An Efficient Method for Deriving Confidence Intervals in Aggregative Quantification

Alejandro Moreo[1] (✉) and Nicola Salvati[2]

[1] Istituto di Scienza e Tecnologie dell'Informazione, Consiglio Nazionale delle Ricerche, Pisa, Italy alejandro.moreo@isti.cnr.it
[2] Dipartimento di Economia e Management, Università di Pisa, Pisa, Italy nicola.salvati@unipi.it

Abstract. This paper explores efficient methods for deriving confidence intervals in quantification, the area of machine learning concerned with estimating class prevalence values. By focusing on computationally efficient strategies, we propose a robust framework for quantifying uncertainty. The key idea is to disentangle the two main phases of current aggregative quantifiers (classification followed by aggregation) and apply bootstrap only to the second phase. We investigate different methods for constructing confidence regions, including confidence intervals, confidence ellipses in the simplex, and confidence regions in the transformed Centered Log-Ratio space. Additionally, we examine various bootstrap strategies, including model-based, population-based, and a combined approach. Our results demonstrate the effectiveness of combining model-based and population-based bootstrap approaches, particularly when used with traditional confidence intervals, while also achieving significant efficiency gains compared to a naive application of bootstrap.

Keywords: Confidence intervals · Class prevalence estimation · Quantification.

1 Introduction

Many disciplines, like the social sciences [16], epidemiology [18], ecological modelling [1], and sentiment analysis [12], are interested in knowing the class prevalence distribution of the population under study. Quantification is a key branch of supervised machine learning that focuses on estimating the prevalence of class labels in a population [13,7]. Such methods are called "quantifiers" and, just like with any other point estimators, their predictions are inevitably affected by error. It would therefore be desirable to accompany the outputs of a quantifier with a measure of *uncertainty*, in the form of confidence regions [24].

A standard method used in statistics for deriving confidence regions is *bootstrap* [6]. Bootstrap is a resampling method used to estimate the distribution of a statistic by repeatedly sampling with replacement from the observed data. It allows for the construction of confidence regions by calculating the statistic of interest (in our case, the sample prevalence) across multiple resampled bags and

determining the region that captures the desired percentage of the resampled estimates (e.g., 95% of these).

In this report, we analyse how to apply bootstrap to quantification. In particular, we focus our attention to one class of quantifiers called *aggregative quantifiers* (see §4.2 in [7]), which rely, in order to predict class prevalence values, on the outputs of a surrogate classifier. This family of methods seems appealing for our scope since the procedure follows a pipeline of two steps: the learning phase comes down to (i) training a classifier which operates at the individual level, and (ii) fitting an *adjustment* method which operates at the aggregate level; while the inference phase can be subdivided into (i') generating classifier predictions for each test instance, and (ii') applying the adjustment function to the sample to produce prevalence estimates. Steps (i') and especially (i) are computational heavier than steps (ii') and (ii). Note thus that this two-step process is convenient for deriving confidence intervals via bootstrap since it offers us the possibility to carry out steps (i) and (i') only once, and apply bootstrap only in steps (ii) and (ii'), thus substantially speeding up the otherwise unaffordable computational cost.

The rest of this report is structured as follows. In Section 2 we overview past work on confidence intervals in quantification. Section 3 is devoted to explaining our application of bootstrap to aggregative quantifiers. Section 4 reports the experiments we have carried out while Section 5 wraps up and offers pointers for future research.

2 Related Work

The use of bootstrap in quantification is not novel. Probably, the first published record discussing the application of bootstrap for deriving confidence intervals in quantification is by [16]. However, the authors only mention the application of "standard bootstrapping procedures" without providing specific clues on how they apply it.

[17] proposes a generative probabilistic model for text quantification which assumes a document's text is generated conditional on the document label. As the generative model the authors explore multinomial Naïve Bayes and additive log-linear methods. Inference is carried out via marginal log-likelihood over the priors. While this paper uses real-world data in the experiments, these are restricted to binary quantification and textual data.

[4] propose a method called "Error-Adjusted Bootstrapping" which also relies on bootstrap resampling. The idea is to use a (crisp) classifier to issue label predictions for all the instances of the bootstrap samples. However, these labels are not directly used to obtain bootstrap prevalence predictions by simply averaging. Instead, the authors propose to randomly sample the predicted instance label based on the misclassification-rates distribution $P(Y|\hat{Y})$ that characterizes the classifier, which is modelled on training data. However, as noted by [24], this method is not appropriate for constructing confidence intervals in the presence of prior probability shift (which is the focus of our work –Section 3.1). The

reason is that the conditional distribution $P(Y|\hat{Y})$ is not stationary under such conditions. Also [4] limits the experimental section to binary quantification.

[24] carries out a simulation study in which several quantifiers are tested in their ability to derive confidence intervals and prediction intervals. This paper also clarifies that there is a distinction between confidence intervals and prediction intervals [19] which is often neglected in the literature. Be it as it may, the author himself raises questions on the necessity of pushing forth this distinction in practical applications of quantification. However, this study is limited to binary quantification only, and to the idealizations typical of simulated conditions that may obscure real-world complexities of the data.

Other works exist which tries to derive confidence intervals analytically. One example is by [9], which propose the "ratio estimator" method for quantification. The authors derive confidence intervals by analysing the asymptotic properties of the method, thus avoiding iterative resampling procedures. As a downside, the method is specific for the ratio estimator quantifier. In contrast, the method we propose is generic and applicable to any of the "aggregative" methods described in the quantification literature (by far, the largest class of methods in the field).

[5] shows that the Probabilistic Classify and Count method (PCC), a method that predicts prevalence values by averaging across the posterior probabilities on each data point that a probabilistic classifier generates [2], can be endowed with confidence intervals. The idea resides in treating this point estimate as the mean of a Poisson binomial distribution of the posterior probabilities, for which confidence intervals can be computed easily. However, PCC is known to be a weak quantifier under prior probability shift conditions [14].

Recently, a (tractable) Bayesian approach for deriving confidence intervals around prevalence point estimators based on black-box classifiers is presented in [25]. While this method caters for multiclass problems, it also suffers from a high computationally cost since the method relies on Hamiltonian Markov Chain Monte Carlo sampling.

3 Method

In this section, we turn to describe a methodology for applying bootstrap to aggregative quantification methods. First, we set down the notation we use.

3.1 Notation

Let $q : \mathbb{N}^{\mathcal{X}} \to \Delta^{n-1}$ be a function (a *quantifier*) that maps bags of elements from the input space $\mathcal{X}$ into class label distributions, i.e., into elements of the probability simplex $\Delta^{n-1} \equiv \{(\pi_1, \ldots, \pi_n) \mid \pi_i \geq 0, \sum_{i=1}^{n} \pi_i = 1\}$ where n is the number of classes and π_i represents the proportion of elements from the bag that belong to class i, with $\mathcal{Y} = \{1, \ldots, n\}$ the classes of interest.

Given an unlabelled test set U with (unknown) prevalence $\boldsymbol{\pi}^* \in \Delta^{n-1}$, we are interested in training a quantifier q such that $q(U) = \hat{\boldsymbol{\pi}}$ is a good approximation of $\boldsymbol{\pi}^*$ in terms of any given divergence metric.

To this aim, we assume to have access to a labelled training set $L = \{(x_i, y_i)\}_{i=1}^{l}$ of l labelled datapoints $x_i \in \mathcal{X}$, $y_i \in \mathcal{Y}$. We also consider a learner device $\Psi : (\mathcal{X} \times \mathcal{Y})^l \to \mathcal{Q}$ that takes as input a training set and generates (i.e., learns) a quantifier function $q \in \mathcal{Q}$ from a class of quantification functions $\mathcal{Q}$.

We assume to be in the presence of prior probability shift (PPS), a type of dataset shift in which the training distribution P and the test distribution Q are different, but in which the following assumptions are made:

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

where Y is the random variable taking values on the output space $\mathcal{Y}$ and X is the random variable taking values in the input space $\mathcal{X}$.

3.2 Bootstrap

This work aims to provide not only a point estimate but also a confidence region around it. *Bootstrap* is a standard and versatile method for constructing confidence regions by resampling data [6]. The idea is to uniformly draw a series of samples $U_1, \ldots, U_m$ from U, with replacement and with $|U_i| = |U|$, and to use q to generate the corresponding predictions $\hat{\boldsymbol{\pi}}_1 = q(U_1), \ldots, \hat{\boldsymbol{\pi}}_m = q(U_m)$, which are then used to define the confidence region around the final point estimate which is finally computed as $\underline{\hat{\boldsymbol{\pi}}} = \frac{1}{m}\sum_{i=1}^{m} \hat{\boldsymbol{\pi}}_i$. This way of deriving confidence intervals based on resampling the test set is called the *population-based* approach.

A different way for deriving confidence regions is the so-called *model-based* approach, in which the resampling is instead applied to the training set L. This way, a series of training samples $L_1, \ldots, L_{m'}$ is generated from L, with replacement and with $|L_i| = |L|$. The series of point estimates are then computed as $\hat{\boldsymbol{\pi}}_1 = q_1(U), \ldots, \hat{\boldsymbol{\pi}}_{m'} = q_{m'}(U)$, where $q_i = \Psi(L_i)$. The final estimate is computed in the usual way: $\underline{\hat{\boldsymbol{\pi}}} = \frac{1}{m'}\sum_{i=1}^{m'} \hat{\boldsymbol{\pi}}_i$, and the series of estimates $\hat{\boldsymbol{\pi}}_1, \ldots, \hat{\boldsymbol{\pi}}_{m'}$ is used to derive confidence regions around $\underline{\hat{\boldsymbol{\pi}}}$.

Yet a third method for deriving confidence regions comes down to combining the population-based approach with the model-based approach. In this way, we generate a grid of $m \times m'$ point estimates $G = [g_{ij}] = q_j(U_i)$ where $q_j = \Psi(L_j)$. The point estimate is thus computed as $\underline{\hat{\boldsymbol{\pi}}} = \frac{1}{mm'}\sum_{i=1}^{m}\sum_{j=1}^{m'} g_{ij}$, and all the estimates in G concur in deriving a confidence region around $\underline{\hat{\boldsymbol{\pi}}}$.

Bootstrap is Quantification Experiments Due to repeated resampling, bootstrap methods incur a considerable computational cost, specially when a model-based approach is adopted. A typical value for m or m' is 500 repetitions. Generating 500 predictions for a test set is already costly (population-based approach), but the problem is even more exacerbated if the number of trainings rounds increases to 500 (model-based approach). When a combination of population-based and model-based approaches are adopted, the prohibitive number of combinations 500×500 is often reduced to 100×100. Still, the cost seems unaffordable from a computational point of view.

The latter is especially true in a discipline like quantification, in which the evaluation of a model often consists of confronting the quantifier against *many* test samples. The reason is that one test set counts as only one test *instance* for a quantifier; this is in contrast to other disciplines like classification, or regression, in which one test set represents a number of test instances that equals the cardinality of the set.

Quantification papers do often resort to sampling protocols to generate, out of a test sample, many samples each characterized by (possibly radically) different prevalence values. The number of test samples thus generated is often high in order to allow for statistically significant results. Typical values encountered in the quantification literature easily reach 100 samples [20], 1000 samples [21], or even more [8,22]. The aforementioned "resampling" cost that bootstrap incurs (500, or 100×100 repetitions, depending on the case) must therefore be undertaken *for each test sample* extracted from a dataset and *for each dataset* under consideration (modern papers involve dozens of datasets [22,23,21]). With this landscape, the adoption of bootstrap in quantification clearly seems impractical.

Bootstrapping Aggregative Quantifiers In this study, we propose a variant of bootstrap applied on a specific class of quantification methods called *aggregative* (see §4.2 in [7]). These methods are such that the training procedure can be disentangled as follows:

1. Learn a crisp (resp. soft) classifier $h \in \mathcal{H}$, i.e., a functional $h : \mathcal{X} \to \mathcal{Y}$ (resp. $h : \mathcal{X} \to \Delta^{n-1}$) using a learner device $\mathcal{C} : (\mathcal{X} \times \mathcal{Y})^l \to \mathcal{H}$ and a training set L; then use h to generate classifier predictions for each element in the training set via cross-validation, thus obtaining $\tilde{L} = \{(h(x_i), y_i) : (x_i, y_i) \in L\}$.
2. Learn an *adjustment* function $A : (\mathcal{Y} \times \mathcal{Y})^l \to \Delta^{n-1}$ (resp. $A : (\Delta^{n-1} \times \mathcal{Y})^l \to \Delta^{n-1}$) using the classifier's outputs L. The adjustment function is responsible for generating, out of the classifier predictions, the final class prevalence estimates.

Similarly, the inference procedure can be decomposed as:

1. Use h to generate classifier predictions for all instances in the test set $\tilde{U} = \{h(x_i) : x_i \in U\}$.
2. Return the prevalence values generated via $A(\tilde{U})$.

