete the experiments we have shown in the main paper.

## B.1 Performance in terms of RNOD

We have repeated all of our experiments in terms of the *Root Normalised Order-aware Divergence* (RNOD) evaluation measure, instead of NMD, as proposed in (Sakai, 2018) and as defined as

$$\mathrm{RNOD}(p, \hat{p}) = \left( \frac{\sum_{y_i \in \mathcal{Y}^*} \sum_{y_j \in \mathcal{Y}} d(y_j, y_i)(p(y_j) - \hat{p}(y_j))^2}{|\mathcal{Y}^*|(n-1)} \right)^{\frac{1}{2}} \tag{16}$$

where $\mathcal{Y}^* = \{y_i \in \mathcal{Y} | p(y_i) > 0\}$.

From examining the RNOD results from Tab. 3, we may note that, while some methods change positions in the ranking, as compared to their ranks in terms of NMD, general conclusions from the NMD evaluation also hold in terms of RNOD.

Table 3: Average performance in terms of RNOD (lower is better), in analogy to the NMD results from Tab. 2. For each data set (Amazon-OQ-BK and FACT-OQ), we present the results of the two protocols APP and APP-OQ. The best performance in each column is highlighted in boldface. We further highlight all methods which are not significantly different from the best method, as according to a Wilcoxon signed rank test with $p = 0.01$.

| method | Amazon-OQ-BK | | Fact-OQ | |
|---|---|---|---|---|
| | APP | APP-OQ | APP | APP-OQ |
| CC | .1151 ± .048 | .0606 ± .020 | .1319 ± .036 | .1071 ± .027 |
| PCC | .1360 ± .054 | .0758 ± .025 | .1372 ± .034 | .1096 ± .026 |
| ACC | .0487 ± .024 | .0374 ± .016 | .1563 ± .040 | .1375 ± .030 |
| PACC | .0419 ± .019 | .0327 ± .014 | .1750 ± .056 | .1719 ± .047 |
| SLD | **.0363 ± .017** | .0302 ± .014 | .0890 ± .029 | .0767 ± .021 |
| OQT | .1542 ± .064 | .0960 ± .032 | .1456 ± .035 | .1225 ± .032 |
| ARC | .1303 ± .056 | .0770 ± .027 | .1242 ± .032 | .0973 ± .022 |
| IBU | .0534 ± .025 | .0357 ± .014 | **.0822 ± .028** | .0649 ± .018 |
| RUN | .0531 ± .025 | .0361 ± .014 | .0869 ± .029 | .0685 ± .019 |
| o-ACC | .0487 ± .024 | .0353 ± .014 | .1032 ± .033 | .0754 ± .016 |
| o-PACC | .0419 ± .019 | .0316 ± .012 | .0914 ± .029 | **.0625 ± .016** |
| o-SLD | **.0365 ± .017** | **.0296 ± .013** | .0857 ± .027 | .0658 ± .015 |

We do not choose RNOD as the main evaluation function (and prefer NMD for the main paper instead) because we do not think RNOD is a satisfactory measure for OQ. The reason why we do not consider RNOD a satisfactory OQ measure is that, without (we think) reason, it penalises more heavily mistakes (i.e., "transfers" of probability mass from a class to another) closer to the extremes of the codeframe. For instance, given $\mathcal{Y} = \{y_1, y_3, y_3, y_4, y_5\}$, assume $p = (0.2, 0.2, 0.2, 0.2, 0.2)$, and assume two predicted distributions $\hat{p}' = (0.2, 0.2, 0.3, 0.1, 0.2)$ and $\hat{p}'' = (0.2, 0.2, 0.2, 0.3, 0.1)$. The two predicted distributions make essentially the same mistake, i.e., erroneously "transfer" a probability mass of 0.1 from a class $y_i$ to a class $y_{(i-1)}$, the difference being that in $\hat{p}'$ it is the case that $i = 4$ and in $\hat{p}''$ it is the case that $i = 5$. According to our intuitions, $\hat{p}'$ and $\hat{p}''$ should be equally penalised. While NMD indeed penalises them equally (since $\mathrm{NMD}(p, \hat{p}') = \mathrm{NMD}(p, \hat{p}'') = 0.1$), RNOD does not (since $\mathrm{RNOD}(p, \hat{p}') \approx 0.077$ while $\mathrm{RNOD}(p, \hat{p}'') \approx 0.092$). Sakai (2021) has proposed other OQ evaluation measures, such as *Root Symmetric Normalised Order-aware Divergence* (RSNOD) and *Root Normalised Average Distance-Weighted sum of squares* (RNADW), but we do not consider them here since they are variants of RNOD that suffer anyway from the problem mentioned above.

### B.2 Results on other data sets

We have repeated our experiment from Tab. 2 also several other data sets.

First, we employ a different representation of the Amazon-OQ-BK data, namely a TFIDF representation instead of the RoBERTa embeddings we employ in the main paper. The results with this representation, both in terms of NMD and RNOD, are presented in Tab. 4.

Second, we evaluate on a collection of 4 public data sets from the UCI repository and OpenML. To this end, we have first selected regression data sets with at least 30 000 items. From there on, we have tried to find an equidistant binning which produces at least 10 bins (= ordered classes), each of which have at least 1000 items. We only maintain data sets for which such a binning was possible and we remove all items that lie outside the 10 equidistant bins. In order to maintain as many samples as possible, we maximize the distance between the left-most and right-most bin boundaries. If less then 30 000 items remain, we omit the data set. From this protocol, we obtain the 4 data sets Uci-blog-feedback-OQ, Uci-online-news-popularity-OQ, OpenMl-Yolanda-OQ, and OpenMl-fried-OQ. We present the results obtained with these data sets in terms of NMD, see Tab. 5, and in terms of RNOD, see Tab. 6.

### B.3 Hyperparameter grids

In our experiments, each method has the opportunity to optimize its hyperparameters on the APP (or APP-OQ) validation samples. These hyper-parameters consist of parameters of the quantifier and of parameters of the classifier, with which the quantifier is equipped. After taking out preliminary experiments, which we omit here for conciseness, we have chosen different hyperparameter grids for the different data sets.

To this end, Tab. 7 and Tab. 8 present the parameters for the Amazon-OQ-BK data set. For instance, CC and PCC can choose between 10 hyperparameter configurations of the classifier (2 class weights $\times$ 5 regularization parameters), but they do not have additional parameters on the quantification level. We note

Table 4: NMD (left) and RNOD (right) on a TFIDF representation, instead of RoBERTa embeddings, of the Amazon-OQ-BK data set.

| method | Amazon-OQ-BK (TFIDF) APP | Amazon-OQ-BK (TFIDF) APP-OQ |
|---|---|---|
| CC | .0867 ± .034 | .0683 ± .031 |
| PCC | .1082 ± .044 | .0950 ± .048 |
| ACC | .0353 ± .015 | .0333 ± .014 |
| PACC | .0301 ± .015 | .0310 ± .015 |
| SLD | .0477 ± .018 | .0381 ± .012 |
| OQT | .1583 ± .065 | .1539 ± .072 |
| ARC | .0989 ± .037 | .0855 ± .038 |
| IBU | .0596 ± .023 | .0454 ± .020 |
| RUN | .0594 ± .023 | .0452 ± .020 |
| o-ACC | .0347 ± .017 | .0227 ± .009 |
| o-PACC | **.0276 ± .014** | **.0194 ± .007** |
| o-SLD | .0477 ± .018 | .0363 ± .011 |

| method | Amazon-OQ-BK (TFIDF) APP | Amazon-OQ-BK (TFIDF) APP-OQ |
|---|---|---|
| CC | .1555 ± .062 | .0953 ± .033 |
| PCC | .1807 ± .063 | .1244 ± .045 |
| ACC | .0786 ± .039 | .0735 ± .035 |
| PACC | .0681 ± .037 | .0708 ± .037 |
| SLD | .1073 ± .051 | .0814 ± .027 |
| OQT | .2168 ± .071 | .1659 ± .058 |
| ARC | .1698 ± .065 | .1123 ± .035 |
| IBU | .1186 ± .052 | .0678 ± .022 |
| RUN | .1185 ± .053 | .0675 ± .022 |
| o-ACC | .0777 ± .038 | .0465 ± .020 |
| o-PACC | **.0624 ± .034** | **.0399 ± .017** |
| o-SLD | .0973 ± .036 | .0688 ± .017 |

Table 5: NMD in additional datasets

| method | Uci-blog-feedback-OQ | | Uci-online-news-popularity-OQ | | OpenMl-Yolanda-OQ | | OpenMl-fried-OQ | |
|---|---|---|---|---|---|---|---|---|
| | APP | APP-OQ | APP | APP-OQ | APP | APP-OQ | APP | APP-OQ |
| CC | .0958 ± .034 | .0884 ± .031 | .1664 ± .047 | .1549 ± .045 | .0767 ± .023 | .0779 ± .025 | .0330 ± .008 | .0243 ± .006 |
| PCC | .0967 ± .042 | .0960 ± .045 | .0996 ± .044 | .0985 ± .047 | .0926 ± .030 | .0921 ± .032 | .0410 ± .010 | .0330 ± .008 |
| ACC | .1147 ± .042 | .1144 ± .045 | .1365 ± .055 | .1357 ± .060 | .0807 ± .024 | .0824 ± .026 | .0454 ± .021 | .0482 ± .023 |
| PACC | .1323 ± .049 | .1437 ± .050 | .1515 ± .063 | .1246 ± .055 | .1068 ± .047 | .1102 ± .050 | .0614 ± .026 | .0659 ± .026 |
| SLD | .1001 ± .044 | .1224 ± .038 | .1576 ± .063 | .1687 ± .069 | .0753 ± .025 | .0784 ± .028 | .0369 ± .009 | .0373 ± .008 |
| OQT | .2222 ± .058 | .2050 ± .057 | .3220 ± .087 | .3177 ± .092 | .2246 ± .056 | .2223 ± .058 | .0566 ± .014 | .0472 ± .012 |
| ARC | .2420 ± .062 | .2474 ± .063 | .3801 ± .085 | .3793 ± .089 | .2513 ± .058 | .2500 ± .060 | .0589 ± .017 | .0598 ± .018 |
| IBU | .0997 ± .046 | .0980 ± .049 | .0886 ± .039 | .0858 ± .043 | **.0558 ± .017** | .0553 ± .018 | **.0168 ± .005** | **.0146 ± .004** |
| RUN | .1348 ± .052 | .1339 ± .054 | .1115 ± .048 | .1181 ± .053 | .0577 ± .017 | .0604 ± .018 | .0206 ± .006 | .0161 ± .005 |
| o-ACC | .0772 ± .031 | .0728 ± .027 | **.0833 ± .030** | **.0718 ± .027** | .0568 ± .016 | .0549 ± .017 | .0264 ± .008 | .0189 ± .004 |
| o-PACC | **.0747 ± .028** | **.0664 ± .025** | .0954 ± .039 | .0804 ± .031 | .0580 ± .014 | **.0537 ± .014** | .0350 ± .018 | **.0146 ± .004** |
| o-SLD | .1195 ± .041 | .1190 ± .040 | .0993 ± .044 | .0992 ± .046 | .0701 ± .019 | .0648 ± .019 | .0322 ± .007 | .0282 ± .005 |

Table 6: RNOD in additional datasets

| method | Uci-blog-feedback-OQ | | Uci-online-news-popularity-OQ | | OpenMl-Yolanda-OQ | | OpenMl-fried-OQ | |
|---|---|---|---|---|---|---|---|---|
| | APP | APP-OQ | APP | APP-OQ | APP | APP-OQ | APP | APP-OQ |
| CC | .2007 ± .049 | .1715 ± .037 | .2981 ± .060 | .2687 ± .051 | .1605 ± .043 | .1362 ± .038 | .1125 ± .034 | .0727 ± .015 |
| PCC | .1643 ± .042 | .1371 ± .038 | .1661 ± .043 | .1372 ± .038 | .1642 ± .041 | .1368 ± .036 | .1290 ± .037 | .0896 ± .021 |
| ACC | .2748 ± .062 | .2559 ± .057 | .2639 ± .056 | .2534 ± .047 | .1656 ± .045 | .1444 ± .043 | .1336 ± .048 | .1352 ± .044 |
| PACC | .2507 ± .069 | .2512 ± .064 | .3056 ± .075 | .2938 ± .078 | .2228 ± .056 | .2108 ± .040 | .1820 ± .055 | .1558 ± .038 |
| SLD | .2299 ± .050 | .2247 ± .039 | .2704 ± .081 | .2531 ± .040 | .2064 ± .059 | .1824 ± .042 | .1009 ± .031 | .0921 ± .023 |
| OQT | .3354 ± .046 | .3122 ± .043 | .3331 ± .060 | .3056 ± .064 | .2612 ± .049 | .2418 ± .050 | .1621 ± .048 | .1238 ± .035 |
| ARC | .2552 ± .031 | .2468 ± .022 | .3976 ± .053 | .3734 ± .054 | .2342 ± .041 | .2079 ± .037 | .1532 ± .055 | .1346 ± .060 |
| IBU | .1598 ± .046 | .1294 ± .040 | .1573 ± .044 | **.1232 ± .034** | .1438 ± .043 | .1172 ± .039 | **.0623 ± .023** | .0531 ± .017 |
| RUN | .1802 ± .047 | .1482 ± .041 | .1698 ± .043 | .1425 ± .040 | .1487 ± .048 | .1223 ± .038 | .0750 ± .026 | .0565 ± .018 |
| o-ACC | .1567 ± .045 | .1363 ± .030 | .1669 ± .045 | .1335 ± .040 | **.1374 ± .038** | **.1081 ± .027** | .1085 ± .036 | .0755 ± .022 |
| o-PACC | **.1526 ± .042** | **.1229 ± .037** | **.1555 ± .041** | .1356 ± .036 | .1439 ± .037 | **.1074 ± .023** | .1146 ± .050 | **.0510 ± .014** |
| o-SLD | .1720 ± .045 | .1502 ± .040 | .1706 ± .045 | .1394 ± .039 | .1542 ± .041 | .1193 ± .029 | .1019 ± .035 | .0730 ± .016 |

that an inspection of the validation results revealed that the fraction of hold-out data does not considerably affect the results of ACC, PACC, OQT, and ARC. Therefore, we save computational resources by omitting some values of this parameter in the final hyperparameter grid.

Tab. 9 and Tab. 10 present the parameters for the FACT-OQ data. For conciseness, they also contain the parameters for the UCI and OpenML data sets. The remaining parameters for the UCI and OpenML data sets are presented in Tab. 11

Table 7: Hyperparameter grid of classifiers when analyzing the AMAZON-OQ-BK data in the experiment from Tab. 2.

| classifier | parameter | values |
|---|---|---|
| logistic regression | class weight | {balanced, unbalanced } |
| | regularization parameter $C$ | $\{0.001, 0.01, 0.1, 1.0, 10.0\}$ |

Table 8: Hyperparameter grid of quantification methods when analyzing the AMAZON-OQ-BK data in the experiment from Tab. 2.

| method | parameter | values |
|---|---|---|
| CC | no parameters | |
| PCC | no parameters | |
| ACC | fraction of hold-out data | $\{\frac{1}{4}, \frac{1}{3}, \frac{1}{2}\}$ |
| PACC | fraction of hold-out data | $\{\frac{1}{4}, \frac{1}{3}, \frac{1}{2}\}$ |
| SLD | no parameters | |
| OQT | fraction of hold-out data | $\{\frac{1}{3}\}$ |
| ARC | fraction of hold-out data | $\{\frac{1}{3}\}$ |
| RUN | $\tau$ | {3e-2, 1e-2, 3e-3, 1e-3, 3e-4, 1e-4, 3e-5, 1e-6} |
| IBU | order of polynomial | $\{0, 1, 2\}$ |
| | interpolation factor | {3e-1, 1e-1, 3e-2, 1e-2, 3e-3, 1e-3} |
| o-ACC | fraction of hold-out data | $\{\frac{1}{4}, \frac{1}{3}\}$ |
| | $\tau$ | {1e-2, 3e-3, 1e-3, 3e-4, 1e-4, 1e-5, 1e-6, 1e-9} |
| o-PACC | fraction of hold-out data | $\{\frac{1}{4}, \frac{1}{3}\}$ |
| | $\tau$ | {1e-2, 3e-3, 1e-3, 3e-4, 1e-4, 1e-5, 1e-6, 1e-9} |
| o-SLD | order of polynomial | $\{0, 1, 2\}$ |
| | interpolation factor | {1e-1, 3e-2, 1e-2, 3e-3, 1e-3} |

Table 9: Hyperparameter grid of classifiers when analyzing the Fact-OQ data in the experiment from Tab. 2.

| classifier | parameter | values |
|---|---|---|
| probability-calibrated decision tree | class weight | {balanced, unbalanced} |
| | split criterion | {Gini index, Entropy} |
| | maximum depth | $\{4, 6, 8, 10, 12\}$ |

Table 10: Hyperparameter grid of quantification methods when analyzing the Fact-OQ data in the experiment from Tab. 2 or any of the data sets Uci-blog-feedback-OQ, Uci-online-news-popularity-OQ, OpenMl-Yolanda-OQ, and OpenMl-fried-OQ.

| method | parameter | values |
|---|---|---|
| CC | no parameters | |
| PCC | no parameters | |
| ACC | fraction of hold-out data | $\{\frac{1}{4}, \frac{1}{3}, \frac{1}{2}\}$ |
| PACC | fraction of hold-out data | $\{\frac{1}{4}, \frac{1}{3}, \frac{1}{2}\}$ |
| SLD | no parameters | |
| OQT | fraction of hold-out data | $\{\frac{1}{3}\}$ |
| ARC | fraction of hold-out data | $\{\frac{1}{3}\}$ |
| RUN | $\tau$ | {1e-1, 1e-3, 1e-5} |
| | number of leaf nodes | $\{60, 120, 180\}$ |
| IBU | order of polynomial | $\{0, 1, 2\}$ |
| | interpolation factor | {0.1, 0.01, 0.0} |
| | number of leaf nodes | $\{60, 120, 180\}$ |
| o-ACC | fraction of hold-out data | $\{\frac{1}{3}\}$ |
| | $\tau$ | {1e-1, 1e-3, 1e-5} |
| o-PACC | fraction of hold-out data | $\{\frac{1}{3}\}$ |
| | $\tau$ | {1e-1, 1e-3, 1e-5} |
| o-SLD | order of polynomial | $\{0, 1, 2\}$ |
| | interpolation factor | {1e-1, 3e-2, 1e-2} |

Table 11: Hyperparameter grid of classifiers when analyzing any of the data sets Uci-blog-feedback-OQ, Uci-online-news-popularity-OQ, OpenMl-Yolanda-OQ, and OpenMl-fried-OQ.

| classifier | parameter | values |
|---|---|---|
| probability-calibrated decision tree | class weight | {balanced, unbalanced} |
| | split criterion | {Gini index, Entropy} |
| | maximum depth | $\{4, 6, 8, 10, 12\}$ |
| logistic regression | class weight | {balanced, unbalanced} |
| | regularization parameter $C$ | $\{0.001, 0.01, 0.1, 1.0, 10.0\}$ |

### B.4 Performance in other APP plausibility levels

Our APP-OQ protocol selects the 20% of validation and test samples which we deem most plausible. For completeness, we include here the results for other plausibility levels, which are the second-most, the third-most, the fourth-most, and the least plausible 20%. In other words: we have divided all APP samples in terms of their conceived plausibility into five levels, the first of which makes our APP-OQ, and we have evaluated all methods in all of these plausibility levels.

As another matter of making our results transparent, we present these tables in a different way, which also includes the hyperparameters that each method has chosen on the validation samples. Since we also include the regular APP in this mode of presentation, we have 6 tables per data set, i.e., regular APP and five plausibility levels. These tables only consider NMD, but the LaTeX sources of the RNOD tables are part of our supplementary material.

Table 12: NMD on Amazon-OQ-BK, regular APP

| quantification method | avg. NMD ± stddev. |
|---|---|
| SLD on LR ($w = n, C = 0.01$) | **$0.0172 \pm 0.0067$** |
| o-SLD ($o = 0, i = 0.001$) on LR ($w = n, C = 0.01$) | $0.0173 \pm 0.0067$ |
| PACC ($v = \frac{1}{4}$) on LR ($w = u, C = 0.1$) | $0.0209 \pm 0.0083$ |
| o-PACC ($r = I, \tau = 1.0e-9, v = \frac{1}{4}$) on LR ($w = u, C = 0.1$) | $0.0209 \pm 0.0083$ |
| ACC ($v = \frac{1}{3}$) on LR ($w = u, C = 10.0$) | $0.0229 \pm 0.0093$ |
| o-ACC ($r = I, \tau = 1.0e-9, v = \frac{1}{3}$) on LR ($w = u, C = 10.0$) | $0.0229 \pm 0.0093$ |
| RUN ($\tau = 1.0e-6$) on LR ($w = u, C = 0.01$) | $0.0252 \pm 0.0099$ |
| IBU ($o = 2, i = 0.001$) on LR ($w = u, C = 0.01$) | $0.0253 \pm 0.0099$ |
| CC on LR ($w = u, C = 10.0$) | $0.0526 \pm 0.0190$ |
| PCC on LR ($w = u, C = 10.0$) | $0.0629 \pm 0.0215$ |
| ARC ($v = \frac{1}{3}$) on LR ($w = u, C = 1.0$) | $0.0641 \pm 0.0226$ |
| OQT ($v = \frac{1}{3}$) on LR ($w = n, C = 1.0$) | $0.0775 \pm 0.0262$ |

Table 13: NMD on Amazon-OQ-BK, APP-OQ = level 1 out of 5 (the smoothest)

| quantification method | avg. NMD ± stddev. |
|---|---|
| o-SLD ($o = 2, i = 0.01$) on LR ($w = n, C = 0.01$) | **$0.0152 \pm 0.0057$** |
| SLD on LR ($w = n, C = 0.01$) | $0.0154 \pm 0.0058$ |
| o-PACC ($r = C_2, \tau = 0.001, v = \frac{1}{4}$) on LR ($w = u, C = 0.1$) | $0.0174 \pm 0.0068$ |
| PACC ($v = \frac{1}{4}$) on LR ($w = u, C = 0.1$) | $0.0176 \pm 0.0070$ |
| o-ACC ($r = C_2, \tau = 0.003, v = \frac{1}{4}$) on LR ($w = u, C = 0.1$) | $0.0188 \pm 0.0072$ |
| ACC ($v = \frac{1}{4}$) on LR ($w = u, C = 0.1$) | $0.0193 \pm 0.0075$ |
| IBU ($o = 2, i = 0.001$) on LR ($w = u, C = 1.0$) | $0.0197 \pm 0.0074$ |
| RUN ($\tau = 1.0e-6$) on LR ($w = u, C = 1.0$) | $0.0198 \pm 0.0074$ |
| CC on LR ($w = u, C = 10.0$) | $0.0344 \pm 0.0127$ |
| PCC on LR ($w = u, C = 10.0$) | $0.0440 \pm 0.0165$ |
| ARC ($v = \frac{1}{3}$) on LR ($w = u, C = 1.0$) | $0.0477 \pm 0.0155$ |
| OQT ($v = \frac{1}{3}$) on LR ($w = u, C = 10.0$) | $0.0587 \pm 0.0268$ |