Note that the first step of the training phase, and the first step of the inference phase, are often much more expensive, from a computational point of view, than the corresponding steps number 2. We take advantage of this characteristic and propose to apply:

- Model-based bootstrap:
  - During training, carry out step 1 once and for all, and apply resampling only during step 2, i.e., generating many samples to $\tilde{L}_1, \ldots, \tilde{L}_{m'}$ out of $\tilde{L}$, thus obtaining a series of adjustment functions $A_1, \ldots, A_{m'}$.

  - During inference, generate $\tilde{U}$ (step 1) and generate a series of point estimates $A_1(\tilde{U}), \ldots, A_{m'}(\tilde{U})$ applying step 2 on every adjustment function.
- Population-based bootstrap:
  - During training, run steps 1 and 2 of the training procedure the usual way.
  - During inference, generate all classifier predictions (step 1) only once, and then apply resampling over $\tilde{U}$ for step 2, thus generating a series of bags $\tilde{U}_1, \ldots, \tilde{U}_m$ with which a series of point estimates $A(\tilde{U}_1), \ldots, A(\tilde{U}_m)$ is computed.
- Combined model-based and population-based approach: a trivial combination of the above methods.
  - During training, carry out step 1 once and for all, and apply resampling only during step 2, i.e., generating many samples to $\tilde{L}_1, \ldots, \tilde{L}_{m'}$ out of $\tilde{L}$, thus obtaining a series of adjustment functions $A_1, \ldots, A_{m'}$.
  - During inference, generate all classifier predictions (step 1) only once, and then apply resampling over $\tilde{U}$ for step 2, thus generating a series of bags $\tilde{U}_1, \ldots, \tilde{U}_m$. Generate a grid of point estimates pairing every A_i $(1 \leq i \leq m')$ adjustment function with every $\tilde{U}_j$ $(1 \leq j \leq m)$ bag.

3.3 Methods for Constructing Confidence Regions

In Section 3.2 we have discussed three bootstrap-based techniques for obtaining a series of point estimates. Regardless of the method used to generate the series, we will henceforth denote these series as $\hat{\boldsymbol{\pi}}_1, \ldots, \hat{\boldsymbol{\pi}}_m$, where m represents the total number of estimates, assuming an abuse of notation for simplicity. In this section, we turn to discuss methods for constructing confidence regions based on these estimates. All confidence regions are constructed around the final estimate $\underline{\hat{\boldsymbol{\pi}}} = \frac{1}{m}\sum_{i=1}^{m} \hat{\boldsymbol{\pi}}_i$ with a certain confidence value α. We will denote CR_α a generic confidence region constructed around the mean estimate at confidence level α.

Confidence Intervals Perhaps the simplest way to derive confidence regions are the confidence intervals (CI) generated via bootstrap quantiles.

This method is simple and non-parametric. The idea is to consider each class prevalence as an independent random variable of unknown distribution, and derive their confidence intervals using the empirical distribution generated via bootstrap.

We will therefore consider the (independent) random variable P_i which takes on values $[\hat{\boldsymbol{\pi}}_1]_i, \ldots, [\hat{\boldsymbol{\pi}}_m]_i$, where $[\cdot]_i$ is the operator that returns the prevalence of the ith class in the input vector.

For each such random variable we arrange the bootstrap estimates in ascending order to obtain their empirical distribution, and identify the lower ($\mathbf{L}_i$) and upper ($\mathbf{U}_i$) bounds at confidence $(1-\alpha)\%$ (e.g., at 95%) as the $\left(\frac{\alpha}{2}\right)$ percentile (e.g., 2.5%) and the $\left(100-\frac{\alpha}{2}\right)$ percentile (e.g., 97.5%) of the sorted distribution. For instance, given a dataset with a prevalence estimate of 0.65 and a

bootstrap-derived distribution of prevalence values ranging from 0.55 to 0.75, the 95% confidence interval would be [0.56, 0.74].

Given the bounds $\mathbf{L}_i, \mathbf{U}_i$ $(1 \leq i \leq n)$ for all classes, we can determine whether a prevalence vector $\boldsymbol{\pi} = \{\pi_1, \ldots, \pi_n\}$ lies within the confidence region by computing:

$$\mathrm{CI}_\alpha(\boldsymbol{\pi}) = \begin{cases} 1: & \mathbf{L}_i \leq \pi_i \leq \mathbf{U}_i \ , \forall i, 1 \leq i \leq n \\ 0: & \text{otherwise} \end{cases} \tag{1}$$

Despite its simplicity, this method presents some drawbacks. First, it assumes the random variables are independent; however, we know the random variables take on values drawn from a more complex structure, the probability simplex, which imposes certain contraints. Second, the confidence region forms a hyper-rectangle (with each dimension representing the confidence interval of one variable) which intersects but also extends outside the probability simplex (see Section 3.4).

Confidence Ellipse in $\boldsymbol{\Delta^{n-1}}$ A typical procedure for deriving confidence regions in multivariate spaces is to define a confidence (hyper-)ellipse (CE) around the point estimate $\underline{\hat{\boldsymbol{\pi}}}$.

A typical procedure for checking whether a prevalence vector $\boldsymbol{\pi}$ belongs to the confidence ellipse around $\underline{\hat{\boldsymbol{\pi}}}$ comes down to assuming $\hat{\boldsymbol{\pi}}_i \sim \mathcal{N}(\boldsymbol{\mu}, \Sigma)$, with $\boldsymbol{\mu}$ the (unknown) population mean which is estimated as $\underline{\hat{\boldsymbol{\pi}}}$, and Σ the (unknown) population covariance matrix which is estimated with the sample covariance matrix S, and then check whether:

$$\mathrm{CE}_\alpha(\boldsymbol{\pi}) = \begin{cases} 1: & (\boldsymbol{\pi} - \underline{\hat{\boldsymbol{\pi}}})^\top S^{-1} (\boldsymbol{\pi} - \underline{\hat{\boldsymbol{\pi}}}) \leq \chi^2_{(n-1)}(1-\alpha) \\ 0: & \text{otherwise} \end{cases} \tag{2}$$

where χ^2 is the chi-squared distribution, S^{-1} is the inverse of the covariance matrix S (aka the *precision* matrix), and α is the desired confidence level (e.g., $\alpha = 0.05$ for 95%).

While this method guarantees that the ellipse lies on the probability simplex (more on this in Section 3.4), it rests upon the incorrect assumption that the prevalence estimates are normally distributed. We know all our estimates lie on the probability simplex, which clashes with this assumption.

Confidence Ellipse in the CLR-space Confidence intervals and confidence ellipses do not take into account the inner geometry of the probability simplex space (i.e., all components must be positive and sum up to one). A more sophisticated way to derive confidence regions comes down to mapping the prevalence estimates onto an unconstrained Euclidean space using the Centered Log-Ratio (CLR) transformation. We call CT this confidence region in the transformed space. The CLR transformation $T : \Delta^{n-1} \to \mathbb{R}^n$ is given by:

$$T(\boldsymbol{\pi}) = \left[\log \frac{\pi_1}{g(\boldsymbol{\pi})}, \ldots, \log \frac{\pi_n}{g(\boldsymbol{\pi})}\right]$$

where $g(\boldsymbol{\pi})$ is the the geometric mean of $\boldsymbol{\pi}$.

We can define the confidence ellipse around the new center $\underline{\boldsymbol{\pi}}' = \frac{1}{m}\sum_{i=1}^{m} T(\boldsymbol{\pi}_i)$ as the region containing all (CLR-transformed) points verifying:

$$\mathrm{CT}_\alpha(\boldsymbol{\pi}) = \begin{cases} 1: & (T(\boldsymbol{\pi}) - \underline{\boldsymbol{\pi}}')^\top (S')^{-1} (T(\boldsymbol{\pi}) - \underline{\boldsymbol{\pi}}') \leq \chi^2_{(n-1)}(1-\alpha) \\ 0: & \text{otherwise} \end{cases} \tag{3}$$

where S' is the sample covariance matrix of the CLR-transformed estimates $T(\boldsymbol{\pi}_1), \ldots, T(\boldsymbol{\pi}_n)$, and χ^2 is the chi-squared distribution, as before. Note that the degrees of freedom of the distribution is still $(n-1)$ since the original components are constrained.

3.4 Goodness of the Confidence Region

Confidence regions are typically evaluated in terms of the *coverage* ($\mathcal{C}$) and their *amplitude* ($\mathcal{A}$).

The coverage measures the fraction of experiments in which the true prevalence value was contained in the confidence region (the higher, the better), and is simply computed as:

$$\mathcal{C} = \frac{\sum_{i=1}^{E} \mathrm{CR}_\alpha(\boldsymbol{\pi}_i^*)}{E} \tag{4}$$

with E the total number of experiments, and CR_α any of the regions discussed in Section 3.3 (Equations 1, 2, and 3), independently generated for the training set L and test set U of each experiment.

The amplitude measures the portion of the probability simplex that is included in the confidence region (the smaller, the better), and is computed as:

$$\mathcal{A} = \frac{V(\mathrm{CR}_\alpha)}{V(\Delta^{n-1})} \tag{5}$$

with V a function measuring the *volume* of a space.

While computing the volume of the simplex is trivial, and comes down to calculating $V(\Delta^{n-1}) = \frac{1}{(n-1)!}$, computing the volume of a confidence region is not that easy. For example, one might be tempted to compute $V(\mathrm{CI}_\alpha)$ simply as $\prod_{i=1}^{n}(\mathbf{U}_i - \mathbf{L}_i)$. However, this would be misleading, since the volume we are computing corresponds to a region that does not lie within the simplex, but in $[0,1]^n$ (the region intersects with the simplex, but covers also part of the space outside the simplex).

Similarly, one might be tempted to compute

$$V(\mathrm{CE}_\alpha) = \frac{\pi^{\frac{n-1}{2}}}{\Gamma(\frac{n-1}{2}+1)} \prod_{i=1}^{n-1} \sqrt{\lambda_i \cdot \chi^2_{n-1}(1-\alpha)} \tag{6}$$

with π the pi constant (not to be confused with a prevalence value), λ_i the ith eigenvalue of the covariance matrix.[3] Again, this would be incorrect. The reason is that a confidence ellipse generated from points on a simplex may not be entirely confined within the simplex. While the centre of the ellipse is guaranteed to lie within the simplex, the ellipse's edges could well extend beyond the simplex's boundaries; see Figure 1.

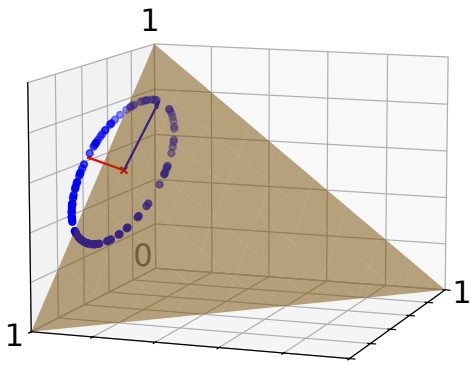

Fig. 1: Ellipse in a 2-dimensional simplex. The center is contained in the simplex, but part of the ellipse lies outside it.

Finally, note that $V(\mathrm{CT}_\alpha)$ would effectively account for the volume of the ellipse in the CLR-space, but there is no simple way to transform back such scalar to the original space.

We therefore propose a simple, yet effective method, to approximate the amplitude, based on Monte Carlo sampling. To this end, we generate $t = 10,000$ random prevalence points from the probability simplex, uniformly at random, and compute the fraction of these that belong to the region. That is:

$$\mathcal{A} \approx \frac{\sum_{i=1}^{t} \mathrm{CR}_\alpha(\tilde{\boldsymbol{\pi}}_i)}{t}, \tilde{\boldsymbol{\pi}}_i \sim \mathrm{Dir}(\mathbf{1}_n) \tag{7}$$

with $\mathbf{1}_n$ an n-dimensional vector of ones, the concentration parameter of the Dirichlet distribution. This method is generic, and is valid for CI_α, CE_α, and CT_α.

4 Experiments

In this section, we turn to describe the experiments we have carried out in order to evaluate our methods for deriving confidence regions.

[3] Only $(n-1)$ such eigenvalues are considered, since the last one is 0, given that the ellipse lies in an $(n-1)$-dimensional space, and thus has one null semiaxis corresponding to the nth dimension.

4.1 Datasets

The datasets we have used are taken from the publicly available UCI Machine Learning repository[4], and consists of 26 binary datasets and 24 multiclass datasets. The statistics of these datasets can be consulted online.[5]

For each dataset, we randomly extract 70% of the data for training our quantifiers, and use the remaining 30% for testing them. In the few cases in which the training set exceeds 25,000 instances, we randomly use 25,000 for training and the rest for testing.

Each value reported in the tables is the aggregation of 100 test samples of 500 instances each, drawn at random with the artificial prevalence protocol (APP) described in [22]. This protocol works by uniformly sampling prevalence values from the probability simplex, and then extracting, from the test pool, bags realizing these prevalence values. The APP protocol thus generates samples at widely varying prevalence values effectively representing cases of prior probability shift.

4.2 Evaluation metrics

In order to evaluate the quality of the quantifiers' predictions, we adopt the Absolute Error (AE) of the prevalence predictions with respect to the true prevalence values. AE is given by:

$$\mathrm{AE}(\boldsymbol{\pi}^*, \hat{\boldsymbol{\pi}}) = \frac{1}{n}\sum_{i=1}^{n} |\pi_i^* - \hat{\pi}_i|$$

we report the Mean Absolute Error (MAE) across all experiments.

We evaluate the quality of the confidence regions by means of the coverage ($\mathcal{C}$, see Equation 4), i.e., the proportion of experiments in which the true prevalence was contained in the confidence region, and the amplitude ($\mathcal{A}$, see Equation 7), a measure of the proportion of the simplex' volume that the region includes; see Section 3.4 for further details.

A good confidence region is one that displays high coverage and small amplitude. There is, to the best of our knowledge, no standard measure for combining both metrics into a single one. We propose a measure of goodness that we call Log-Ratio Goodness (LRG) and that we define as:

$$\mathrm{LRG}(\mathrm{CR}_\alpha) = \log\left(1 + \frac{\mathcal{C}}{\mathcal{A} + \epsilon}\right) \tag{8}$$

with ϵ a small value to prevent zero division in degenerated null regions.

In all cases, we consider confidence regions at a confidence level of $\alpha = 0.05$.

[4] `https://archive.ics.uci.edu`

[5] `https://hlt-isti.github.io/QuaPy/manuals/datasets.html#uci-machine-learning`

4.3 Quantification methods

We choose three representative aggregative quantifiers for testing the confidence regions (but any aggregative quantifier can be used in place):

- ACC: Adjusted Classify and Count [11] uses a crisp classifier h to compute an estimate of the class priors (by classifying and counting), which is then corrected by taking into consideration the matrix of the misclassification rates of the classifier M_h. The adjustment procedure (in which model-based bootstrap is applied) thus amounts to constructing the misclassification rates matrix M_h, while the aggregation phase (in which the population-based bootstrap is applied) consists of correcting the estimate by inverting the misclassification rates matrix.
- PACC: Probabilistic Adjusted Classify and Count [2] is the probabilistic counterpart of ACC, in which the classifier h is a soft one, and in which the counts and the misclassification rates matrix are computed on soft counts (posterior probabilities).
- DM: Distribution Matching is a multiclass extension, as proposed by [10,3], of the Hellinger Distance y (HDy) method proposed by [15]. In a nutshell, DM-based approaches try to reconstruct the distribution of the test datapoints by looking for the closest mixture of the class-conditional distributions of the training datapoints. Distributions are modelled as independent univariate histograms. The mixture parameter yielding the best such matching, in terms of the HD is an estimate of the sought class prevalence values.

In all cases we use Logistic Regression as the underlying classifier. Logistic Regression has become very popular in the quantification literature for its ability to produce well-calibrated posterior probabilities and due to the fact that is has shown good performance across a large variety of problems, a fundamental aspect for obtaining good confidence regions [24].

We took the implementations of these methods made available in the `QuaPy` software package [20]. Our implementations of the confidence regions and bootstrap methods as well as all the required scripts to reproduce our experiments can be accessed via GitHub.[6] The method has been integrated into QuaPy and will be made available in the upcoming v0.2.0; a pre-release is available at the `devel` branch [7].

4.4 Results

In this section we report on the results we have obtained. In the tables to come we use the following conventions. Each table reports the quantification errors (MAE, lower is better), accompanied by the coverage ($\mathcal{C}$, higher is better) and their amplitude ($\mathcal{A}$, lower is better); bold indicates the best value in each case. A summary of all the binary and multiclass results is reported in Tables 1 and

[6] `https://github.com/AlexMoreo/ConfidenceIntervalsQuantification`
[7] `https://github.com/HLT-ISTI/QuaPy/blob/devel/quapy/method/confidence.py`

Table 1: Summary of binary results grouped by bootstrap and confidence region. Symbol ↑ marks metrics for which the higher the better, while ↓ indicates the lower the better.

Bootstrap	Conf. region	MAE ↓	$\mathcal{C}$ ↑	$\mathcal{A}$ ↓	LRG ↑
None	None	0.057	–	–	–
population-based	CI	0.055	0.710	**0.137**	1.617
	CE	0.055	0.643	0.202	1.298
	CT	0.055	0.601	0.176	1.221
model-based	CI	**0.054**	**0.871**	0.210	**1.767**
	CE	**0.054**	0.744	0.255	1.363
	CT	**0.054**	0.662	0.263	1.147
combined	CI	0.055	**0.871**	0.211	**1.767**
	CE	0.055	0.747	0.258	1.357
	CT	0.055	0.627	0.245	1.096

Table 2: Summary of multiclass results grouped by bootstrap and confidence region. Symbol ↑ marks metrics for which the higher the better, while ↓ indicates the lower the better.

Bootstrap	Conf. region	MAE ↓	$\mathcal{C}$ ↑	$\mathcal{A}$ ↓	LRG ↑
None	None	0.054	–	–	–
population-based	CI	0.053	0.560	**0.033**	5.148
	CE	0.053	0.611	0.052	4.666
	CT	0.053	0.641	0.093	4.083
model-based	CI	**0.048**	0.723	0.076	5.765
	CE	**0.048**	0.772	0.100	5.141
	CT	**0.048**	**0.807**	0.209	4.145
combined	CI	**0.048**	0.726	0.076	**5.769**
	CE	**0.048**	0.753	0.093	4.833
	CT	**0.048**	0.789	0.205	3.809

2, in which all methods and datasets have been aggregated for the reader's convenience. The detailed tables displaying the individual performance of each quantification method in each dataset can be consulted in the additional material.

These results reveal some interesting findings. First, that in most cases, the point estimate generated by averaging across all bootstrap estimates often improves over the raw estimate obtained by the original method, which directly provides one single estimate. However, this is not always true if we take a closer look at the individual experiments (see the additional material), and there is one case (DM in the binary setting) where the original estimate is better, in terms of MAE, than those provided through bootstrap. In other cases, despite being worse, the results are statistically similar (3 cases out of 6).

Regarding the different ways for applying bootstrap (population-based, model-based, or combined), it seems that, in terms of MAE, there are no significant differences on the quality of the final estimates in the binary setup. However, the model-based and the combined approach seems to deliver slightly better results in the multiclass case, although the differences in performance are not statistically significant.

In terms of coverage ($\mathcal{C}$), the best bootstrap strategy seems to be the model-based one, while there is no clear winner within the various strategies for deriving confidence regions. What clearly emerges from the results is that there is a strict order CI $\succ$ CE $\succ$ CT in the binary setup (with A $\succ$ B meaning "A outperforms B"), while this order is reversed in the multiclass case.

While it may seem counter-intuitive that the coverage consistently falls below the nominal level of $1 - \alpha = 0.95$, this stems from a mismatch between the assumed distributional framework (prior probability shift) and the i.i.d. assumption underlying the standard bootstrap. As a result, resampling methods such as the bootstrap must be interpreted with caution. In practice, the bootstrap can approximate the variability of a quantifier with respect to the observed test sample, but it does not account for the additional uncertainty induced by distributional shift. In other words, it fails to fully reproduce the different sources of variability. Bootstrap intervals therefore reflect the internal stability of the quantifier given the available data, but they should not be interpreted as providing a formal coverage guarantee for the true prevalence. We are currently investigating alternative bootstrap schemes aimed at incorporating distributional shift into the variability estimation, with the goal of achieving more reliable coverage properties.

In terms of amplitude ($\mathcal{A}$) we observe a tendency towards CI $\succ$ CE $\succ$ CT. While there are few exceptions to this, CI always stands out in its ability to display low amplitude.

The combined evaluation of confidence regions LRG clearly indicates CI is the top performing method for deriving confidence regions with good coverage and small amplitude, simultaneously.

Overall, the experiments we have carried out seem to indicate the best strategy for generating confidence regions around the point estimate is the approach that combines both population-based and model-based bootstrap methods, paired with confidence intervals.

4.5 Execution times

A novel aspect of our approach resides in the fact that bootstrap is only applied to the pre-classified instances that concur in training the aggregation function (thus avoiding the cost of training different classifiers), and/or to pre-classified instances that are given as input to the aggregation function (thus avoiding the cost of classifying each bootstrap sample).

In this section, we report clocked times to showcase the improvements in efficiency that our proposal brings about with respect to a naive implementation of bootstrap in quantification, i.e., with respect to an implementation that, for

Table 3: Summary of clocked times in binary experiments grouped by bootstrap method.

Bootstrap	Implem.	Train time (s)	(%Rel.Red)	Test time (s)	(%Rel.Red)
None	`QuaPy`	0.023	–	0.003	–
population-based	Naive	0.023	–	1.681	–
	Ours	0.023	(=)	1.311	(-22.02%)
model-based	Naive	11.461	–	1.681	–
	Ours	0.287	(-97.50%)	1.360	(-19.10%)
combined	Naive	2.292	–	33.615	–
	Ours	0.073	(-96.83%)	27.404	(-18.48%)

Table 4: Summary of clocked times in multiclass experiments grouped by bootstrap method.

Bootstrap	Implem.	Train time (s)	(%Rel.Red)	Test time (s)	(%Rel.Red)
None	`QuaPy`	0.638	–	0.010	–
population-based	Naive	0.638	–	4.798	–
	Ours	0.638	(=)	4.666	(-2.75%)
model-based	Naive	318.760	–	4.798	–
	Ours	3.742	(-98.83%)	4.509	(-6.03%)
combined	Naive	63.752	–	95.962	–
	Ours	1.535	(-97.59%)	93.381	(-2.69%)

every bootstrap sample generated in a model-based approach, it undergoes the total cost associated with training a classifier and an aggregation function, and that for every bootstrap sample generated in a population-based approach, it undergoes the total cost of classifying all instances and aggregating them. We call this implementation "Naive".

Tables 3 and 4 report the times we have clocked in on a desktop computer equipped with a 12th Gen Intel(R) i9-12900K processor and 64GB of RAM, running Ubuntu 22. Recall that, for these experiments, we are resampling 500 times during training with the model-based approach, 500 times during test with the population-based; for the combined approach we resample 100 times during training and 100 times during test (thus predicting 100×100=10,000 estimates). We also display the percentage of relative reduction with respect to the naive approach.

Overall, these tables clearly show that there are important gains in efficiency with respect to a naive implementation. These gains are more marked for the model-based and combined training times. The reason is that training the classifier is the most time-consuming task, which is carried out once and for all in our implementation. There are significant reductions also at inference time; however, these reductions appear less marked than the training ones, since the

classification of the instances (that our methods undergoes only once) is far less costly than training a classifier.

5 Conclusions

In this paper, we have proposed a methodology for deriving confidence regions around class prevalence estimates. This methodology relies on bootstrap, and is aimed at circumventing the computational complexity this strategy entails. To do so, we have focused on a specific type of quantification methods called "aggregative". These methods carry out the training and inference phases in two stages, the first of which is considerably slower than the other. We exploit this characteristic to apply the bootstrap resampling only in the second phase of the training or inference phases, thus speeding up the otherwise unaffordable cost.

We have empirically evaluated three strategies for applying bootstrap: the model-based approach, which applies resampling to the training set; the population-based approach, which applies resampling to the test set; and a combined approach that applies bootstrap both to the training and test sets. We have also investigated three ways for deriving confidence regions, including traditional confidence intervals, confidence ellipses in the simplex, and confidence ellipses in a transformed space. Our findings suggest that the combined approach paired with confidence intervals is the most effective method for deriving confidence regions. Practitioners should consider this approach when computational efficiency is a priority and when robustness under prior probability shift is required.

By construction, standard bootstrap methods rely on the i.i.d. assumption and therefore cannot provide valid coverage guarantees under prior probability shift. In such cases, bootstrap intervals merely quantify the internal variability of the quantifier on the given sample, while the additional uncertainty induced by distributional shift is not captured. In the near future, we plan to investigate other resampling schemes that explicitly incorporate this shift as a promising direction for achieving more reliable coverage properties.

Acknowledgments

This work has been funded by the QuaDaSh project "*Finanziato dall'Unione europea—Next Generation EU, Missione 4 Componente 2 CUP B53D23026250001*".