Table 14: NMD on AMAZON-OQ-BK, level 2 out of 5

| quantification method | avg. NMD ± stddev. |
|---|---|
| SLD on LR ($w = n, C = 0.01$) | $\mathbf{0.0164 \pm 0.0061}$ |
| o-SLD ($o = 2, i = 0.001$) on LR ($w = n, C = 0.01$) | $0.0164 \pm 0.0061$ |
| PACC ($v = \frac{1}{4}$) on LR ($w = u, C = 0.1$) | $0.0190 \pm 0.0070$ |
| o-PACC ($r = I, \tau = 1.0e-9, v = \frac{1}{4}$) on LR ($w = u, C = 0.1$) | $0.0190 \pm 0.0070$ |
| ACC ($v = \frac{1}{4}$) on LR ($w = u, C = 0.1$) | $0.0210 \pm 0.0077$ |
| o-ACC ($r = I, \tau = 1.0e-9, v = \frac{1}{4}$) on LR ($w = u, C = 0.1$) | $0.0210 \pm 0.0077$ |
| RUN ($\tau = 1.0e-6$) on LR ($w = u, C = 1.0$) | $0.0221 \pm 0.0079$ |
| IBU ($o = 2, i = 0.001$) on LR ($w = u, C = 1.0$) | $0.0222 \pm 0.0079$ |
| CC on LR ($w = u, C = 10.0$) | $0.0423 \pm 0.0122$ |
| PCC on LR ($w = u, C = 10.0$) | $0.0524 \pm 0.0156$ |
| ARC ($v = \frac{1}{3}$) on LR ($w = u, C = 1.0$) | $0.0527 \pm 0.0168$ |
| OQT ($v = \frac{1}{3}$) on LR ($w = n, C = 10.0$) | $0.0654 \pm 0.0225$ |

Table 15: NMD on AMAZON-OQ-BK, level 3 out of 5

| quantification method | avg. NMD ± stddev. |
|---|---|
| SLD on LR ($w = n, C = 0.01$) | $\mathbf{0.0172 \pm 0.0066}$ |
| o-SLD ($o = 0, i = 0.01$) on LR ($w = n, C = 0.001$) | $\mathbf{0.0174 \pm 0.0076}$ |
| PACC ($v = \frac{1}{4}$) on LR ($w = u, C = 0.1$) | $0.0199 \pm 0.0077$ |
| o-PACC ($r = I, \tau = 1.0e-9, v = \frac{1}{4}$) on LR ($w = u, C = 0.1$) | $0.0199 \pm 0.0077$ |
| ACC ($v = \frac{1}{3}$) on LR ($w = u, C = 10.0$) | $0.0218 \pm 0.0085$ |
| o-ACC ($r = I, \tau = 1.0e-9, v = \frac{1}{3}$) on LR ($w = u, C = 10.0$) | $0.0218 \pm 0.0085$ |
| RUN ($\tau = 1.0e-6$) on LR ($w = u, C = 0.001$) | $0.0244 \pm 0.0089$ |
| IBU ($o = 2, i = 0.001$) on LR ($w = u, C = 0.001$) | $0.0246 \pm 0.0089$ |
| CC on LR ($w = u, C = 10.0$) | $0.0503 \pm 0.0116$ |
| PCC on LR ($w = u, C = 10.0$) | $0.0603 \pm 0.0146$ |
| ARC ($v = \frac{1}{3}$) on LR ($w = u, C = 1.0$) | $0.0604 \pm 0.0179$ |
| OQT ($v = \frac{1}{3}$) on LR ($w = n, C = 1.0$) | $0.0738 \pm 0.0231$ |

Table 16: NMD on AMAZON-OQ-BK, level 4 out of 5

| quantification method | avg. NMD ± stddev. |
|---|---|
| o-SLD ($o = 0, i = 0.01$) on LR ($w = n, C = 0.001$) | $\mathbf{0.0177 \pm 0.0072}$ |
| SLD on LR ($w = n, C = 0.01$) | $0.0178 \pm 0.0068$ |
| PACC ($v = \frac{1}{3}$) on LR ($w = u, C = 0.01$) | $0.0215 \pm 0.0081$ |
| o-PACC ($r = I, \tau = 1.0e-9, v = \frac{1}{3}$) on LR ($w = u, C = 0.01$) | $0.0215 \pm 0.0081$ |
| ACC ($v = \frac{1}{3}$) on LR ($w = u, C = 10.0$) | $0.0238 \pm 0.0093$ |
| o-ACC ($r = I, \tau = 1.0e-9, v = \frac{1}{3}$) on LR ($w = u, C = 10.0$) | $0.0238 \pm 0.0093$ |
| RUN ($\tau = 1.0e-6$) on LR ($w = u, C = 0.01$) | $0.0267 \pm 0.0091$ |
| IBU ($o = 2, i = 0.001$) on LR ($w = u, C = 0.01$) | $0.0269 \pm 0.0091$ |
| CC on LR ($w = u, C = 1.0$) | $0.0595 \pm 0.0116$ |
| ARC ($v = \frac{1}{3}$) on LR ($w = u, C = 1.0$) | $0.0695 \pm 0.0172$ |
| PCC on LR ($w = u, C = 10.0$) | $0.0700 \pm 0.0139$ |
| OQT ($v = \frac{1}{3}$) on LR ($w = n, C = 1.0$) | $0.0823 \pm 0.0219$ |

Table 17: NMD on Amazon-OQ-BK, level 5 out of 5 (the least smooth)

| quantification method | avg. NMD ± stddev. |
|---|---|
| o-SLD ($o = 0, i = 0.01$) on LR ($w = n, C = 0.001$) | **0.0177 ± 0.0071** |
| SLD on LR ($w = n, C = 0.01$) | 0.0193 ± 0.0073 |
| PACC ($v = \frac{1}{4}$) on LR ($w = n, C = 0.1$) | 0.0234 ± 0.0081 |
| o-PACC ($r = I, \tau = 1.0e - 9, v = \frac{1}{4}$) on LR ($w = n, C = 0.1$) | 0.0234 ± 0.0081 |
| ACC ($v = \frac{1}{3}$) on LR ($w = u, C = 10.0$) | 0.0286 ± 0.0106 |
| o-ACC ($r = I, \tau = 1.0e - 9, v = \frac{1}{3}$) on LR ($w = u, C = 10.0$) | 0.0286 ± 0.0106 |
| RUN ($\tau = 1.0e - 6$) on LR ($w = u, C = 0.001$) | 0.0328 ± 0.0105 |
| IBU ($o = 0, i = 0.001$) on LR ($w = u, C = 0.001$) | 0.0329 ± 0.0105 |
| CC on LR ($w = u, C = 1.0$) | 0.0761 ± 0.0135 |
| PCC on LR ($w = u, C = 10.0$) | 0.0878 ± 0.0158 |
| ARC ($v = \frac{1}{3}$) on LR ($w = u, C = 0.1$) | 0.0895 ± 0.0166 |
| OQT ($v = \frac{1}{3}$) on LR ($w = n, C = 0.01$) | 0.1023 ± 0.0193 |

Table 18: NMD on Fact-OQ, regular APP

| quantification method | avg. NMD ± stddev. |
|---|---|
| IBU ($o = 0, i = 0.01, J = 60$) | **0.0213 ± 0.0054** |
| RUN ($\tau = 1.0e - 5, J = 60$) | 0.0222 ± 0.0056 |
| o-PACC ($r = I, \tau = 0.001, v = \frac{1}{3}$) on DT ($w = n, c = E, d = 8$) | 0.0230 ± 0.0057 |
| o-ACC ($r = C_2, \tau = 0.001, v = \frac{1}{3}$) on DT ($w = u, c = G, d = 8$) | 0.0274 ± 0.0073 |
| o-SLD ($o = 0, i = 0.03$) on DT ($w = n, c = E, d = 4$) | 0.0327 ± 0.0077 |
| SLD on DT ($w = n, c = G, d = 6$) | 0.0373 ± 0.0098 |
| CC on DT ($w = u, c = G, d = 8$) | 0.0534 ± 0.0120 |
| ARC ($v = \frac{1}{3}$) on DT ($w = u, c = G, d = 8$) | 0.0566 ± 0.0142 |
| ACC ($v = \frac{1}{4}$) on DT ($w = n, c = G, d = 10$) | 0.0582 ± 0.0281 |
| PCC on DT ($w = u, c = E, d = 6$) | 0.0651 ± 0.0174 |
| OQT ($v = \frac{1}{3}$) on DT ($w = u, c = G, d = 6$) | 0.0746 ± 0.0194 |
| PACC ($v = \frac{1}{3}$) on DT ($w = n, c = G, d = 10$) | 0.0791 ± 0.0475 |

Table 19: NMD on Fact-OQ, APP-OQ = level 1 out of 5 (the smoothest)

| quantification method | avg. NMD ± stddev. |
|---|---|
| o-PACC ($r = I, \tau = 0.001, v = \frac{1}{3}$) on DT ($w = n, c = E, d = 8$) | **0.0178 ± 0.0041** |
| IBU ($o = 2, i = 0.01, J = 60$) | 0.0187 ± 0.0044 |
| RUN ($\tau = 1.0e - 5, J = 60$) | 0.0194 ± 0.0046 |
| o-ACC ($r = C_2, \tau = 0.001, v = \frac{1}{3}$) on DT ($w = u, c = G, d = 8$) | 0.0230 ± 0.0062 |
| o-SLD ($o = 0, i = 0.03$) on DT ($w = n, c = E, d = 4$) | 0.0289 ± 0.0071 |
| SLD on DT ($w = n, c = G, d = 6$) | 0.0355 ± 0.0091 |
| CC on DT ($w = u, c = G, d = 8$) | 0.0494 ± 0.0112 |
| ARC ($v = \frac{1}{3}$) on DT ($w = n, c = E, d = 6$) | 0.0568 ± 0.0161 |
| ACC ($v = \frac{1}{4}$) on DT ($w = n, c = G, d = 10$) | 0.0575 ± 0.0281 |
| PCC on DT ($w = u, c = E, d = 6$) | 0.0621 ± 0.0171 |
| OQT ($v = \frac{1}{3}$) on DT ($w = u, c = G, d = 6$) | 0.0731 ± 0.0200 |
| PACC ($v = \frac{1}{3}$) on DT ($w = n, c = G, d = 10$) | 0.0816 ± 0.0485 |

Table 20: NMD on Fact-OQ, level 2 out of 5

| quantification method | avg. NMD ± stddev. |
|---|---|
| IBU ($o = 0, i = 0.01, J = 60$) | **0.0199 ± 0.0047** |
| o-PACC ($r = I, \tau = 0.001, v = \frac{1}{3}$) on DT ($w = n, c = E, d = 8$) | 0.0203 ± 0.0039 |
| RUN ($\tau = 1.0e-5, J = 60$) | 0.0205 ± 0.0049 |
| o-ACC ($r = C_2, \tau = 0.001, v = \frac{1}{3}$) on DT ($w = u, c = G, d = 8$) | 0.0248 ± 0.0060 |
| o-SLD ($o = 0, i = 0.03$) on DT ($w = n, c = E, d = 4$) | 0.0307 ± 0.0068 |
| SLD on DT ($w = n, c = G, d = 6$) | 0.0359 ± 0.0091 |
| CC on DT ($w = u, c = G, d = 8$) | 0.0506 ± 0.0112 |
| ARC ($v = \frac{1}{3}$) on DT ($w = u, c = G, d = 8$) | 0.0556 ± 0.0147 |
| ACC ($v = \frac{1}{4}$) on DT ($w = n, c = G, d = 10$) | 0.0585 ± 0.0285 |
| PCC on DT ($w = u, c = E, d = 6$) | 0.0623 ± 0.0170 |
| OQT ($v = \frac{1}{3}$) on DT ($w = u, c = G, d = 6$) | 0.0728 ± 0.0197 |
| PACC ($v = \frac{1}{4}$) on DT ($w = n, c = G, d = 4$) | 0.0802 ± 0.0298 |

Table 21: NMD on Fact-OQ, level 3 out of 5

| quantification method | avg. NMD ± stddev. |
|---|---|
| IBU ($o = 2, i = 0.01, J = 60$) | **0.0210 ± 0.0049** |
| RUN ($\tau = 1.0e-5, J = 60$) | 0.0217 ± 0.0050 |
| o-PACC ($r = I, \tau = 0.001, v = \frac{1}{3}$) on DT ($w = n, c = E, d = 8$) | 0.0225 ± 0.0039 |
| o-ACC ($r = C_2, \tau = 0.001, v = \frac{1}{3}$) on DT ($w = u, c = G, d = 8$) | 0.0267 ± 0.0060 |
| o-SLD ($o = 0, i = 0.03$) on DT ($w = n, c = E, d = 4$) | 0.0326 ± 0.0068 |
| SLD on DT ($w = n, c = G, d = 6$) | 0.0374 ± 0.0095 |
| CC on DT ($w = u, c = G, d = 8$) | 0.0523 ± 0.0105 |
| ARC ($v = \frac{1}{3}$) on DT ($w = u, c = G, d = 8$) | 0.0562 ± 0.0141 |
| ACC ($v = \frac{1}{4}$) on DT ($w = n, c = G, d = 10$) | 0.0579 ± 0.0285 |
| PCC on DT ($w = u, c = E, d = 6$) | 0.0644 ± 0.0160 |
| OQT ($v = \frac{1}{3}$) on DT ($w = u, c = G, d = 6$) | 0.0744 ± 0.0193 |
| PACC ($v = \frac{1}{3}$) on DT ($w = n, c = G, d = 10$) | 0.0785 ± 0.0481 |

Table 22: NMD on Fact-OQ, level 4 out of 5

| quantification method | avg. NMD ± stddev. |
|---|---|
| IBU ($o = 0, i = 0.01, J = 60$) | **0.0224 ± 0.0052** |
| RUN ($\tau = 1.0e-5, J = 60$) | 0.0234 ± 0.0052 |
| o-PACC ($r = I, \tau = 0.001, v = \frac{1}{3}$) on DT ($w = n, c = E, d = 8$) | 0.0251 ± 0.0040 |
| o-ACC ($r = C_2, \tau = 0.001, v = \frac{1}{3}$) on DT ($w = u, c = G, d = 8$) | 0.0292 ± 0.0064 |
| o-SLD ($o = 0, i = 0.03$) on DT ($w = n, c = E, d = 4$) | 0.0342 ± 0.0069 |
| SLD on DT ($w = n, c = G, d = 6$) | 0.0380 ± 0.0094 |
| CC on DT ($w = u, c = G, d = 8$) | 0.0543 ± 0.0110 |
| ARC ($v = \frac{1}{3}$) on DT ($w = u, c = G, d = 8$) | 0.0561 ± 0.0138 |
| ACC ($v = \frac{1}{4}$) on DT ($w = n, c = G, d = 10$) | 0.0582 ± 0.0277 |
| PCC on DT ($w = u, c = E, d = 6$) | 0.0653 ± 0.0162 |
| OQT ($v = \frac{1}{3}$) on DT ($w = u, c = G, d = 6$) | 0.0745 ± 0.0184 |
| PACC ($v = \frac{1}{4}$) on DT ($w = u, c = E, d = 12$) | 0.0788 ± 0.0320 |

Table 23: NMD on Fact-OQ, level 5 out of 5 (the least smooth)

| quantification method | avg. NMD ± stddev. |
|---|---|
| IBU ($o = 1, i = 0.0, J = 60$) | $\mathbf{0.0245 \pm 0.0067}$ |
| RUN ($\tau = 1.0e-5, J = 60$) | $0.0262 \pm 0.0058$ |
| o-PACC ($r = I, \tau = 0.001, v = \frac{1}{3}$) on DT ($w = u, c = G, d = 10$) | $0.0298 \pm 0.0049$ |
| o-ACC ($r = C_2, \tau = 0.001, v = \frac{1}{3}$) on DT ($w = u, c = G, d = 10$) | $0.0330 \pm 0.0062$ |
| o-SLD ($o = 0, i = 0.01$) on DT ($w = n, c = E, d = 4$) | $0.0368 \pm 0.0096$ |
| SLD on DT ($w = n, c = E, d = 6$) | $0.0393 \pm 0.0112$ |
| ARC ($v = \frac{1}{3}$) on DT ($w = u, c = G, d = 8$) | $0.0583 \pm 0.0131$ |
| CC on DT ($w = u, c = G, d = 8$) | $0.0604 \pm 0.0129$ |
| ACC ($v = \frac{1}{3}$) on DT ($w = n, c = E, d = 8$) | $0.0646 \pm 0.0274$ |
| PCC on DT ($w = u, c = E, d = 6$) | $0.0715 \pm 0.0188$ |
| PACC ($v = \frac{1}{3}$) on DT ($w = n, c = G, d = 10$) | $0.0776 \pm 0.0455$ |
| OQT ($v = \frac{1}{3}$) on DT ($w = u, c = G, d = 6$) | $0.0783 \pm 0.0193$ |

Table 24: NMD on Amazon-OQ-BK, in an alternative TFIDF representation, regular APP

| quantification method | avg. NMD ± stddev. |
|---|---|
| SLD on LR ($w = n, C = 0.01$) | $\mathbf{0.0172 \pm 0.0067}$ |
| o-SLD ($o = 0, i = 0.001$) on LR ($w = n, C = 0.01$) | $0.0173 \pm 0.0067$ |
| PACC ($v = \frac{1}{4}$) on LR ($w = u, C = 0.1$) | $0.0209 \pm 0.0083$ |
| o-PACC ($r = I, \tau = 1.0e-9, v = \frac{1}{4}$) on LR ($w = u, C = 0.1$) | $0.0209 \pm 0.0083$ |
| ACC ($v = \frac{1}{3}$) on LR ($w = u, C = 10.0$) | $0.0229 \pm 0.0093$ |
| o-ACC ($r = I, \tau = 1.0e-9, v = \frac{1}{3}$) on LR ($w = u, C = 10.0$) | $0.0229 \pm 0.0093$ |
| RUN ($\tau = 1.0e-6$) on LR ($w = u, C = 0.01$) | $0.0252 \pm 0.0099$ |
| IBU ($o = 2, i = 0.001$) on LR ($w = u, C = 0.01$) | $0.0253 \pm 0.0099$ |
| CC on LR ($w = u, C = 10.0$) | $0.0526 \pm 0.0190$ |
| PCC on LR ($w = u, C = 10.0$) | $0.0629 \pm 0.0215$ |
| ARC ($v = \frac{1}{3}$) on LR ($w = u, C = 1.0$) | $0.0641 \pm 0.0226$ |
| OQT ($v = \frac{1}{3}$) on LR ($w = n, C = 1.0$) | $0.0775 \pm 0.0262$ |

Table 25: NMD on Amazon-OQ-BK, in an alternative TFIDF representation, APP-OQ = level 1 out of 5 (the smoothest)

| quantification method | avg. NMD ± stddev. |
|---|---|
| o-SLD ($o = 2, i = 0.01$) on LR ($w = n, C = 0.01$) | $\mathbf{0.0152 \pm 0.0057}$ |
| SLD on LR ($w = n, C = 0.01$) | $0.0154 \pm 0.0058$ |
| o-PACC ($r = C_2, \tau = 0.001, v = \frac{1}{4}$) on LR ($w = u, C = 0.1$) | $0.0174 \pm 0.0068$ |
| PACC ($v = \frac{1}{4}$) on LR ($w = u, C = 0.1$) | $0.0176 \pm 0.0070$ |
| o-ACC ($r = C_2, \tau = 0.003, v = \frac{1}{4}$) on LR ($w = u, C = 0.1$) | $0.0188 \pm 0.0072$ |
| ACC ($v = \frac{1}{4}$) on LR ($w = u, C = 0.1$) | $0.0193 \pm 0.0075$ |
| IBU ($o = 2, i = 0.001$) on LR ($w = u, C = 1.0$) | $0.0197 \pm 0.0074$ |
| RUN ($\tau = 1.0e-6$) on LR ($w = u, C = 1.0$) | $0.0198 \pm 0.0074$ |
| CC on LR ($w = u, C = 10.0$) | $0.0344 \pm 0.0127$ |
| PCC on LR ($w = u, C = 10.0$) | $0.0440 \pm 0.0165$ |
| ARC ($v = \frac{1}{3}$) on LR ($w = u, C = 1.0$) | $0.0477 \pm 0.0155$ |
| OQT ($v = \frac{1}{3}$) on LR ($w = u, C = 10.0$) | $0.0587 \pm 0.0268$ |

Table 26: NMD on Amazon-OQ-BK, in an alternative TFIDF representation, level 2 out of 5

| quantification method | avg. NMD ± stddev. |
|---|---|
| SLD on LR ($w = n, C = 0.01$) | **$0.0164 \pm 0.0061$** |
| o-SLD ($o = 2, i = 0.001$) on LR ($w = n, C = 0.01$) | $0.0164 \pm 0.0061$ |
| PACC ($v = \frac{1}{4}$) on LR ($w = u, C = 0.1$) | $0.0190 \pm 0.0070$ |
| o-PACC ($r = I, \tau = 1.0e-9, v = \frac{1}{4}$) on LR ($w = u, C = 0.1$) | $0.0190 \pm 0.0070$ |
| ACC ($v = \frac{1}{4}$) on LR ($w = u, C = 0.1$) | $0.0210 \pm 0.0077$ |
| o-ACC ($r = I, \tau = 1.0e-9, v = \frac{1}{4}$) on LR ($w = u, C = 0.1$) | $0.0210 \pm 0.0077$ |
| RUN ($\tau = 1.0e-6$) on LR ($w = u, C = 1.0$) | $0.0221 \pm 0.0079$ |
| IBU ($o = 2, i = 0.001$) on LR ($w = u, C = 1.0$) | $0.0222 \pm 0.0079$ |
| CC on LR ($w = u, C = 10.0$) | $0.0423 \pm 0.0122$ |
| PCC on LR ($w = u, C = 10.0$) | $0.0524 \pm 0.0156$ |
| ARC ($v = \frac{1}{3}$) on LR ($w = u, C = 1.0$) | $0.0527 \pm 0.0168$ |
| OQT ($v = \frac{1}{3}$) on LR ($w = n, C = 10.0$) | $0.0654 \pm 0.0225$ |