References

1. Beijbom, O., Hoffman, J., Yao, E., Darrell, T., Rodriguez-Ramirez, A., Gonzalez-Rivero, M., Hoegh-Guldberg, O.: Quantification in-the-wild: Data-sets and baselines (2015), coRR abs/1510.04811 (2015). Presented at the NIPS 2015 Workshop on Transfer and Multi-Task Learning, Montreal, CA
2. Bella, A., Ferri, C., Hernández-Orallo, J., Ramírez-Quintana, M.J.: Quantification via probability estimators. In: Proceedings of the 11th IEEE International Conference on Data Mining (ICDM 2010). pp. 737–742. Sydney, AU (2010). https://doi.org/10.1109/icdm.2010.75

3. Bunse, M.: Unification of algorithms for quantification and unfolding. INFORMATIK 2022 (2022). https://doi.org/10.18420/inf2022_37
4. Daughton, A.R., Paul, M.J.: Constructing accurate confidence intervals when aggregating social media data for public health monitoring. In: Proceedings of the 3rd AAAI International Workshop on Health Intelligence (W3PHIAI 2019). pp. 9–17. Phoenix, US (2019)
5. Denham, B., Lai, E.M., Sinha, R., Naeem, M.A.: Gain-some-lose-some: Reliable quantification under general dataset shift. In: 2021 IEEE International Conference on Data Mining (ICDM). pp. 1048–1053. IEEE (2021)
6. Efron, B., Tibshirani, R.J.: An introduction to the bootstrap (1993)
7. Esuli, A., Fabris, A., Moreo, A., Sebastiani, F.: Learning to quantify. Springer Nature, Cham, CH (2023). https://doi.org/10.1007/978-3-031-20467-8
8. Esuli, A., Moreo, A., Sebastiani, F., Sperduti, G.: A detailed overview of LeQua 2022: Learning to quantify. In: Working Notes of the 13th Conference and Labs of the Evaluation Forum (CLEF 2022). Bologna, IT (2022)
9. Fernandes Vaz, A., Izbicki, R., Bassi Stern, R.: Quantification under prior probability shift: The ratio estimator and its extensions. Journal of Machine Learning Research **20**, 79:1–79:33 (2019)
10. Firat, A.: Unified framework for quantification (2016)
11. Forman, G.: Counting positives accurately despite inaccurate classification. In: Proceedings of the 16th European Conference on Machine Learning (ECML 2005). pp. 564–575. Porto, PT (2005). https://doi.org/10.1007/11564096_55
12. Gao, W., Sebastiani, F.: From classification to quantification in tweet sentiment analysis. Social Network Analysis and Mining **6**(19), 1–22 (2016). https://doi.org/10.1007/s13278-016-0327-z
13. González, P., Castaño, A., Chawla, N.V., del Coz, J.J.: A review on quantification learning. ACM Computing Surveys **50**(5), 74:1–74:40 (2017). https://doi.org/10.1145/3117807
14. González, P., Moreo, A., Sebastiani, F.: Binary quantification and dataset shift: an experimental investigation. Data Mining and Knowledge Discovery pp. 1–43 (2024)
15. González-Castro, V., Alaiz-Rodríguez, R., Alegre, E.: Class distribution estimation based on the Hellinger distance. Information Sciences **218**, 146–164 (2013). https://doi.org/10.1016/j.ins.2012.05.028
16. Hopkins, D.J., King, G.: A method of automated nonparametric content analysis for social science. American Journal of Political Science **54**(1), 229–247 (2010). https://doi.org/10.1111/j.1540-5907.2009.00428.x
17. Keith, K.A., O'Connor, B.: Uncertainty-aware generative models for inferring document class prevalence. In: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (EMNLP 2018). Brussels, BE (2018)
18. King, G., Lu, Y., Shibuya, K.: Designing verbal autopsy studies. Population Health Metrics **19**(8) (2010). https://doi.org/10.1186/1478-7954-8-19
19. Meeker, W.Q., Hahn, G.J., Escobar, L.A.: Statistical intervals: a guide for practitioners and researchers, vol. 541. John Wiley & Sons (2017)
20. Moreo, A., Esuli, A., Sebastiani, F.: QuaPy: A Python-based framework for quantification. In: Proceedings of the 30th ACM International Conference on Knowledge Management (CIKM 2021). pp. 4534–4543. Gold Coast, AU (2021). https://doi.org/10.1145/3459637.3482015
21. Moreo, A., González, P., del Coz, J.J.: Kernel density estimation for multiclass quantification. Machine Learning **114**(4) (2025). https://doi.org/10.1007/s10994-024-06726-5

22. Moreo, A., Sebastiani, F.: Tweet sentiment quantification: An experimental re-evaluation. PLOS ONE **17**(9), 1–23 (September 2022). https://doi.org/10.1371/journal.pone.0263449
23. Schumacher, T., Strohmaier, M., Lemmerich, F.: A comparative evaluation of quantification methods (2023)
24. Tasche, D.: Confidence intervals for class prevalences under prior probability shift. Machine Learning and Knowledge Extraction **1**(3), 805–831 (2019). https://doi.org/10.3390/make1030047
25. Ziegler, A., Czyż, P.: Bayesian quantification with black-box estimators (2023), https://arxiv.org/abs/2302.09159

A Additional Material

In this section, we report, in disaggregated form, the results summarized in Tables 1 and 2 of the paper "An Efficient Method for Deriving Confidence Intervals in Aggregative Quantification".

The notational conventions we used are as follows. For each dataset and each metric, we highlight in **boldface** the best results. For the sake of a rapid interpretation of the results, we highlight, independently for each evaluation metric, the best result in intense green and the worst result in intense red; the rest of the results are interpolated between both extremes. All results are confronted against the best one by means of a Wilcoxon signed-rank test. We use superscripts † and ‡ to indicate the methods (if any) whose scores are *not* statistically significantly different from the best one at different confidence levels: symbol † indicates $0.001 < p < 0.01$, while symbol ‡ indicates $p \geq 0.01$. Each table is devoted to one quantification method, and displays the results of the original quantification method (i.e., one that does not provide confidence regions), only in terms of MAE, and all combinations of bootstrap methods (population-based, model-based, and combined) with the tree methods for generating confidence regions (CI, CE, and CT), in terms of MAE, $\mathcal{C}$, $\mathcal{A}$.

Tables 5a, 5b, and 5c report the results we have obtained for ACC, PACC, and DM, in the binary datasets, respectively; while Tables 6a, 6b, and 6c report the results we have obtained for ACC, PACC, and DM, in the multiclass datasets, respectively. The goodness measures for ACC, PACC, and DM are reported in Tables 7a, 7b, and 7c for the binary datasets, and in Tables 8a, 8b, and 8c for the multiclass case.

	ACC																							
	-	Population-based								Model-based								Combined					
	-	CI			CE			CT			CI			CE			CT			CI			CE			CT		
	MAE	MAE	$\bar{C}$	$\bar{A}$	MAE	$\bar{C}$	$\bar{A}$	MAE	$\bar{C}$	$\bar{A}$	MAE	$\bar{C}$	$\bar{A}$	MAE	$\bar{C}$	$\bar{A}$	MAE	$\bar{C}$	$\bar{A}$	MAE	$\bar{C}$	$\bar{A}$	MAE	$\bar{C}$	$\bar{A}$	MAE	$\bar{C}$	$\bar{A}$
balance.1	.028	**.023**	.82	**.08**	.024‡	.77	.12	**.023**	.81	.10	.029	.84	.09	.029	.76	.13	.029	.72	.12	.029	**.85**	.09	.029	.76	.13	.029	.66	.12
balance.3	.021	.014‡	.95	.08	**.014**	.91	.11	.014‡	.79	**.07**	.020	.97	.09	.020	.86	.13	.020	.76	.13	.020	**.98**	.10	.020	.82	.13	.020	.72	.10
breast-cancer	.020	.028	.70	**.07**	.028	.71	.12	.028	.71	.10	**.020**	.88	.09	.020‡	.77	.13	**.020**	.81	.17	.020‡	**.91**	.09	.020‡	.75	.12	.020‡	.78	.16
cmc.1	.110	.122	.53	**.24**	.122	.53	.34	.122	.57	.35	.106‡	**.80**	.30	**.106**	.65	.37	.106‡	.65	.47	.113	.76	.30	.113	.63	.37	.113	.57	.46
cmc.2	.171	.133‡	.84	**.40**	**.132**	.81	.46	.133‡	.63	.52	.158	**.91**	.54	.158	.72	.49	.158	.70	.67	.167	.88	.53	.167	.74	.52	.167	.58	.57
cmc.3	.187	.243	.37	**.36**	.243	.33	.43	.243	.45	.42	**.168**	.85	.54	.168‡	.65	.48	**.168**	.49	.49	.168‡	**.88**	.54	.169‡	.65	.49	.168‡	.56	.54
german	.080	.082	.66	**.20**	.083	.59	.29	.082	.58	.30	**.077**	.83	.28	.078‡	.74	.35	**.077**	.64	.38	.079	**.84**	.28	.079	.66	.35	.079	.65	.40
haberman	.149	.102‡	.85	**.42**	.102‡	.76	.47	.102‡	.56	.47	.096‡	**1.00**	**1.00**	**.095**	.72	.64	.096‡	.59	.59	.111	**1.00**	**1.00**	.111	.91	.78	.111	.53	.53
ionosphere	.083	.072‡	.16	**.10**	**.072**	.21	.14	.072‡	.20	.11	.084	.50	.17	.084	.46	.22	.084	.40	.23	.082	.45	.17	.082	**.53**	.23	.082	.43	.24
iris.1	**.000**	.001	**1.00**	**.07**	.001	.89	.10	.001	.88	.07	.001	**1.00**	**.07**	.001	.89	.10	.001	.88	.07	.000	**1.00**	.07	.000	.95	.10	.000	.92	.07
iris.2	.064	.049†	.95	**.24**	**.049**	.86	.32	.049†	.76	.33	.065	**1.00**	.57	.065	.87	.62	.065	.57	.56	.068	**1.00**	.54	.068	.96	.61	.068	.55	.53
iris.3	.085	.091	.00	**.08**	.091	.01	.20	.091	.03	.09	.085‡	.01	.14	**.084**	.21	.19	.085‡	**.34**	.22	.087	.05	.15	.087	.19	.20	.087	.29	.19
mammographic	.044	.049	.63	**.12**	.049	.60	.16	.049	.58	.12	.043	.90	.15	.043	.78	.21	.043	.67	.19	.041†	**.91**	.15	**.041**	.81	.21	.041†	.61	.19
pageblocks.5	.115	**.081**	.53	.15	.081‡	.51	.23	**.081**	.47	**.15**	.112	**.85**	.26	.112	.54	.37	.112	.58	.27	.114	.76	.24	.114	.54	.35	.114	.54	.25
sonar	.126	.104‡	.34	**.13**	**.103**	.33	.49	.104‡	.46	.23	.117‡	.53	.27	.117‡	.50	.41	.117‡	.48	.40	.119‡	.54	.28	.119‡	**.56**	.39	.119‡	.53	.42
spambase	**.015**	.016	**1.00**	**.09**	.016	.93	.12	.016	.90	.12	.016	**1.00**	.09	.016	.90	.13	.016	.78	.14	.016	**1.00**	.09	.016	.88	.13	.016	.73	.13
spectf	.109	.060†	.81	**.21**	**.060**	.66	.28	.060†	.62	.28	.095	**.96**	.43	.095	.74	.51	.095	.64	.57	.095	.93	.41	.095	.86	.51	.095	.54	.46
tictactoe	.008	**.006**	**1.00**	**.07**	.006†	.91	.10	**.006**	.87	.09	.007	**1.00**	.08	.008	.83	.11	.007	.78	.11	.008	**1.00**	.08	.008	.88	.11	.008	.77	.10
transfusion	.132	.120†	.86	**.48**	.120†	.74	.52	.120†	.59	.55	.117	**1.00**	.64	.117	.82	.60	.117	.61	.58	**.112**	**1.00**	.70	.112‡	.94	.69	**.112**	.68	.66
wdbc	.018	.017†	.84	.08	**.017**	.77	.11	.017†	.71	.08	.018‡	**.87**	.08	.018‡	.79	.11	.018‡	.72	.09	.018‡	.86	.08	.018‡	.68	.11	.018‡	.62	**.07**
wine.1	.006	.019	.80	.07	.019	.79	.13	.019	.69	**.06**	**.006**	**1.00**	.07	.006‡	.93	.10	**.006**	.83	.11	.007	**1.00**	.08	.007	.93	.11	.007	.88	.09
wine.2	.019	.011‡	.92	**.07**	**.011**	.89	.10	.011‡	.86	.07	.019	**1.00**	.09	.019	.88	.12	.019	.78	.12	.020	**1.00**	.09	.020	.88	.13	.020	.81	.14
wine.3	.023	.020†	.88	**.07**	.020†	.84	.11	.020†	.85	.08	.022	**1.00**	.10	.022	.87	.14	.022	.74	.14	**.020**	**1.00**	.10	.020†	.91	.14	**.020**	.77	.15
wine-q-red	.042	.032‡	**.97**	**.16**	**.032**	.92	.22	.032‡	.78	.26	.039	**.97**	.20	.039	.83	.25	.039	.70	.35	.040	**.97**	.20	.040	.87	.25	.040	.67	.30
wine-q-white	.040	.052	.86	**.20**	.053	.85	.27	.052	.71	.28	**.039**	**.98**	.21	.040†	.80	.28	**.039**	.75	.36	.040†	**.98**	.21	.040†	.88	.28	.040†	.73	.33
yeast	.158	.130†	.41	**.21**	**.130**	.35	.29	.130†	.47	.31	.156	.49	.29	.155	.43	.36	.156	**.56**	.45	.146	**.56**	.30	.146	.45	.38	.146	.49	.39
Mean	.071‡	.065‡	.72	**.17**	**.065**	.67	.24	.065‡	.64	.22	.066‡	**.85**	.26	.066‡	.73	.29	.066‡	.66	.31	.067‡	.85	.26	.067‡	.75	.30	.067‡	.64	.29

(a) Binary experiments for ACC.