Table 27: NMD on Amazon-OQ-BK, in an alternative TFIDF representation, level 3 out of 5

| quantification method | avg. NMD ± stddev. |
|---|---|
| SLD on LR ($w = n, C = 0.01$) | **$0.0172 \pm 0.0066$** |
| o-SLD ($o = 0, i = 0.01$) on LR ($w = n, C = 0.001$) | **$0.0174 \pm 0.0076$** |
| PACC ($v = \frac{1}{4}$) on LR ($w = u, C = 0.1$) | $0.0199 \pm 0.0077$ |
| o-PACC ($r = I, \tau = 1.0e-9, v = \frac{1}{4}$) on LR ($w = u, C = 0.1$) | $0.0199 \pm 0.0077$ |
| ACC ($v = \frac{1}{3}$) on LR ($w = u, C = 10.0$) | $0.0218 \pm 0.0085$ |
| o-ACC ($r = I, \tau = 1.0e-9, v = \frac{1}{3}$) on LR ($w = u, C = 10.0$) | $0.0218 \pm 0.0085$ |
| RUN ($\tau = 1.0e-6$) on LR ($w = u, C = 0.001$) | $0.0244 \pm 0.0089$ |
| IBU ($o = 2, i = 0.001$) on LR ($w = u, C = 0.001$) | $0.0246 \pm 0.0089$ |
| CC on LR ($w = u, C = 10.0$) | $0.0503 \pm 0.0116$ |
| PCC on LR ($w = u, C = 10.0$) | $0.0603 \pm 0.0146$ |
| ARC ($v = \frac{1}{3}$) on LR ($w = u, C = 1.0$) | $0.0604 \pm 0.0179$ |
| OQT ($v = \frac{1}{3}$) on LR ($w = n, C = 1.0$) | $0.0738 \pm 0.0231$ |

Table 28: NMD on Amazon-OQ-BK, in an alternative TFIDF representation, level 4 out of 5

| quantification method | avg. NMD ± stddev. |
|---|---|
| o-SLD ($o = 0, i = 0.01$) on LR ($w = n, C = 0.001$) | **$0.0177 \pm 0.0072$** |
| SLD on LR ($w = n, C = 0.01$) | $0.0178 \pm 0.0068$ |
| PACC ($v = \frac{1}{3}$) on LR ($w = u, C = 0.01$) | $0.0215 \pm 0.0081$ |
| o-PACC ($r = I, \tau = 1.0e-9, v = \frac{1}{3}$) on LR ($w = u, C = 0.01$) | $0.0215 \pm 0.0081$ |
| ACC ($v = \frac{1}{3}$) on LR ($w = u, C = 10.0$) | $0.0238 \pm 0.0093$ |
| o-ACC ($r = I, \tau = 1.0e-9, v = \frac{1}{3}$) on LR ($w = u, C = 10.0$) | $0.0238 \pm 0.0093$ |
| RUN ($\tau = 1.0e-6$) on LR ($w = u, C = 0.01$) | $0.0267 \pm 0.0091$ |
| IBU ($o = 2, i = 0.001$) on LR ($w = u, C = 0.01$) | $0.0269 \pm 0.0091$ |
| CC on LR ($w = u, C = 1.0$) | $0.0595 \pm 0.0116$ |
| ARC ($v = \frac{1}{3}$) on LR ($w = u, C = 1.0$) | $0.0695 \pm 0.0172$ |
| PCC on LR ($w = u, C = 10.0$) | $0.0700 \pm 0.0139$ |
| OQT ($v = \frac{1}{3}$) on LR ($w = n, C = 1.0$) | $0.0823 \pm 0.0219$ |

Table 29: NMD on Amazon-OQ-BK, in an alternative TFIDF representation, level 5 out of 5 (the least smooth)

| quantification method | avg. NMD ± stddev. |
|---|---|
| o-SLD ($o=0, i=0.01$) on LR ($w=n, C=0.001$) | **0.0177 ± 0.0071** |
| SLD on LR ($w=n, C=0.01$) | $0.0193 \pm 0.0073$ |
| PACC ($v=\frac{1}{4}$) on LR ($w=n, C=0.1$) | $0.0234 \pm 0.0081$ |
| o-PACC ($r=I, \tau=1.0e-9, v=\frac{1}{4}$) on LR ($w=n, C=0.1$) | $0.0234 \pm 0.0081$ |
| ACC ($v=\frac{1}{3}$) on LR ($w=u, C=10.0$) | $0.0286 \pm 0.0106$ |
| o-ACC ($r=I, \tau=1.0e-9, v=\frac{1}{3}$) on LR ($w=u, C=10.0$) | $0.0286 \pm 0.0106$ |
| RUN ($\tau=1.0e-6$) on LR ($w=u, C=0.001$) | $0.0328 \pm 0.0105$ |
| IBU ($o=0, i=0.001$) on LR ($w=u, C=0.001$) | $0.0329 \pm 0.0105$ |
| CC on LR ($w=u, C=1.0$) | $0.0761 \pm 0.0135$ |
| PCC on LR ($w=u, C=10.0$) | $0.0878 \pm 0.0158$ |
| ARC ($v=\frac{1}{3}$) on LR ($w=u, C=0.1$) | $0.0895 \pm 0.0166$ |
| OQT ($v=\frac{1}{3}$) on LR ($w=n, C=0.01$) | $0.1023 \pm 0.0193$ |

Table 30: NMD on Uci-blog-feedback-OQ, regular APP

| quantification method | avg. NMD ± stddev. |
|---|---|
| o-PACC ($r=I, \tau=0.001, v=\frac{1}{3}$) on LR ($w=u, C=0.1$) | **0.0747 ± 0.0278** |
| o-ACC ($r=I, \tau=0.1, v=\frac{1}{3}$) on LR ($w=u, C=0.01$) | $0.0772 \pm 0.0310$ |
| CC on LR ($w=u, C=1.0$) | $0.0958 \pm 0.0337$ |
| PCC on LR ($w=u, C=10.0$) | $0.0967 \pm 0.0420$ |
| IBU ($o=0, i=0.1, J=60$) | $0.0997 \pm 0.0458$ |
| SLD on DT ($w=u, c=G, d=12$) | $0.1001 \pm 0.0442$ |
| ACC ($v=\frac{1}{2}$) on LR ($w=u, C=0.001$) | $0.1147 \pm 0.0419$ |
| o-SLD ($o=1, i=0.1$) on DT ($w=n, c=G, d=10$) | $0.1195 \pm 0.0413$ |
| PACC ($v=\frac{1}{2}$) on DT ($w=u, c=E, d=8$) | $0.1323 \pm 0.0487$ |
| RUN ($\tau=1.0e-5, J=60$) | $0.1348 \pm 0.0518$ |
| OQT ($v=\frac{1}{3}$) on LR ($w=u, C=10.0$) | $0.2222 \pm 0.0578$ |
| ARC ($v=\frac{1}{3}$) on LR ($w=u, C=0.01$) | $0.2420 \pm 0.0618$ |

Table 31: NMD on Uci-blog-feedback-OQ, APP-OQ = level 1 out of 5 (the smoothest)

| quantification method | avg. NMD ± stddev. |
|---|---|
| o-PACC ($r=I, \tau=0.001, v=\frac{1}{3}$) on LR ($w=u, C=0.1$) | **0.0664 ± 0.0249** |
| o-ACC ($r=I, \tau=0.1, v=\frac{1}{3}$) on LR ($w=u, C=0.001$) | $0.0728 \pm 0.0268$ |
| CC on LR ($w=u, C=1.0$) | $0.0884 \pm 0.0310$ |
| PCC on LR ($w=u, C=10.0$) | $0.0960 \pm 0.0454$ |
| IBU ($o=0, i=0.1, J=60$) | $0.0980 \pm 0.0495$ |
| ACC ($v=\frac{1}{2}$) on LR ($w=u, C=0.001$) | $0.1144 \pm 0.0451$ |
| o-SLD ($o=1, i=0.1$) on DT ($w=n, c=G, d=10$) | $0.1190 \pm 0.0402$ |
| SLD on DT ($w=n, c=G, d=8$) | $0.1224 \pm 0.0376$ |
| RUN ($\tau=1.0e-5, J=60$) | $0.1339 \pm 0.0539$ |
| PACC ($v=\frac{1}{2}$) on DT ($w=u, c=E, d=8$) | $0.1437 \pm 0.0497$ |
| OQT ($v=\frac{1}{3}$) on LR ($w=u, C=10.0$) | $0.2050 \pm 0.0566$ |
| ARC ($v=\frac{1}{3}$) on LR ($w=u, C=0.01$) | $0.2474 \pm 0.0630$ |

Table 32: NMD on Uci-blog-feedback-OQ, level 2 out of 5

| quantification method | avg. NMD ± stddev. |
|---|---|
| o-PACC ($r = I, \tau = 0.001, v = \frac{1}{3}$) on LR ($w = u, C = 0.1$) | **0.0699 ± 0.0242** |
| o-ACC ($r = I, \tau = 0.1, v = \frac{1}{3}$) on LR ($w = u, C = 0.001$) | 0.0752 ± 0.0274 |
| CC on LR ($w = u, C = 1.0$) | 0.0902 ± 0.0312 |
| o-SLD ($o = 0, i = 0.1$) on DT ($w = n, c = G, d = 8$) | 0.0926 ± 0.0374 |
| PCC on LR ($w = u, C = 10.0$) | 0.0933 ± 0.0410 |
| IBU ($o = 0, i = 0.1, J = 60$) | 0.0981 ± 0.0470 |
| RUN ($\tau = 0.1, J = 60$) | 0.1091 ± 0.0476 |
| ACC ($v = \frac{1}{2}$) on LR ($w = u, C = 0.001$) | 0.1114 ± 0.0409 |
| SLD on DT ($w = n, c = G, d = 8$) | 0.1231 ± 0.0372 |
| PACC ($v = \frac{1}{2}$) on DT ($w = u, c = E, d = 8$) | 0.1360 ± 0.0478 |
| OQT ($v = \frac{1}{3}$) on LR ($w = u, C = 10.0$) | 0.2126 ± 0.0559 |
| ARC ($v = \frac{1}{3}$) on LR ($w = u, C = 0.01$) | 0.2450 ± 0.0606 |