	PACC																							
	-	Population-based								Model-based								Combined					
	-	CI			CE			CT			CI			CE			CT			CI			CE			CT		
	MAE	MAE	$\bar{C}$	$\bar{A}$	MAE	$\bar{C}$	$\bar{A}$	MAE	$\bar{C}$	$\bar{A}$	MAE	$\bar{C}$	$\bar{A}$	MAE	$\bar{C}$	$\bar{A}$	MAE	$\bar{C}$	$\bar{A}$	MAE	$\bar{C}$	$\bar{A}$	MAE	$\bar{C}$	$\bar{A}$	MAE	$\bar{C}$	$\bar{A}$
balance.1	**.024**	.033	.67	**.08**	.033	.57	.12	.033	.58	.10	.025	**.94**	.09	.025	.81	.13	.025	.73	.13	.025	.91	.09	.025	.78	.13	.025	.70	.10
balance.3	.026	.023‡	.90	**.08**	**.022**	.76	.11	.023‡	.74	.08	.026	**1.00**	.09	.025	.84	.13	.026	.74	.11	.028	**1.00**	.09	.028	.76	.13	.028	.75	.13
breast-cancer	.016	**.015**	.84	**.07**	.015‡	.74	.11	**.015**	.77	.11	.016‡	**.92**	.09	.016‡	.83	.12	.016‡	.73	.14	.016‡	.91	.09	.016‡	.74	.12	.016‡	.71	.11
cmc.1	.075	.079	.70	**.20**	.079	.57	.29	.079	.61	.27	.074‡	**.86**	.26	**.074**	.84	.34	.074‡	.56	.30	.080	.85	.26	.080	.70	.32	.080	.66	.39
cmc.2	.047	.054	.88	**.21**	.054	.81	.28	.054	.59	.25	**.045**	**.98**	.28	.045‡	.81	.36	**.045**	.67	.39	.045‡	**.98**	.29	.045‡	.79	.36	.045‡	.65	.40
cmc.3	.125	.104‡	.67	**.23**	**.104**	.53	.31	.104‡	.52	.31	.121	.68	.30	.121	.54	.38	.121	.50	.37	.120	**.70**	.30	.120	.60	.36	.120	.51	.39
german	.037	.037‡	.85	**.15**	.036‡	.71	.21	.037‡	.65	.23	**.034**	.97	.22	.034‡	.88	.30	**.034**	.67	.35	.035‡	**.98**	.22	.034‡	.86	.28	.035‡	.51	.26
haberman	.117	.113	.51	**.23**	.113	.39	.41	.113	.47	.33	.098	.94	.47	.098	.73	.41	.098	.49	.46	**.094**	**.97**	.50	.094‡	.84	.44	**.094**	.44	.44
ionosphere	.097	.086‡	.01	**.09**	**.086**	.04	.16	.086‡	.10	.09	.097	.18	.16	.097	.25	.21	.097	**.29**	.19	.098	.13	.16	.098	.28	.21	.098	.26	.19
iris.1	.005	.013	.90	**.07**	.013	.85	.11	.013	.77	.08	.005‡	**1.00**	.07	**.005**	.85	.10	.005‡	.78	.11	.005	**1.00**	.07	.005	.79	.10	.005	.80	.09
iris.2	.062	.047†	.74	**.15**	**.047**	.63	.22	.047‡	.58	.21	.054	**1.00**	.38	.054	.85	.42	.054	.65	.56	.058	**1.00**	.37	.058	.88	.44	.058	.55	.47
iris.3	.011	.008‡	.98	**.08**	**.008**	.88	.11	.008‡	.80	.10	.011†	**1.00**	.13	.011†	.84	.17	.011‡	.73	.16	.011†	**1.00**	.13	.011†	.87	.17	.011†	.66	.14
mammographic	.037	.044	.62	**.11**	.045	.61	.15	.044	.52	.12	.036	**.93**	.14	.037	.82	.20	.036	.73	.20	.035‡	.92	.14	**.035**	.79	.19	.035‡	.66	.17
pageblocks.5	.029	**.021**	.96	**.12**	.022‡	.83	.16	**.021**	.74	.15	.031	**1.00**	.18	.031	.84	.25	.031	.78	.22	.033	**1.00**	.18	.033	.85	.24	.033	.66	.19
sonar	.164	.095‡	.39	**.12**	**.095**	.36	.36	.095‡	.42	.18	.157	.36	.24	.157	.30	.33	.157	.45	.35	.157	.35	.24	.157	.33	.36	.157	**.46**	.36
spambase	**.012**	.013	.99	**.08**	.014	.86	.12	.013	.70	.11	.012	**1.00**	.09	.012	.90	.12	.012	.76	.15	.012	**1.00**	.09	.012	.91	.12	.012	.73	.12
spectf	.085	**.070**	.54	**.15**	.070‡	.51	.21	**.070**	.54	.25	.074	**.96**	.33	.074	.73	.40	.074	.70	.54	.073	.94	.32	.073	.74	.39	.073	.56	.43
tictactoe	.014	.013‡	.94	**.08**	**.013**	.79	.12	.013‡	.78	.10	.014‡	.94	.08	.014‡	.76	.12	.014‡	.73	.10	.014†	**.96**	.08	.014†	.78	.13	.014†	.67	.10
transfusion	.061	.065	.71	**.18**	.065	.62	.27	.065	.53	.26	**.058**	**.99**	.28	.058†	.74	**.37**	**.058**	.62	.36	.059	**.99**	.28	.059	.86	.36	.059	.61	.30
wdbc	.017	.013†	**.94**	.07	**.013**	.85	.11	.013‡	.71	**.07**	.017	.87	.08	.017	.75	.11	.017	.68	.11	.017	.86	.08	.017	.72	.11	.017	.62	.09
wine.1	.005	.015	.81	**.07**	.015	.73	.13	.015	.75	.08	.004‡	**1.00**	.09	.004‡	.89	.12	.004‡	.78	.13	.004‡	**1.00**	.09	**.004**	.86	.13	.004‡	.77	.12
wine.2	.017	**.009**	.95	**.07**	.009‡	.80	.11	**.009**	.77	.08	.017	**1.00**	.09	.017	.83	.14	.017	.82	.15	.016	**1.00**	.10	.016	.91	.13	.016	.67	.13
wine.3	.026	.026‡	.81	**.07**	.026‡	.67	.12	.026‡	.72	.08	**.026**	**.99**	.09	.026‡	.84	.13	**.026**	.71	.12	.026‡	**.99**	.09	.026‡	.75	.13	.026‡	.73	.13
wine-q-red	.030	.032‡	.90	**.14**	.032‡	.80	.19	.032‡	.68	.22	.029‡	**.98**	.17	**.029**	.87	.23	.029‡	.72	.30	.029‡	**.98**	.18	.029‡	.81	.23	.029‡	.68	.31
wine-q-white	.037	.062	.70	**.15**	.063	.63	.21	.062	.58	.20	**.036**	**.93**	.17	.037‡	.84	**.23**	**.036**	.61	.21	.037‡	**.93**	.17	.036‡	.76	**.24**	.037‡	.62	.21
yeast	.128	.131	.07	**.16**	.131	.06	.25	.131	.19	.19	.128	.33	.21	.128	.31	.29	.128	.35	.26	.122‡	.37	.21	**.122**	**.39**	.29	.122‡	.37	.22
Mean	.050‡	**.047**	.73	**.12**	.047‡	.64	.19	**.047**	.61	.16	.048‡	**.88**	.18	.048‡	.75	.23	.048‡	.65	.25	.048‡	.87	.19	.048‡	.74	.24	.048‡	.62	.23

(b) Binary experiments for PACC.

	DM																							
	-	Population-based								Model-based								Combined					
	-	CI			CE			CT			CI			CE			CT			CI			CE			CT		
	MAE	MAE	$\bar{C}$	$\bar{A}$	MAE	$\bar{C}$	$\bar{A}$	MAE	$\bar{C}$	$\bar{A}$	MAE	$\bar{C}$	$\bar{A}$	MAE	$\bar{C}$	$\bar{A}$	MAE	$\bar{C}$	$\bar{A}$	MAE	$\bar{C}$	$\bar{A}$	MAE	$\bar{C}$	$\bar{A}$	MAE	$\bar{C}$	$\bar{A}$
balance.1	.012	.029	.72	**.08**	.029	.67	.13	.029	.66	.11	.011‡	**.95**	.09	**.011**	.83	.13	.011‡	.75	.11	.011	**.95**	.09	.011	.77	.12	.011	.74	.12
balance.3	.024	.020†	.94	.08	**.019**	.82	.12	.020†	.69	.08	.024	**.95**	.08	.024	.73	.13	.024	.74	.08	.025	.93	.08	.025	.76	.12	.025	.69	**.07**
breast-cancer	.018	**.015**	**.96**	**.07**	.015‡	.84	.10	**.015**	.73	.08	.019	.93	.08	.019	.83	.12	.019	.82	.10	.019	.93	.08	.019	.79	.11	.019	.71	.10
cmc.1	.069	.079	.72	**.21**	.079	.62	.27	.079	.51	.27	.062‡	**.93**	.26	**.061**	.77	.33	.062‡	.65	.39	.067	.92	.25	.067	.77	.33	.067	.62	.31
cmc.2	.053	.053†	.87	**.20**	**.053**	.81	.28	.053†	.68	.31	.054†	.96	.27	.054†	.84	.35	.054†	.58	.38	.055†	**.97**	.28	.055†	.82	.35	.055†	.65	.37
cmc.3	.105	**.082**	.76	**.23**	.082†	.65	.30	**.082**	.58	.34	.099	.78	.30	.098	.59	.36	.099	.51	.38	.098	**.79**	.31	.098	.63	.37	.098	.46	.36
german	.063	.063	.62	**.14**	.063	.53	.27	.063	.54	.22	.053†	.88	.23	**.052**	.66	.30	.053†	.61	.33	.053†	**.90**	.23	.053†	.65	.29	.053†	.54	.31
haberman	.072	**.049**	.90	**.22**	.049†	.75	.30	**.049**	.61	.23	.074	**1.00**	.54	.074	.80	.60	.074	.69	.64	.066	**1.00**	.51	.066	.88	.62	.066	.47	.42
ionosphere	.084	.065†	.09	**.09**	**.065**	.21	.15	.065†	.22	.10	.075	.53	.16	.075	**.61**	.22	.075	.49	.22	.074	.55	.16	.074	.54	.21	.074	.45	.20
iris.1	.002	.016	**1.00**	.07	.016	.91	.10	.016	.85	**.07**	**.002**	**1.00**	.07	.002†	.93	.10	**.002**	.84	.08	.003	**1.00**	.07	.003	.87	.10	.003	.77	.07
iris.2	**.034**	.051	.62	**.12**	.051	.57	.21	.051	.60	.18	.038	**1.00**	.34	.038	.80	.43	.038	.70	.40	.040	**1.00**	.34	.040	.80	.44	.040	.55	.33
iris.3	.051	**.031**	.65	**.08**	.031‡	.64	.13	**.031**	.59	.08	.042	**.99**	.13	.042	.81	.18	.042	.74	.14	.042	.98	.12	.042	.79	.17	.042	.64	.13
mammographic	**.037**	.043	.63	**.11**	.043	.62	.15	.043	.48	.11	.039	.91	.14	.039	.74	.19	.039	.69	.16	.038	**.92**	.14	.038	.72	.19	.038	.61	.13
pageblocks.5	.048	**.042**	.71	**.10**	.042†	.62	.14	**.042**	.62	.11	.060	.83	.14	.060	.65	.20	.060	.60	.17	.061	**.89**	.15	.061	.62	.21	.061	.67	.20
sonar	.220	**.135**	.09	**.11**	.135‡	.07	.16	**.135**	.10	.11	.205	.01	.25	.205	.08	.33	.205	.30	.30	.208	.01	.26	.208	.05	.35	.208	**.34**	.34
spambase	.011	.012‡	**.99**	**.08**	.012‡	.86	.11	.012‡	.81	.10	**.011**	**.99**	.08	.011‡	.83	.12	**.011**	.78	.10	.011‡	**.99**	.08	.011‡	.91	.12	.011‡	.77	.10
spectf	**.090**	.208	.01	**.13**	.208	.04	.28	.208	.09	.15	.121	**.97**	.37	.121	.77	.47	.121	.65	.49	.126	.93	.36	.126	.74	.44	.126	.69	.49
tictactoe	.014	**.010**	**.99**	**.07**	.010†	.92	.11	**.010**	.87	.09	.013	**.99**	.08	.013	.92	.12	.013	.80	.11	.014	**.99**	.08	.014	.89	.13	.014	.73	.14
transfusion	**.074**	.107	.33	**.17**	.107	.37	.24	.107	.38	.25	.090	**.93**	.28	.091	.76	.37	.090	.65	.41	.093	.86	.27	.093	.70	.35	.093	.55	.37
wdbc	**.012**	.017	.89	.07	.017	.78	.11	.017	.67	**.07**	.012	**.92**	.08	.012	.81	.12	.012	.72	.09	.012	**.92**	.08	.012	.81	.11	.012	.72	.07
wine.1	**.027**	.065	.10	.07	.065	.11	.11	.065	.10	**.06**	.031	**.99**	.09	.031	.82	.13	.031	.76	.08	.031	**.99**	.09	.031	.83	.13	.031	.77	.09
wine.2	**.009**	.012	.98	.07	.012	.84	.12	.012	.68	**.07**	.012	**.99**	.09	.012	.83	.13	.012	.71	.10	.011	**.99**	.09	.011	.82	.13	.011	.68	.08
wine.3	.006	.006	**.99**	**.07**	.006	.87	.11	.006	.83	.09	.005	**.99**	.08	.005	.90	.11	.005	.77	.11	.004†	**.99**	.08	**.004**	.86	.11	.004†	.67	.07
wine-q-red	.028	.030†	.92	**.14**	.030†	.83	.19	.030†	.67	.19	.026†	**1.00**	.17	**.026**	.83	.23	.026†	.74	.23	.026†	**1.00**	.17	.026†	.83	.24	.026†	.70	.25
wine-q-white	**.037**	.056	.74	**.16**	.057	.62	.22	.056	.59	.22	.038	**.94**	.17	.039	.83	.23	.038	.64	.22	.038	**.94**	.17	.038	.80	.23	.038	.60	.19
yeast	.075	.070†	.52	**.14**	.070†	.49	.20	.070†	.42	.15	.069	.72	.18	.069	.66	.25	.069	.57	.24	**.066**	**.77**	.18	.066†	.65	.25	**.066**	.49	.21
Mean	**.049**	.052‡	.68	**.12**	.053‡	.62	.18	.052‡	.56	.15	.049‡	.89	.18	.049‡	.76	.24	.049‡	.67	.23	.050‡	**.89**	.18	.050‡	.74	.24	.050‡	.63	.21