Table 33: NMD on Uci-blog-feedback-OQ, level 3 out of 5

| quantification method | avg. NMD ± stddev. |
|---|---|
| o-ACC ($r = I, \tau = 0.1, v = \frac{1}{3}$) on LR ($w = u, C = 0.01$) | **0.0735 ± 0.0293** |
| o-PACC ($r = I, \tau = 0.001, v = \frac{1}{3}$) on LR ($w = u, C = 0.01$) | 0.0809 ± 0.0328 |
| CC on LR ($w = u, C = 1.0$) | 0.0921 ± 0.0317 |
| PCC on LR ($w = u, C = 10.0$) | 0.0933 ± 0.0420 |
| IBU ($o = 0, i = 0.1, J = 60$) | 0.0980 ± 0.0445 |
| SLD on DT ($w = u, c = G, d = 12$) | 0.0999 ± 0.0480 |
| ACC ($v = \frac{1}{2}$) on LR ($w = u, C = 0.001$) | 0.1121 ± 0.0423 |
| o-SLD ($o = 1, i = 0.1$) on DT ($w = n, c = G, d = 10$) | 0.1200 ± 0.0396 |
| PACC ($v = \frac{1}{2}$) on DT ($w = u, c = E, d = 8$) | 0.1301 ± 0.0453 |
| RUN ($\tau = 1.0e-5, J = 60$) | 0.1331 ± 0.0503 |
| OQT ($v = \frac{1}{3}$) on LR ($w = u, C = 10.0$) | 0.2194 ± 0.0556 |
| ARC ($v = \frac{1}{3}$) on LR ($w = u, C = 0.01$) | 0.2422 ± 0.0593 |

Table 34: NMD on Uci-blog-feedback-OQ, level 4 out of 5

| quantification method | avg. NMD ± stddev. |
|---|---|
| o-ACC ($r = I, \tau = 0.1, v = \frac{1}{3}$) on LR ($w = u, C = 0.01$) | **0.0778 ± 0.0303** |
| o-PACC ($r = I, \tau = 0.001, v = \frac{1}{3}$) on LR ($w = u, C = 0.01$) | 0.0875 ± 0.0401 |
| PCC on LR ($w = u, C = 10.0$) | 0.0952 ± 0.0416 |
| SLD on DT ($w = u, c = G, d = 12$) | 0.0970 ± 0.0402 |
| CC on LR ($w = u, C = 1.0$) | 0.0976 ± 0.0342 |
| IBU ($o = 0, i = 0.1, J = 60$) | 0.0989 ± 0.0431 |
| RUN ($\tau = 0.1, J = 60$) | 0.1047 ± 0.0425 |
| ACC ($v = \frac{1}{3}$) on LR ($w = u, C = 10.0$) | 0.1110 ± 0.0342 |
| o-SLD ($o = 1, i = 0.1$) on DT ($w = n, c = G, d = 10$) | 0.1166 ± 0.0417 |
| PACC ($v = \frac{1}{2}$) on DT ($w = u, c = E, d = 8$) | 0.1246 ± 0.0481 |
| OQT ($v = \frac{1}{3}$) on LR ($w = u, C = 10.0$) | 0.2271 ± 0.0550 |
| ARC ($v = \frac{1}{3}$) on LR ($w = u, C = 0.01$) | 0.2407 ± 0.0611 |

Table 35: NMD on Uci-blog-feedback-OQ, level 5 out of 5 (the least smooth)

| quantification method | avg. NMD ± stddev. |
|---|---|
| o-ACC ($r = I, \tau = 0.1, v = \frac{1}{3}$) on LR ($w = u, C = 0.01$) | **0.0913 ± 0.0309** |
| o-PACC ($r = C_2, \tau = 0.1, v = \frac{1}{3}$) on LR ($w = u, C = 1.0$) | **0.0962 ± 0.0525** |
| SLD on DT ($w = u, c = G, d = 12$) | 0.0977 ± 0.0329 |
| RUN ($\tau = 0.1, J = 60$) | 0.1052 ± 0.0410 |
| PCC on LR ($w = u, C = 0.1$) | 0.1053 ± 0.0385 |
| IBU ($o = 0, i = 0.1, J = 60$) | 0.1055 ± 0.0444 |
| ACC ($v = \frac{1}{3}$) on LR ($w = u, C = 10.0$) | 0.1090 ± 0.0352 |
| CC on LR ($w = u, C = 0.1$) | 0.1133 ± 0.0360 |
| o-SLD ($o = 1, i = 0.1$) on DT ($w = n, c = G, d = 10$) | 0.1232 ± 0.0444 |
| PACC ($v = \frac{1}{2}$) on DT ($w = u, c = E, d = 8$) | 0.1272 ± 0.0499 |
| ARC ($v = \frac{1}{3}$) on LR ($w = u, C = 0.01$) | 0.2347 ± 0.0644 |
| OQT ($v = \frac{1}{3}$) on LR ($w = u, C = 10.0$) | 0.2471 ± 0.0571 |

Table 36: NMD on Uci-online-news-popularity-OQ, regular APP

| quantification method | avg. NMD ± stddev. |
|---|---|
| o-ACC ($r = I, \tau = 0.1, v = \frac{1}{3}$) on LR ($w = u, C = 0.001$) | **0.0833 ± 0.0298** |
| IBU ($o = 0, i = 0.1, J = 60$) | 0.0886 ± 0.0394 |
| o-PACC ($r = C_2, \tau = 0.1, v = \frac{1}{3}$) on DT ($w = u, c = E, d = 8$) | 0.0954 ± 0.0389 |
| o-SLD ($o = 1, i = 0.1$) on DT ($w = n, c = G, d = 8$) | 0.0993 ± 0.0436 |
| PCC on LR ($w = u, C = 0.01$) | 0.0996 ± 0.0436 |
| RUN ($\tau = 0.1, J = 60$) | 0.1115 ± 0.0481 |
| ACC ($v = \frac{1}{3}$) on LR ($w = u, C = 0.1$) | 0.1365 ± 0.0554 |
| PACC ($v = \frac{1}{3}$) on DT ($w = n, c = E, d = 4$) | 0.1515 ± 0.0632 |
| SLD on DT ($w = n, c = E, d = 10$) | 0.1576 ± 0.0630 |
| CC on LR ($w = u, C = 0.001$) | 0.1664 ± 0.0473 |
| OQT ($v = \frac{1}{3}$) on LR ($w = u, C = 10.0$) | 0.3220 ± 0.0872 |
| ARC ($v = \frac{1}{3}$) on LR ($w = u, C = 1.0$) | 0.3801 ± 0.0846 |

Table 37: NMD on Uci-online-news-popularity-OQ, APP-OQ = level 1 out of 5 (the smoothest)

| quantification method | avg. NMD ± stddev. |
|---|---|
| o-ACC ($r = I, \tau = 0.1, v = \frac{1}{3}$) on LR ($w = u, C = 0.01$) | **0.0718 ± 0.0268** |
| o-PACC ($r = C_2, \tau = 0.1, v = \frac{1}{3}$) on DT ($w = u, c = E, d = 8$) | 0.0804 ± 0.0309 |
| IBU ($o = 0, i = 0.1, J = 60$) | 0.0858 ± 0.0428 |
| PCC on LR ($w = u, C = 0.01$) | 0.0985 ± 0.0474 |
| o-SLD ($o = 1, i = 0.1$) on DT ($w = n, c = G, d = 8$) | 0.0992 ± 0.0459 |
| RUN ($\tau = 0.1, J = 60$) | 0.1181 ± 0.0526 |
| PACC ($v = \frac{1}{3}$) on DT ($w = u, c = E, d = 10$) | 0.1246 ± 0.0546 |
| ACC ($v = \frac{1}{3}$) on LR ($w = u, C = 0.1$) | 0.1357 ± 0.0599 |
| CC on LR ($w = u, C = 0.001$) | 0.1549 ± 0.0448 |
| SLD on DT ($w = n, c = E, d = 10$) | 0.1687 ± 0.0691 |
| OQT ($v = \frac{1}{3}$) on LR ($w = u, C = 10.0$) | 0.3177 ± 0.0925 |
| ARC ($v = \frac{1}{3}$) on LR ($w = u, C = 1.0$) | 0.3793 ± 0.0893 |

Table 38: NMD on Uci-online-news-popularity-OQ, level 2 out of 5

| quantification method | avg. NMD ± stddev. |
|---|---|
| o-ACC ($r = I, \tau = 0.1, v = \frac{1}{3}$) on LR ($w = u, C = 0.001$) | **0.0792 ± 0.0281** |
| IBU ($o = 0, i = 0.1, J = 60$) | 0.0849 ± 0.0407 |
| o-PACC ($r = C_2, \tau = 0.1, v = \frac{1}{3}$) on DT ($w = u, c = E, d = 8$) | 0.0878 ± 0.0342 |
| PCC on LR ($w = u, C = 0.01$) | 0.0952 ± 0.0436 |
| o-SLD ($o = 1, i = 0.1$) on DT ($w = n, c = G, d = 8$) | 0.0956 ± 0.0426 |
| RUN ($\tau = 0.1, J = 60$) | 0.1137 ± 0.0506 |
| ACC ($v = \frac{1}{3}$) on LR ($w = u, C = 0.1$) | 0.1350 ± 0.0532 |
| PACC ($v = \frac{1}{3}$) on DT ($w = n, c = E, d = 4$) | 0.1528 ± 0.0632 |
| CC on LR ($w = u, C = 0.001$) | 0.1583 ± 0.0445 |
| SLD on DT ($w = n, c = E, d = 10$) | 0.1648 ± 0.0662 |
| OQT ($v = \frac{1}{3}$) on LR ($w = u, C = 10.0$) | 0.3201 ± 0.0862 |
| ARC ($v = \frac{1}{3}$) on LR ($w = u, C = 1.0$) | 0.3799 ± 0.0838 |

Table 39: NMD on Uci-online-news-popularity-OQ, level 3 out of 5

| quantification method | avg. NMD ± stddev. |
|---|---|
| o-ACC ($r = I, \tau = 0.1, v = \frac{1}{3}$) on LR ($w = u, C = 0.001$) | **0.0807 ± 0.0291** |
| IBU ($o = 0, i = 0.1, J = 60$) | 0.0865 ± 0.0403 |
| o-PACC ($r = C_2, \tau = 0.1, v = \frac{1}{3}$) on DT ($w = u, c = E, d = 8$) | 0.0952 ± 0.0377 |
| PCC on LR ($w = u, C = 0.01$) | 0.0966 ± 0.0439 |
| o-SLD ($o = 1, i = 0.1$) on DT ($w = n, c = G, d = 8$) | 0.0977 ± 0.0428 |
| RUN ($\tau = 0.1, J = 60$) | 0.1116 ± 0.0506 |
| ACC ($v = \frac{1}{3}$) on LR ($w = u, C = 0.1$) | 0.1352 ± 0.0566 |
| PACC ($v = \frac{1}{3}$) on DT ($w = n, c = E, d = 4$) | 0.1509 ± 0.0630 |
| SLD on DT ($w = n, c = E, d = 10$) | 0.1578 ± 0.0643 |
| CC on LR ($w = u, C = 0.001$) | 0.1630 ± 0.0444 |
| OQT ($v = \frac{1}{3}$) on LR ($w = u, C = 10.0$) | 0.3217 ± 0.0863 |
| ARC ($v = \frac{1}{3}$) on LR ($w = u, C = 1.0$) | 0.3803 ± 0.0843 |