(c) Binary experiments for DM.

Table 5: Binary experiments for ACC, PACC, and DM.

	ACC																							
	-	Population-based								Model-based								Combined							
	-	CI			CE			CT		CI			CE			CT		CI			CE			CT		
	MAE	MAE	$\mathcal{C}$	$\mathcal{A}$	MAE	$\mathcal{C}$	$\mathcal{A}$	MAE	$\mathcal{C}$	$\mathcal{A}$	MAE	$\mathcal{C}$	$\mathcal{A}$	MAE	$\mathcal{C}$	$\mathcal{A}$	MAE	$\mathcal{C}$	$\mathcal{A}$	MAE	$\mathcal{C}$	$\mathcal{A}$	MAE	$\mathcal{C}$	$\mathcal{A}$
hcv	.297	.227‡	.34	.45	.227‡	.35	.48	.227‡	.36	**.36**	.219‡	.69	.83	**.218**	.70	.85	.219‡	**.80**	.85	.219‡	.68	.81	.220‡	.66	.78	.219‡	.79	.85
image_seg	.006	.008	.67	**.00**	.008	.82	.00	.008	.85	.01	.006	.92	.00	.006	.92	.02	.006	**.98**	.01	**.006**	.90	.00	.006‡	.93	.01	**.006**	.94	.02
phishing	.073	.077	.51	**.05**	.077	.46	.12	.077	.61	.16	**.066**	**.84**	.13	.066‡	.82	.15	**.066**	.82	.52	.070	.83	.14	.070	.80	.13	.070	.79	.50
page_block	.054	.068	.19	**.01**	.068	.31	.03	.068	.28	.03	.049‡	.77	.03	**.049**	**.85**	.06	.049‡	.84	.07	.051	.73	.03	.051	.77	.06	.051	.75	.07
chess	.045	.037†	.09	**.00**	.037†	.26	.00	.037†	.13	.05	.036‡	.19	.00	.036‡	.42	.00	.036‡	.16	.07	**.036**	.22	.00	**.036**	**.47**	.01	**.036**	.21	.08
mhr	.126	.124	.31	**.05**	.124	.34	.09	.124	.56	.18	.111‡	.60	.11	.111‡	.62	.11	.111‡	.72	.39	**.111**	.59	.11	.111‡	.59	.10	**.111**	**.76**	.41
connect-4	.134	.111‡	.82	.21	**.111**	.80	.20	.111‡	**.83**	.71	.116‡	.81	.23	.116‡	.78	.20	.116‡	.80	.67	.117‡	.78	.23	.117‡	.79	**.18**	.117‡	.79	.66
poker_hand	.139	.139	.00	**.00**	.139	.00	**.00**	.139	.00	**.00**	.121‡	.00	**.00**	.121‡	.41	.43	.121‡	**.65**	.65	**.121**	.00	**.00**	**.121**	.10	.07	**.121**	.26	.33
molecular	.019	.018‡	.77	**.01**	**.018**	.79	.03	.018‡	.77	.04	.019‡	.78	.01	.019‡	.79	.03	.019‡	**.82**	.03	.019‡	.76	.01	.019†	.77	.02	.019†	.80	.04
waveform-v1	.012	.013‡	.97	**.01**	.012†	.94	.03	.013‡	.94	.05	.012‡	.99	.01	**.012**	.94	.03	.012‡	.91	.05	.012‡	**1.00**	.01	.012†	.93	.04	.012‡	.97	.06
isolet	.002	.002‡	.66	**.00**	.002†	**.97**	.00	.002†	.92	.04	.002	.69	**.00**	.002†	.96	.02	.002	**.97**	.04	**.002**	.73	**.00**	.002†	**.97**	.02	**.002**	.95	.07
cmc	.110	.121†	.62	**.16**	.121†	.61	.16	.121†	.65	.43	**.103**	**.89**	.24	.103†	.82	.25	**.103**	.83	.66	.103†	.88	.24	.103†	.71	.24	.103†	.81	.62
shuttle	**.066**	.068	.12	**.00**	.068	.11	.00	.068	.12	.00	.067	.19	.00	.067	.22	.01	.067	.23	.02	.067	.20	.00	.067	.21	.02	.067	**.23**	.04
satellite	.015	.015‡	.80	**.00**	.015‡	.87	.01	.015‡	.97	.01	.014‡	.88	.00	.014‡	.94	.01	.014‡	**.99**	.02	**.014**	.90	.00	.014‡	.92	.01	**.014**	.95	.02
hand_digits	.004	.004	.96	**.00**	.004	.98	.01	.004	**.99**	.00	.004†	.97	**.00**	.004‡	**.99**	.01	.004†	.96	.01	.004‡	.98	**.00**	**.004**	.95	.01	.004†	.98	.02
yeast	.158	.130†	.41	**.21**	**.130**	.35	.29	.130†	.47	.31	.156	.49	.29	.155	.43	.36	.156	**.56**	.45	.146	**.56**	.30	.146	.45	.38	.146	.49	.39
nursery	.011	.011‡	.95	**.00**	**.010**	.92	.01	.011‡	.96	.02	.011‡	**.98**	.00	.011‡	.97	.01	.011‡	.97	.02	.010‡	**.98**	.00	.010‡	.95	.01	.010‡	**.98**	.02
obesity	.012	.010‡	.86	**.00**	**.010**	.87	.02	.010‡	.91	.02	.011	.94	.00	.011	.91	.01	.011	**.97**	.02	.011	.95	.00	.011	.91	.02	.011	**.97**	.01
abalone	.097	.081	.05	**.01**	.081	.05	.04	.081	.07	.08	.068‡	.34	.10	.068‡	.56	.07	.068‡	.24	.28	.068‡	.39	.11	**.068**	**.60**	.07	.068‡	.23	.24
letter	.006	.007	.49	**.00**	.007	.76	.01	.007	.72	.08	**.006**	.53	**.00**	.006‡	.79	.02	**.006**	.75	.08	.006‡	.58	**.00**	.006‡	**.83**	.03	.006‡	.80	.11
digits	.003	.003‡	.93	**.00**	.004‡	**.98**	.01	.003‡	.96	.02	.003†	.95	**.00**	.003‡	.96	.01	.003†	**.98**	.00	**.003**	.95	**.00**	.003‡	.95	.01	**.003**	**.98**	.01
academic-success	.037	.037‡	.89	**.04**	.037‡	.90	.08	.037‡	.87	.15	**.036**	.93	.05	.036‡	**.94**	.10	**.036**	.92	.18	.036‡	**.94**	.05	.036‡	.92	.11	.036‡	.88	.18
wine-quality	.171	.166	.01	.00	.166	.01	.00	.166	.00	**.00**	.147‡	.28	.15	.147‡	.34	.12	.147‡	.70	.56	.146‡	.39	.17	**.146**	.36	.13	.146‡	**.77**	.60
dry-bean	.005	.005	.97	**.00**	.005	.92	.00	.005	**.98**	.01	.005	.97	.00	.005†	.97	.01	.005	.96	.00	**.005**	.97	.00	.005†	.97	.01	**.005**	.94	.02
Mean	.067	.062†	.56	**.05**	.062†	.60	.07	.062†	.62	.12	.058‡	.69	.09	.058‡	.75	.12	.058‡	**.77**	.23	**.058**	.70	.09	.058‡	.73	.10	**.058**	.75	.22

(a) Multiclass experiments for ACC.

	PACC																							
	-	Population-based								Model-based								Combined							
	-	CI			CE			CT		CI			CE			CT		CI			CE			CT		
	MAE	MAE	$\mathcal{C}$	$\mathcal{A}$	MAE	$\mathcal{C}$	$\mathcal{A}$	MAE	$\mathcal{C}$	$\mathcal{A}$	MAE	$\mathcal{C}$	$\mathcal{A}$	MAE	$\mathcal{C}$	$\mathcal{A}$	MAE	$\mathcal{C}$	$\mathcal{A}$	MAE	$\mathcal{C}$	$\mathcal{A}$	MAE	$\mathcal{C}$	$\mathcal{A}$
hcv	.305	.274	.14	**.17**	.273	.14	.19	.274	.23	.30	**.235**	.52	.66	.236‡	.49	.67	**.235**	.72	.78	.237‡	.51	.65	.237‡	.50	.66	.237‡	**.73**	.77
image_seg	.006	.007	.83	**.00**	.007	.88	.01	.007	.90	.00	.005‡	.95	.00	.006	**.96**	.01	.005‡	.95	.01	.005‡	**.96**	.00	**.005**	.93	.01	.005‡	.94	.01
phishing	.035	.037	.71	**.02**	.037	.68	.06	.037	.82	.09	.033	**.92**	.05	.033†	.89	.09	.033	**.92**	.24	**.032**	.91	.05	**.032**	.87	.10	**.032**	.84	.23
page_block	.042	.042	.40	**.01**	.042	.56	.03	.042	.46	.04	.039‡	**.82**	.02	**.039**	.80	.04	.039‡	.81	.06	.042	.69	.02	.042	.69	.04	.042	.80	.04
chess	.041	.034‡	.18	**.00**	**.034**	**.38**	.01	.034‡	.21	.07	.034‡	.22	.00	.034‡	**.38**	.01	.034‡	.23	.08	.034‡	.22	.00	.034‡	**.38**	.01	.034‡	.25	.07
mhr	.056	**.043**	.79	**.05**	.043‡	.76	.07	**.043**	.85	.17	.051	.86	.08	.052	.86	.10	.051	.85	.32	.049	**.90**	.08	.049	.77	.09	.049	.86	.33
connect-4	.033	.032‡	.88	**.04**	**.032**	.89	.08	.032‡	.94	.13	.032‡	.88	.04	.032‡	.88	.08	.032‡	**.98**	.15	.032‡	.90	.04	.032‡	.86	.09	.032‡	.88	.14
poker_hand	.098	.099	.00	.00	.099	.00	**.00**	.099	.14	.14	.091‡	.12	.10	**.091**	.09	.09	.091‡	**.85**	.86	.091‡	.10	.09	.091‡	.08	.08	.091‡	.83	.86
molecular	.017	.017‡	.77	**.01**	**.017**	.78	.02	.017‡	.79	.03	.017‡	.80	.01	.017‡	**.85**	.02	.017‡	.81	.03	.018‡	.80	.01	.018‡	.74	.02	.018‡	.77	.04
waveform-v1	.010	.010‡	**.99**	**.01**	.010‡	.95	.02	.010‡	.91	.04	.010‡	.98	.01	**.010**	.93	.03	.010‡	.94	.04	.010‡	.98	.01	.010‡	.93	.03	.010‡	.94	.05
isolet	.002	.002‡	.66	**.00**	.002‡	.97	.01	.002‡	.96	.04	.002	.68	**.00**	.002†	.97	.01	.002	**.98**	.06	**.002**	.71	**.00**	**.002**	**.98**	.03	**.002**	.97	.03
cmc	.103	.146	.06	**.05**	.146	.13	.13	.146	.27	.19	.099	.53	.11	.099	.59	.15	.099	**.73**	.41	.097‡	.56	.11	**.097**	.64	.16	.097‡	.72	.41
shuttle	.049	.061	.33	**.00**	.061	.43	.01	.061	.57	.02	**.043**	**.97**	.02	.043‡	.94	.04	**.043**	**.97**	.18	.043	**.97**	.02	.043	.93	.03	.043	.94	.20
satellite	.010	.011	.92	**.00**	.011†	.92	.01	.011	.94	.01	.010‡	.94	.00	.010‡	.93	.01	.010‡	**.98**	.01	**.010**	.95	.00	.010‡	.92	.01	**.010**	.97	.02
hand_digits	.003	.004	.95	**.00**	.004	.96	.00	.004	.97	.00	.003‡	.97	**.00**	.003†	.96	.00	.003‡	**.99**	.02	.003‡	.97	**.00**	**.003**	.93	.01	.003‡	.97	.02
yeast	.128	.131	.07	**.16**	.131	.06	.25	.131	.19	.19	.128	.33	.21	.128	.31	**.29**	.128	.35	.26	.122‡	.37	.21	**.122**	**.39**	**.29**	.122‡	.37	.22
nursery	.010	.008‡	**.98**	**.00**	**.008**	.97	.01	.008†	.94	.02	.010	.96	.00	.010	.97	.01	.010	.94	.02	.010	.97	.00	.010	.91	.02	.010	**.96**	.02
obesity	.015	.013‡	.60	**.00**	**.013**	.73	.03	.013‡	.81	.02	.015†	.66	.00	.015	.78	.02	.015†	**.86**	.00	.015	.66	.00	.015	.81	.03	.015	.82	.02
abalone	.079	.072	.14	**.01**	.072	.16	.01	.072	.18	.14	**.057**	.55	.07	.057‡	**.59**	.05	**.057**	.52	.40	.057‡	.51	.07	.057‡	.58	.05	.057‡	.49	.37
letter	.005	.006	.49	**.00**	.006	.77	.02	.006	.72	.07	.005‡	.55	**.00**	**.005**	.81	.01	.005‡	.86	.07	.005‡	.56	**.00**	.005‡	**.88**	.03	.005‡	.86	.06
digits	**.003**	.003	.91	**.00**	.003	.94	.01	.003	.96	.01	.003	.93	**.00**	**.003**	.94	.02	.003	**.97**	.02	.003	.93	**.00**	.003	.95	.01	.003	.93	.02
academic-success	.035	.038	.69	**.02**	.038	.73	.06	.038	.80	.10	.034	.88	.03	.034	.84	.06	.034	.89	.11	**.032**	.89	.03	.032‡	.83	.07	**.032**	**.90**	.14
wine-quality	.123	.106‡	.27	**.02**	.106‡	.35	.03	.106‡	.34	.16	.106	.43	.06	.106	.51	.06	.106	**.59**	.43	.104‡	.46	.07	**.104**	.51	.07	.104‡	.58	.42
dry-bean	.004	.005†	**.98**	**.00**	.005†	.96	.00	.005†	.95	.02	.005‡	**.98**	**.00**	.005‡	.94	.02	.005‡	.95	.01	**.004**	**.98**	.00	.004‡	.89	.01	**.004**	.93	.02
Mean	.051	.050†	.57	**.02**	.050‡	.63	.04	.050†	.66	.08	.045‡	.73	.06	.045‡	.76	.08	.045‡	**.82**	.19	.044‡	.73	.06	**.044**	.75	.08	.044‡	.80	.19

(b) Multiclass experiments for PACC.

	DM																							
	-	Population-based								Model-based								Combined							
	-	CI			CE			CT		CI			CE			CT		CI			CE			CT		
	MAE	MAE	$\mathcal{C}$	$\mathcal{A}$	MAE	$\mathcal{C}$	$\mathcal{A}$	MAE	$\mathcal{C}$	$\mathcal{A}$	MAE	$\mathcal{C}$	$\mathcal{A}$	MAE	$\mathcal{C}$	$\mathcal{A}$	MAE	$\mathcal{C}$	$\mathcal{A}$	MAE	$\mathcal{C}$	$\mathcal{A}$	MAE	$\mathcal{C}$	$\mathcal{A}$
hcv	.223	.202	.17	**.17**	.202	.20	.23	.202	.48	.51	.183‡	.68	.75	.183‡	.75	.79	.183‡	**.89**	.91	**.182**	.62	.72	.182‡	.72	.76	**.182**	.85	.86
image_seg	.011	.013	.39	**.00**	.013	.77	.00	.013	.82	.02	.011	.84	.00	.011	.96	.00	.011	.95	.02	.011‡	.89	.00	**.011**	**.97**	.00	.011‡	**.97**	.03
phishing	.050	.075	.29	**.02**	.075	.35	.05	.075	.42	.04	.064	.56	.04	.064	.64	.09	.064	.66	.11	.063	.58	.04	.063	.62	.09	.063	**.68**	.11
page_block	**.038**	.057	.16	**.01**	.057	.22	.02	.057	.31	.02	.043	.56	.02	.043	**.69**	.04	.043	**.69**	.05	.045	.46	.02	.045	.54	.04	.045	.62	.06
chess	.038	.033	.19	**.00**	.033	.34	.00	.033	.24	.08	**.031**	.40	.00	.031‡	.58	.00	**.031**	.41	.14	.031‡	.43	.00	.031‡	**.64**	.00	.031‡	.46	.19
mhr	.060	.041‡	.85	**.04**	**.041**	.81	.08	.041‡	.87	.14	.056	.84	.07	.056	.90	.11	.056	.90	.24	.055	.83	.06	.055	.81	.11	.055	**.91**	.24
connect-4	.033	.032‡	.95	**.04**	**.031**	.93	.07	.032‡	**.96**	.13	.032‡	.94	.04	.032‡	.94	.08	.032‡	.93	.14	.032‡	.94	.04	.032‡	.89	.08	.032‡	.93	.14
poker_hand	.118	.118	.05	.01	.118	.00	**.00**	.118	.03	.02	.099‡	.35	.32	.099‡	.49	.52	.099‡	**.89**	.92	.099‡	.34	.33	**.099**	.46	.47	.099‡	.87	.94
molecular	.018	.016‡	.83	**.01**	**.016**	.82	.02	.016‡	.81	.02	.019	.81	.01	.019	**.85**	.02	.019	.83	.03	.019	.82	.01	.019	.84	.02	.019	.81	.03
waveform-v1	.009	.010‡	**.98**	**.01**	.010‡	.93	.02	.010‡	.94	.03	.009‡	**.98**	.01	.009‡	.94	.02	.009‡	.95	.04	.009‡	**.98**	.01	**.009**	.91	.03	.009‡	.97	.04
isolet	.004	.004‡	.78	**.00**	.004	.98	.01	.004†	.93	.03	.004‡	.82	**.00**	.004‡	**1.00**	.02	.004‡	.98	.07	**.004**	.83	**.00**	.004‡	**1.00**	.00	**.004**	.98	.07
cmc	.080	.121	.12	**.06**	.120	.22	.12	.121	.30	.20	.068‡	.87	.12	**.068**	.86	.17	.068‡	**.90**	.41	.068‡	.89	.12	.068‡	**.90**	.18	.068‡	.86	.43
shuttle	**.040**	.050	.52	**.01**	.050	.69	.01	.050	.66	.04	.050	.83	.01	.050	**.93**	.02	.050	.80	.12	.050	.81	.01	.050	.89	.02	.050	.79	.12
satellite	.011	.011	.87	**.00**	.011	.90	.00	.011	.90	.02	.011†	.91	.00	.011‡	.93	.02	.011†	**.95**	.01	**.011**	.90	.00	.011‡	.91	.01	**.011**	**.95**	.01
hand_digits	**.005**	.005	.93	**.00**	.005	.95	.01	.005	.95	.01	.005	.93	**.00**	.005	**.96**	.02	.005	**.96**	.01	.005	.95	**.00**	.005	.93	.01	.005	.94	.03
yeast	.075	.070†	.52	**.14**	.070†	.49	.20	.070‡	.42	.15	.069	.72	.18	.069	.66	.25	.069	.57	.24	**.066**	**.77**	.18	.066‡	.65	.25	**.066**	.49	.21
nursery	**.007**	.009	.96	**.00**	.009	.94	.01	.009	.95	.02	.009	**.99**	.00	.009	.96	.01	.009	.96	.02	.009	**.99**	.00	.009	.95	.01	.009	.97	.02
obesity	.015	.015†	.51	**.00**	.015‡	.72	.01	.015†	.83	.02	.014‡	.78	.00	**.014**	.86	.01	.014‡	**.96**	.01	.014‡	.77	.00	.014‡	.79	.01	.014‡	.95	.04
abalone	.079	.067	.08	**.01**	.067	.12	.02	.067	.14	.09	.055‡	.75	.13	**.054**	**.82**	.07	.055‡	.78	.60	.055‡	.71	.13	.055‡	**.82**	.07	.055‡	.73	.56
letter	.007	.007	.17	**.00**	.007	.32	.00	.007	.36	.06	**.007**	.23	**.00**	.007‡	.37	.03	**.007**	.39	.04	.007‡	.23	**.00**	.007‡	**.45**	.03	.007‡	.42	.08
digits	.005	.005‡	.93	**.00**	.005‡	.87	.02	.005‡	.95	.01	.004‡	.93	**.00**	.005	**.97**	.01	.004‡	.95	.02	**.004**	.93	**.00**	.004‡	.95	.02	**.004**	.91	.03
academic-success	.026	.025†	.87	**.02**	.025†	.87	.08	.025†	.84	.08	.024	.92	.03	.024	.92	.06	.024	.92	.10	.023‡	**.95**	.03	**.023**	.89	.07	.023‡	.89	.09
wine-quality	.141	.134	.11	**.02**	.134	.11	.03	.134	.33	.21	**.124**	.40	.09	.124‡	.39	.06	**.124**	**.72**	.57	.125‡	.36	.09	.125‡	.36	.05	.125‡	.69	.53
dry-bean	**.005**	.005	.95	**.00**	.005	.95	.01	.005	.93	.01	.005	**.97**	.00	.005	.93	.01	.005	**.97**	.01	.005	**.97**	.00	.005	.94	.01	.005	.93	.01
Mean	.046‡	.047†	.55	**.02**	.047†	.60	.04	.047†	.64	.08	.042‡	.75	.08	.042‡	.80	.10	.042‡	**.83**	.20	.041‡	.75	.08	**.041**	.78	.10	.041‡	.82	.20

(c) Multiclass experiments for DM.

Table 6: Multiclass experiments for ACC, PACC, and DM.

	Population-based			Model-based			Combined		
	CI	CE	CT	CI	CE	CT	CI	CE	CT
balance.1	**2.127**	1.763	1.979†	2.050	1.643	1.586	2.097	1.647	1.477
balance.3	**2.516**	2.140	2.013	2.432	1.900	1.695	2.434	1.781	1.670
breast-cancer	1.836‡	1.668	1.716‡	2.159‡	1.658	1.689	**2.262**	1.628	1.618
cmc.1	0.840‡	0.702	0.642	**1.201**	0.862	0.649	1.113‡	0.850	0.506
cmc.2	**1.093**	0.997†	0.552	1.021	0.847	0.571	0.990	0.849	0.426
cmc.3	0.444	0.386	0.495	0.980‡	0.740‡	0.346	**1.023**	0.754‡	0.427
german	1.120‡	0.924†	0.756	1.250‡	1.006	0.712	**1.259**	0.907	0.770
haberman	**1.050**	0.947	0.469	0.693	0.551	0.409	0.693	0.714	0.367
ionosphere	0.392	0.461	0.406	**0.959**	0.803†	0.573	0.868‡	0.910‡	0.593
iris.1	**2.818**	2.232	2.420	**2.818**	2.232	2.420	2.814‡	2.384	2.497
iris.2	**1.580**	1.266	1.011	1.040	0.888	0.399	1.073	1.006	0.393
iris.3	0.000	0.021	0.037	0.023	0.369†	**0.438**	0.116†	0.305‡	0.410†
mammographic	1.413‡	1.168	1.267‡	1.813‡	1.351	1.195	**1.840**	1.414	1.078
pageblocks.5	1.196‡	0.994	0.957	**1.486**	0.879	0.888	1.376‡	0.880	0.901
sonar	0.566‡	0.459†	0.597‡	0.683‡	0.673	0.549	**0.694**	0.655	0.453
spambase	**2.548**	2.102	2.102	2.505	1.997	1.714	2.496	1.966	1.594
spectf	**1.398**	1.019	0.799	1.268	0.923	0.509	1.251	1.105	0.454
tictactoe	**2.750**	2.235	2.255	2.660	1.993	1.868	2.637	2.079	1.892
transfusion	1.010	0.861‡	0.445	**1.023**	0.851†	0.465	0.952	0.915	0.497
wdbc	2.204	1.780	1.812‡	**2.224**	1.770	1.755	2.203‡	1.526	1.551
wine.1	2.148‡	1.864	1.837	**2.736**	2.264	2.082	2.719‡	2.255	2.263
wine.2	2.516	2.162	2.294‡	**2.591**	1.976	1.831	2.573‡	1.969	1.871
wine.3	2.349	1.981	2.216	2.473‡	1.925	1.581	**2.479**	1.969	1.664
wine-q-red	**1.947**	1.632	1.280	1.785	1.410	0.916	1.782	1.457	0.942
wine-q-white	1.547‡	1.362	1.054	1.729‡	1.233	0.998	**1.734**	1.368	1.006
yeast	0.715‡	0.540†	0.732‡	0.750‡	0.581	0.524	**0.862**	0.670	0.452
Average	1.543‡	1.295	1.236	**1.629**	1.282	1.091	1.629‡	1.306	1.068

(a) Goodness results for binary experiments using ACC.