Table 40: NMD on Uci-online-news-popularity-OQ, level 4 out of 5

| quantification method | avg. NMD ± stddev. |
|---|---|
| o-ACC ($r = I, \tau = 0.1, v = \frac{1}{3}$) on LR ($w = u, C = 0.001$) | **0.0832 ± 0.0291** |
| IBU ($o = 0, i = 0.1, J = 60$) | **0.0874 ± 0.0360** |
| PCC on LR ($w = u, C = 0.01$) | 0.0978 ± 0.0423 |
| o-SLD ($o = 1, i = 0.1$) on DT ($w = n, c = G, d = 8$) | 0.0989 ± 0.0426 |
| o-PACC ($r = C_2, \tau = 0.001, v = \frac{1}{3}$) on DT ($w = u, c = E, d = 8$) | 0.1004 ± 0.0415 |
| RUN ($\tau = 0.1, J = 60$) | 0.1048 ± 0.0434 |
| ACC ($v = \frac{1}{3}$) on LR ($w = u, C = 0.1$) | 0.1361 ± 0.0548 |
| PACC ($v = \frac{1}{3}$) on DT ($w = n, c = E, d = 4$) | 0.1488 ± 0.0618 |
| SLD on DT ($w = n, c = E, d = 10$) | 0.1528 ± 0.0562 |
| CC on LR ($w = u, C = 0.001$) | 0.1677 ± 0.0467 |
| OQT ($v = \frac{1}{3}$) on LR ($w = u, C = 10.0$) | 0.3220 ± 0.0849 |
| ARC ($v = \frac{1}{3}$) on LR ($w = u, C = 1.0$) | 0.3790 ± 0.0827 |

Table 41: NMD on Uci-online-news-popularity-OQ, level 5 out of 5 (the least smooth)

| quantification method | avg. NMD ± stddev. |
|---|---|
| o-ACC ($r = I, \tau = 0.1, v = \frac{1}{3}$) on LR ($w = u, C = 0.01$) | **0.0923 ± 0.0267** |
| IBU ($o = 1, i = 0.1, J = 60$) | **0.0967 ± 0.0342** |
| RUN ($\tau = 0.1, J = 60$) | 0.1095 ± 0.0414 |
| PCC on LR ($w = u, C = 0.01$) | 0.1099 ± 0.0390 |
| o-PACC ($r = C_2, \tau = 0.001, v = \frac{1}{3}$) on DT ($w = u, c = E, d = 8$) | 0.1105 ± 0.0392 |
| PACC ($v = \frac{1}{2}$) on LR ($w = u, C = 0.001$) | 0.1149 ± 0.0404 |
| o-SLD ($o = 0, i = 0.01$) on DT ($w = n, c = E, d = 10$) | 0.1161 ± 0.0475 |
| ACC ($v = \frac{1}{3}$) on LR ($w = u, C = 0.1$) | 0.1404 ± 0.0519 |
| SLD on DT ($w = n, c = E, d = 10$) | 0.1437 ± 0.0549 |
| CC on LR ($w = u, C = 0.001$) | 0.1881 ± 0.0485 |
| OQT ($v = \frac{1}{3}$) on LR ($w = u, C = 10.0$) | 0.3285 ± 0.0859 |
| ARC ($v = \frac{1}{3}$) on LR ($w = u, C = 1.0$) | 0.3822 ± 0.0826 |

Table 42: NMD on OpenMl-Yolanda-OQ, regular APP

| quantification method | avg. NMD ± stddev. |
|---|---|
| IBU ($o = 0, i = 0.01, J = 60$) | **0.0558 ± 0.0168** |
| o-ACC ($r = I, \tau = 0.1, v = \frac{1}{3}$) on LR ($w = u, C = 0.001$) | 0.0568 ± 0.0156 |
| RUN ($\tau = 1.0e-5, J = 60$) | 0.0577 ± 0.0169 |
| o-PACC ($r = C_2, \tau = 0.1, v = \frac{1}{3}$) on LR ($w = u, C = 0.001$) | 0.0580 ± 0.0143 |
| o-SLD ($o = 0, i = 0.1$) on DT ($w = n, c = G, d = 4$) | 0.0701 ± 0.0187 |
| SLD on LR ($w = n, C = 10.0$) | 0.0753 ± 0.0254 |
| CC on LR ($w = u, C = 0.01$) | 0.0767 ± 0.0225 |
| ACC ($v = \frac{1}{4}$) on LR ($w = u, C = 10.0$) | 0.0807 ± 0.0238 |
| PCC on LR ($w = u, C = 0.01$) | 0.0926 ± 0.0305 |
| PACC ($v = \frac{1}{2}$) on LR ($w = u, C = 0.01$) | 0.1068 ± 0.0466 |
| OQT ($v = \frac{1}{3}$) on LR ($w = u, C = 0.001$) | 0.2246 ± 0.0562 |
| ARC ($v = \frac{1}{3}$) on LR ($w = n, C = 10.0$) | 0.2513 ± 0.0585 |

Table 43: NMD on OpenMl-Yolanda-OQ, APP-OQ = level 1 out of 5 (the smoothest)

| quantification method | avg. NMD ± stddev. |
|---|---|
| o-PACC ($r = C_2, \tau = 0.1, v = \frac{1}{3}$) on LR ($w = u, C = 0.001$) | **0.0537 ± 0.0138** |
| o-ACC ($r = I, \tau = 0.1, v = \frac{1}{3}$) on LR ($w = u, C = 0.001$) | 0.0549 ± 0.0167 |
| IBU ($o = 0, i = 0.1, J = 60$) | 0.0553 ± 0.0179 |
| RUN ($\tau = 0.001, J = 60$) | 0.0604 ± 0.0179 |
| o-SLD ($o = 0, i = 0.1$) on DT ($w = n, c = G, d = 4$) | 0.0648 ± 0.0188 |
| CC on LR ($w = u, C = 0.01$) | 0.0779 ± 0.0245 |
| SLD on LR ($w = n, C = 10.0$) | 0.0784 ± 0.0276 |
| ACC ($v = \frac{1}{4}$) on LR ($w = u, C = 10.0$) | 0.0824 ± 0.0259 |
| PCC on LR ($w = u, C = 10.0$) | 0.0921 ± 0.0320 |
| PACC ($v = \frac{1}{2}$) on LR ($w = u, C = 0.01$) | 0.1102 ± 0.0502 |
| OQT ($v = \frac{1}{3}$) on LR ($w = u, C = 0.001$) | 0.2223 ± 0.0579 |
| ARC ($v = \frac{1}{3}$) on LR ($w = n, C = 10.0$) | 0.2500 ± 0.0596 |

Table 44: NMD on OpenMl-Yolanda-OQ, level 2 out of 5

| quantification method | avg. NMD ± stddev. |
|---|---|
| o-PACC ($r = C_2, \tau = 0.1, v = \frac{1}{3}$) on LR ($w = u, C = 0.001$) | $\mathbf{0.0552 \pm 0.0129}$ |
| o-ACC ($r = I, \tau = 0.1, v = \frac{1}{3}$) on LR ($w = u, C = 0.001$) | $\mathbf{0.0554 \pm 0.0154}$ |
| IBU ($o = 0, i = 0.01, J = 60$) | $\mathbf{0.0555 \pm 0.0168}$ |
| RUN ($\tau = 1.0e-5, J = 60$) | $0.0574 \pm 0.0172$ |
| o-SLD ($o = 0, i = 0.1$) on DT ($w = n, c = G, d = 4$) | $0.0671 \pm 0.0174$ |
| SLD on LR ($w = n, C = 10.0$) | $0.0763 \pm 0.0255$ |
| CC on LR ($w = u, C = 0.01$) | $0.0769 \pm 0.0220$ |
| ACC ($v = \frac{1}{4}$) on LR ($w = u, C = 10.0$) | $0.0813 \pm 0.0233$ |
| PCC on LR ($w = u, C = 0.01$) | $0.0923 \pm 0.0293$ |
| PACC ($v = \frac{1}{2}$) on LR ($w = u, C = 0.01$) | $0.1083 \pm 0.0454$ |
| OQT ($v = \frac{1}{3}$) on LR ($w = u, C = 0.001$) | $0.2235 \pm 0.0561$ |
| ARC ($v = \frac{1}{3}$) on LR ($w = n, C = 10.0$) | $0.2508 \pm 0.0574$ |

Table 45: NMD on OpenMl-Yolanda-OQ, level 3 out of 5

| quantification method | avg. NMD ± stddev. |
|---|---|
| IBU ($o = 0, i = 0.01, J = 60$) | $\mathbf{0.0555 \pm 0.0169}$ |
| o-ACC ($r = I, \tau = 0.1, v = \frac{1}{3}$) on LR ($w = u, C = 0.001$) | $\mathbf{0.0562 \pm 0.0159}$ |
| o-PACC ($r = C_2, \tau = 0.1, v = \frac{1}{3}$) on LR ($w = u, C = 0.001$) | $\mathbf{0.0569 \pm 0.0138}$ |
| RUN ($\tau = 1.0e-5, J = 60$) | $0.0573 \pm 0.0170$ |
| o-SLD ($o = 0, i = 0.1$) on DT ($w = n, c = G, d = 4$) | $0.0685 \pm 0.0171$ |
| SLD on LR ($w = n, C = 10.0$) | $0.0753 \pm 0.0259$ |
| CC on LR ($w = u, C = 0.01$) | $0.0759 \pm 0.0237$ |
| ACC ($v = \frac{1}{4}$) on LR ($w = u, C = 10.0$) | $0.0798 \pm 0.0254$ |
| PCC on LR ($w = u, C = 0.01$) | $0.0911 \pm 0.0317$ |
| PACC ($v = \frac{1}{2}$) on LR ($w = u, C = 0.01$) | $0.1083 \pm 0.0470$ |
| OQT ($v = \frac{1}{3}$) on LR ($w = u, C = 0.001$) | $0.2239 \pm 0.0554$ |
| ARC ($v = \frac{1}{3}$) on LR ($w = n, C = 10.0$) | $0.2514 \pm 0.0561$ |

Table 46: NMD on OpenMl-Yolanda-OQ, level 4 out of 5

| quantification method | avg. NMD ± stddev. |
|---|---|
| IBU ($o = 0, i = 0.01, J = 60$) | $\mathbf{0.0564 \pm 0.0162}$ |
| o-ACC ($r = I, \tau = 0.1, v = \frac{1}{3}$) on LR ($w = u, C = 0.001$) | $\mathbf{0.0569 \pm 0.0143}$ |
| RUN ($\tau = 1.0e-5, J = 60$) | $0.0583 \pm 0.0163$ |
| o-PACC ($r = C_2, \tau = 0.1, v = \frac{1}{3}$) on LR ($w = u, C = 0.001$) | $0.0590 \pm 0.0132$ |
| o-SLD ($o = 0, i = 0.03$) on LR ($w = n, C = 10.0$) | $0.0733 \pm 0.0244$ |
| SLD on LR ($w = n, C = 0.1$) | $0.0751 \pm 0.0238$ |
| CC on LR ($w = u, C = 0.01$) | $0.0761 \pm 0.0212$ |
| ACC ($v = \frac{1}{4}$) on LR ($w = u, C = 10.0$) | $0.0800 \pm 0.0227$ |
| PCC on LR ($w = u, C = 0.01$) | $0.0917 \pm 0.0304$ |
| PACC ($v = \frac{1}{2}$) on LR ($w = u, C = 0.01$) | $0.1063 \pm 0.0444$ |
| OQT ($v = \frac{1}{3}$) on LR ($w = u, C = 0.001$) | $0.2257 \pm 0.0550$ |
| ARC ($v = \frac{1}{3}$) on LR ($w = n, C = 10.0$) | $0.2518 \pm 0.0580$ |

Table 47: NMD on OpenMl-Yolanda-OQ, level 5 out of 5 (the least smooth)

| quantification method | avg. NMD ± stddev. |
|---|---|
| IBU ($o = 0, i = 0.01, J = 60$) | **0.0575 ± 0.0159** |
| RUN ($\tau = 1.0e-5, J = 60$) | 0.0596 ± 0.0153 |
| o-ACC ($r = I, \tau = 0.1, v = \frac{1}{3}$) on LR ($w = u, C = 0.001$) | 0.0606 ± 0.0149 |
| o-PACC ($r = C_2, \tau = 0.1, v = \frac{1}{3}$) on LR ($w = u, C = 0.001$) | 0.0652 ± 0.0149 |
| o-SLD ($o = 0, i = 0.03$) on LR ($w = n, C = 0.1$) | 0.0702 ± 0.0218 |
| SLD on LR ($w = n, C = 0.1$) | 0.0711 ± 0.0219 |
| CC on LR ($w = u, C = 0.01$) | 0.0768 ± 0.0209 |
| ACC ($v = \frac{1}{4}$) on LR ($w = u, C = 10.0$) | 0.0799 ± 0.0213 |
| PCC on LR ($w = u, C = 0.01$) | 0.0953 ± 0.0289 |
| PACC ($v = \frac{1}{2}$) on LR ($w = u, C = 0.01$) | 0.1007 ± 0.0454 |
| OQT ($v = \frac{1}{3}$) on LR ($w = u, C = 0.001$) | 0.2275 ± 0.0563 |
| ARC ($v = \frac{1}{3}$) on LR ($w = n, C = 10.0$) | 0.2524 ± 0.0612 |

Table 48: NMD on OpenMl-fried-OQ, regular APP

| quantification method | avg. NMD ± stddev. |
|---|---|
| IBU ($o = 0, i = 0.01, J = 180$) | **0.0168 ± 0.0054** |
| RUN ($\tau = 1.0e-5, J = 120$) | 0.0206 ± 0.0059 |
| o-ACC ($r = C_2, \tau = 0.001, v = \frac{1}{3}$) on LR ($w = u, C = 1.0$) | 0.0264 ± 0.0079 |
| o-SLD ($o = 0, i = 0.03$) on DT ($w = n, c = G, d = 6$) | 0.0322 ± 0.0066 |
| CC on LR ($w = u, C = 10.0$) | 0.0330 ± 0.0085 |
| o-PACC ($r = C_2, \tau = 1.0e-5, v = \frac{1}{3}$) on DT ($w = n, c = G, d = 10$) | 0.0350 ± 0.0184 |
| SLD on DT ($w = n, c = G, d = 6$) | 0.0369 ± 0.0090 |
| PCC on LR ($w = u, C = 10.0$) | 0.0410 ± 0.0101 |
| ACC ($v = \frac{1}{4}$) on DT ($w = u, c = E, d = 12$) | 0.0454 ± 0.0211 |
| OQT ($v = \frac{1}{3}$) on LR ($w = u, C = 10.0$) | 0.0566 ± 0.0144 |
| ARC ($v = \frac{1}{3}$) on LR ($w = n, C = 1.0$) | 0.0589 ± 0.0166 |
| PACC ($v = \frac{1}{2}$) on DT ($w = u, c = E, d = 12$) | 0.0614 ± 0.0256 |

Table 49: NMD on OpenMl-fried-OQ, APP-OQ = level 1 out of 5 (the smoothest)

| quantification method | avg. NMD ± stddev. |
|---|---|
| o-PACC ($r = I, \tau = 0.001, v = \frac{1}{3}$) on LR ($w = u, C = 10.0$) | **0.0146 ± 0.0037** |
| IBU ($o = 1, i = 0.01, J = 120$) | **0.0146 ± 0.0041** |
| RUN ($\tau = 1.0e-5, J = 60$) | 0.0161 ± 0.0045 |
| o-ACC ($r = I, \tau = 0.1, v = \frac{1}{3}$) on LR ($w = u, C = 1.0$) | 0.0189 ± 0.0042 |
| CC on LR ($w = u, C = 10.0$) | 0.0243 ± 0.0056 |
| o-SLD ($o = 0, i = 0.1$) on DT ($w = n, c = G, d = 6$) | 0.0282 ± 0.0047 |
| PCC on LR ($w = u, C = 10.0$) | 0.0330 ± 0.0078 |
| SLD on DT ($w = n, c = G, d = 6$) | 0.0373 ± 0.0082 |
| OQT ($v = \frac{1}{3}$) on LR ($w = u, C = 10.0$) | 0.0472 ± 0.0122 |
| ACC ($v = \frac{1}{4}$) on DT ($w = u, c = E, d = 12$) | 0.0482 ± 0.0230 |
| ARC ($v = \frac{1}{3}$) on LR ($w = n, C = 1.0$) | 0.0598 ± 0.0183 |
| PACC ($v = \frac{1}{2}$) on DT ($w = u, c = E, d = 12$) | 0.0659 ± 0.0260 |

Table 50: NMD on OpenMl-fried-OQ, level 2 out of 5

| quantification method | avg. NMD ± stddev. |
|---|---|
| IBU ($o = 1, i = 0.01, J = 180$) | $\mathbf{0.0151 \pm 0.0042}$ |
| RUN ($\tau = 1.0e - 5, J = 120$) | $0.0186 \pm 0.0048$ |
| o-PACC ($r = I, \tau = 0.001, v = \frac{1}{3}$) on DT ($w = n, c = G, d = 10$) | $0.0222 \pm 0.0063$ |
| o-ACC ($r = C_2, \tau = 0.001, v = \frac{1}{3}$) on DT ($w = u, c = G, d = 10$) | $0.0256 \pm 0.0062$ |
| CC on LR ($w = u, C = 10.0$) | $0.0289 \pm 0.0051$ |
| o-SLD ($o = 1, i = 0.03$) on DT ($w = n, c = G, d = 6$) | $0.0311 \pm 0.0061$ |
| PCC on LR ($w = u, C = 10.0$) | $0.0365 \pm 0.0072$ |
| SLD on DT ($w = n, c = G, d = 6$) | $0.0375 \pm 0.0085$ |
| ACC ($v = \frac{1}{4}$) on DT ($w = u, c = E, d = 12$) | $0.0474 \pm 0.0225$ |
| OQT ($v = \frac{1}{3}$) on LR ($w = u, C = 10.0$) | $0.0515 \pm 0.0124$ |
| ARC ($v = \frac{1}{3}$) on LR ($w = n, C = 1.0$) | $0.0592 \pm 0.0173$ |
| PACC ($v = \frac{1}{2}$) on DT ($w = u, c = E, d = 12$) | $0.0643 \pm 0.0255$ |

Table 51: NMD on OpenMl-fried-OQ, level 3 out of 5

| quantification method | avg. NMD ± stddev. |
|---|---|
| IBU ($o = 1, i = 0.01, J = 180$) | $\mathbf{0.0164 \pm 0.0044}$ |
| RUN ($\tau = 1.0e - 5, J = 120$) | $0.0197 \pm 0.0048$ |
| o-PACC ($r = I, \tau = 0.001, v = \frac{1}{3}$) on DT ($w = n, c = G, d = 10$) | $0.0249 \pm 0.0070$ |
| o-ACC ($r = C_2, \tau = 0.001, v = \frac{1}{3}$) on LR ($w = u, C = 1.0$) | $0.0256 \pm 0.0071$ |
| o-SLD ($o = 0, i = 0.03$) on DT ($w = n, c = G, d = 6$) | $0.0319 \pm 0.0064$ |
| CC on LR ($w = u, C = 10.0$) | $0.0324 \pm 0.0052$ |
| SLD on DT ($w = n, c = G, d = 6$) | $0.0370 \pm 0.0091$ |
| PCC on LR ($w = u, C = 10.0$) | $0.0399 \pm 0.0076$ |
| ACC ($v = \frac{1}{4}$) on DT ($w = u, c = E, d = 12$) | $0.0448 \pm 0.0207$ |
| OQT ($v = \frac{1}{3}$) on LR ($w = u, C = 10.0$) | $0.0554 \pm 0.0123$ |
| ARC ($v = \frac{1}{3}$) on LR ($w = n, C = 1.0$) | $0.0590 \pm 0.0181$ |
| PACC ($v = \frac{1}{2}$) on DT ($w = u, c = E, d = 12$) | $0.0608 \pm 0.0256$ |

Table 52: NMD on OpenMl-fried-OQ, level 4 out of 5

| quantification method | avg. NMD ± stddev. |
|---|---|
| IBU ($o = 0, i = 0.0, J = 180$) | $\mathbf{0.0177 \pm 0.0056}$ |
| RUN ($\tau = 1.0e - 5, J = 120$) | $0.0221 \pm 0.0053$ |
| o-ACC ($r = I, \tau = 0.001, v = \frac{1}{3}$) on DT ($w = u, c = G, d = 10$) | $0.0330 \pm 0.0100$ |
| o-SLD ($o = 0, i = 0.01$) on DT ($w = n, c = G, d = 6$) | $0.0335 \pm 0.0073$ |
| o-PACC ($r = C_2, \tau = 1.0e - 5, v = \frac{1}{3}$) on DT ($w = n, c = G, d = 10$) | $0.0338 \pm 0.0162$ |
| CC on LR ($w = u, C = 10.0$) | $0.0362 \pm 0.0051$ |
| SLD on DT ($w = n, c = G, d = 6$) | $0.0369 \pm 0.0091$ |
| PCC on LR ($w = u, C = 10.0$) | $0.0436 \pm 0.0074$ |
| ACC ($v = \frac{1}{4}$) on DT ($w = u, c = E, d = 12$) | $0.0452 \pm 0.0200$ |
| ARC ($v = \frac{1}{3}$) on LR ($w = n, C = 1.0$) | $0.0581 \pm 0.0153$ |
| PACC ($v = \frac{1}{2}$) on DT ($w = u, c = E, d = 12$) | $0.0592 \pm 0.0241$ |
| OQT ($v = \frac{1}{3}$) on LR ($w = u, C = 10.0$) | $0.0597 \pm 0.0119$ |

Table 53: NMD on OpenMl-fried-OQ, level 5 out of 5 (the least smooth)

| quantification method | avg. NMD ± stddev. |
| --- | --- |
| IBU ($o = 0, i = 0.0, J = 180$) | **0.0202 ± 0.0062** |
| RUN ($\tau = 1.0e-5, J = 120$) | 0.0258 ± 0.0057 |
| o-SLD ($o = 0, i = 0.01$) on DT ($w = n, c = G, d = 6$) | 0.0335 ± 0.0083 |
| o-ACC ($r = I, \tau = 0.001, v = \frac{1}{3}$) on DT ($w = u, c = G, d = 10$) | 0.0345 ± 0.0095 |
| SLD on DT ($w = n, c = G, d = 6$) | 0.0356 ± 0.0098 |
| o-PACC ($r = C_2, \tau = 1.0e-5, v = \frac{1}{3}$) on DT ($w = n, c = G, d = 10$) | 0.0366 ± 0.0157 |
| ACC ($v = \frac{1}{4}$) on DT ($w = u, c = E, d = 12$) | 0.0415 ± 0.0187 |
| CC on LR ($w = u, C = 10.0$) | 0.0432 ± 0.0065 |
| PCC on LR ($w = u, C = 10.0$) | 0.0518 ± 0.0089 |
| PACC ($v = \frac{1}{2}$) on DT ($w = u, c = E, d = 12$) | 0.0569 ± 0.0259 |
| ARC ($v = \frac{1}{3}$) on LR ($w = n, C = 1.0$) | 0.0584 ± 0.0135 |
| OQT ($v = \frac{1}{3}$) on LR ($w = u, C = 10.0$) | 0.0693 ± 0.0122 |