	Population-based			Model-based			Combined		
	CI	CE	CT	CI	CE	CT	CI	CE	CT
balance.1	1.710‡	1.281	1.364	**2.305**	1.733	1.592	2.221‡	1.682	1.596
balance.3	2.373	1.776	1.823‡	2.484‡	1.825	1.654	**2.503**	1.648	1.653
breast-cancer	2.217	1.702	1.811‡	**2.315**	1.799	1.575	2.283‡	1.634	1.604
cmc.1	1.293†	0.896	0.858	**1.395**	1.215	0.706	1.381‡	1.023	0.746
cmc.2	**1.542**	1.253	0.842	1.537	1.143	0.751	1.509	1.112	0.701
cmc.3	**1.129**	0.777	0.623	0.968	0.701	0.460	0.992	0.795	0.486
german	1.708	1.279‡	1.029	1.693‡	1.372	0.835	**1.729**	1.388	0.641
haberman	0.698	0.452	0.459	1.187	0.957†	0.389	**1.224**	1.129‡	0.331
ionosphere	0.023	0.080	0.323‡	0.385†	**0.462**	0.383‡	0.297†	0.433‡	0.330†
iris.1	2.449‡	2.024	1.999	**2.753**	2.077	1.905	2.749‡	1.966	2.019
iris.2	**1.472**	1.102	0.921	1.357‡	1.128†	0.527	1.378‡	1.220‡	0.479
iris.3	**2.603**	2.074	1.964	2.201	1.628	1.431	2.211	1.682	1.324
mammographic	1.434‡	1.226	1.139	**1.952**	1.497	1.334	1.937‡	1.445	1.255
pageblocks.5	**2.235**	1.703	1.527	1.994	1.504	1.433	1.997	1.494	1.200
sonar	**0.796**	0.629	0.666†	0.490	0.365	0.466†	0.481	0.387	0.430†
spambase	**2.552**	1.963	1.608	2.540	2.036	1.651	2.537	2.072	1.620
spectf	1.035†	0.889	0.789	**1.444**	1.002	0.626	1.438	1.038	0.506
tictactoe	**2.483**	1.842	1.935	2.410	1.748	1.735	2.469	1.790	1.583
transfusion	1.310†	1.015†	0.738	1.541‡	1.078	0.701	**1.549**	1.224	0.762
wdbc	**2.506**	2.001	1.824	2.219	1.696	1.574	2.199	1.625	1.465
wine.1	2.102†	1.689	1.872‡	**2.552**	2.032	1.779	2.520	1.940	1.762
wine.2	**2.563**	1.897	1.936	2.517	1.808	1.808	2.496	2.034	1.431
wine.3	2.158†	1.585	1.785†	**2.510**	1.866	1.581	2.501†	1.626	1.657
wine-q-red	1.891	1.495	1.132	**1.896**	1.532	1.052	1.877	1.411	0.939
wine-q-white	1.435†	1.144	1.026	1.805†	1.423	0.989	**1.812**	1.285	0.985
yeast	0.136	0.101	0.281	0.568‡	0.454	0.418‡	**0.634**	0.569†	0.471†
Average	1.687†	1.303	1.241	**1.808**	1.388	1.129	1.805†	1.371	1.076

(b) Goodness results for binary experiments using PACC.

	Population-based			Model-based			Combined		
	CI	CE	CT	CI	CE	CT	CI	CE	CT
balance.1	1.865†	1.541	1.591†	2.390†	1.844	1.710	**2.399**	1.678	1.671
balance.3	**2.510**	1.924	1.721	2.465	1.663	1.816	2.405	1.726	1.706
breast-cancer	**2.627**	2.063	1.867	2.424	1.869	1.981	2.387	1.810	1.686
cmc.1	1.282‡	0.954	0.631	**1.543**	1.199	0.739	1.539‡	1.182	0.784
cmc.2	**1.558**	1.302	0.992	1.524	1.207	0.596	1.536	1.191	0.759
cmc.3	**1.314**	1.075	0.669	1.172	0.796	0.492	1.172	0.863	0.402
german	1.165†	0.842	0.819	1.494†	1.017	0.783	**1.540**	0.993	0.623
haberman	**1.556**	1.130	0.927	1.075	0.808	0.511	1.104	0.909	0.358
ionosphere	0.219	0.450	0.454	1.049†	**1.120**	0.762†	1.107‡	0.993‡	0.686†
iris.1	**2.811**	2.289	2.296	2.762	2.293	2.169	2.748	2.121	1.988
iris.2	1.329†	1.024†	1.059	1.410†	0.983	0.797	**1.415**	0.994	0.600
iris.3	1.718‡	1.486	1.492	**2.224**	1.611	1.491	2.216†	1.612	1.311
mammographic	1.447†	1.242	1.040	1.922†	1.344	1.298	**1.942**	1.304	1.181
pageblocks.5	1.769†	1.355	1.402‡	1.780	1.237	1.129	**1.867**	1.155	1.215
sonar	0.287†	0.158‡	0.227†	0.016	0.094†	**0.305**	0.016	0.056	0.289†
spambase	**2.603**	1.996	1.997	2.565	1.906	1.852	2.560	2.085	1.821
spectf	0.017	0.071	0.093	**1.356**	1.048	0.589	1.323†	0.994	0.659
tictactoe	**2.684**	2.208	2.219	2.586	2.154	1.890	2.577	2.061	1.664
transfusion	0.675	0.677	0.504	**1.460**	1.072	0.681	1.399	1.043	0.587
wdbc	**2.386**	1.831	1.748	2.379	1.860	1.734	2.376	1.840	1.786
wine.1	0.333	0.321	0.282	**2.500**	1.819	1.842	2.494‡	1.817	1.836
wine.2	**2.690**	2.055	1.747	2.516	1.814	1.654	2.492	1.800	1.616
wine.3	**2.741**	2.109	2.106	2.647	2.114	1.843	2.651	2.033	1.648
wine-q-red	1.966	1.575	1.186	**1.982**	1.452	1.195	1.952	1.437	1.078
wine-q-white	1.491‡	1.064	0.951	1.843‡	1.439	1.021	**1.849**	1.389	1.006
yeast	1.130†	0.929	0.771	1.382‡	1.128	0.859	**1.475**	1.117	0.761
Average	1.622‡	1.295	1.184	1.864†	1.419	1.221	**1.867**	1.392	1.143

(c) Goodness results for binary experiments using DM.

Table 7: Goodness results for binary experiments using ACC, PACC, and DM in terms of Log-Ratio Goodness (LRG).

	Population-based			Model-based			Combined		
	CI	CE	CT	CI	CE	CT	CI	CE	CT
hcv	0.332	0.323	0.292	0.523†	0.527†	**0.574**	0.524‡	0.501‡	0.571†
image_seg	10.378	9.596	9.843	**14.200**	10.099	10.219	13.318‡	9.981	8.156
phishing	1.555‡	1.245	1.114	**2.012**	1.951‡	0.950	1.989‡	1.925‡	0.949
page_block	0.893	1.128	1.028	**2.911**	2.598	2.419	2.796‡	2.319	2.152
chess	0.777	2.163†	0.485	1.361	3.321‡	0.571	1.553	**3.648**	0.676
mhr	0.883	0.947†	1.017	1.364†	**1.461**	0.946†	1.338	1.385‡	0.960
connect-4	1.401	1.523‡	0.701	1.346	1.487†	0.733	1.305	**1.554**	0.716
poker_hand	0.000	0.000	0.000	0.000	0.490†	**0.608**	0.000	0.271	0.364‡
molecular	**3.605**	3.001	2.930	3.593	2.947	3.152	3.466	2.897	2.946
waveform-v1	4.407	3.512	3.359	4.401‡	3.406	3.054	**4.416**	3.355	3.208
isolet	10.638‡	**12.750**	10.137†	11.121‡	12.487‡	11.713‡	11.766‡	10.986†	10.089†
cmc	1.198‡	1.178‡	0.730	**1.533**	1.332	0.764	1.483	1.119	0.811
shuttle	0.839‡	0.721‡	0.680†	1.204‡	1.170†	1.011‡	**1.258**	1.064‡	0.953‡
satellite	8.635‡	7.180	6.292	8.963‡	7.638	5.850	**9.372**	7.167	5.259
hand_digits	15.473‡	13.883	13.865	15.635‡	12.882	13.450	**15.796**	11.787	12.641
yeast	0.715‡	0.540†	0.732‡	0.750‡	0.581	0.524	**0.862**	0.670	0.452
nursery	6.235†	4.973	4.796	**6.339**	5.017	4.836	6.287‡	4.890	4.724
obesity	13.160‡	9.604	7.371	**13.675**	10.479	6.941	13.450‡	8.479	6.363
abalone	0.186	0.104	0.054	0.856	1.712‡	0.191	0.999	**1.848**	0.205
letter	7.898†	10.001‡	5.925	8.543‡	**10.393**	6.227	9.348‡	9.944‡	6.055
digits	14.990‡	13.035	13.723†	**15.312**	13.103	13.952‡	**15.312**	11.625	12.532
academic-success	**2.869**	2.518	1.876	2.852	2.447	1.901	2.866	2.368	1.809
wine-quality	0.069	0.074	0.000	0.579‡	0.746†	0.690‡	**0.767**	0.764‡	0.749‡
dry-bean	**15.220**	10.253	10.183	15.013‡	10.792	10.605	15.075‡	10.749	8.558
Average	5.098	4.594†	4.047	5.587‡	4.961‡	4.245	**5.639**	4.637‡	3.829

(a) Goodness results for multiclass experiments using ACC.

	Population-based			Model-based			Combined		
	CI	CE	CT	CI	CE	CT	CI	CE	CT
hcv	0.170	0.149	0.183	0.434‡	0.412‡	0.518‡	0.428‡	0.428‡	**0.522**
image_seg	12.887†	10.099	10.465	**14.753**	11.370	9.164	14.499‡	9.835	9.121
phishing	2.633‡	2.087†	2.112	**2.814**	2.394	1.725	2.795‡	2.279	1.528
page_block	1.896	2.105	1.686	**3.372**	2.687	2.496	2.905‡	2.394	2.676
chess	1.648	**3.335**	0.571	1.893	3.085‡	0.697	1.870	3.250‡	0.845
mhr	**2.491**	2.189	1.700	2.281	2.162	1.296	2.419	1.911	1.274
connect-4	2.977	2.458	2.140	2.928‡	2.418	2.137	**2.984**	2.298	1.936
poker_hand	0.000	0.000	0.215	0.279	0.230	**0.634**	0.232	0.227	0.610‡
molecular	3.629	2.987	3.088	**3.750**	3.226	3.156	3.710‡	2.847	2.956
waveform-v1	**4.617**	3.672	3.341	4.438	3.456	3.442	4.398	3.423	3.207
isolet	10.638†	**13.062**	11.172†	10.960†	12.473‡	11.055‡	11.444†	10.927†	9.799
cmc	0.146	0.266	0.401	1.301†	1.214‡	0.926†	**1.358**	1.348‡	0.861†
shuttle	2.003	2.219	2.389	**4.387**	3.908	2.360	4.362‡	3.832	2.084
satellite	**11.742**	8.340	6.448	11.430‡	8.562	6.520	11.405‡	7.471	5.853
hand_digits	15.312‡	13.652	13.696†	**15.635**	13.335	13.608†	**15.635**	11.934	12.252
yeast	0.136	0.101	0.281	0.568‡	0.454	0.418‡	**0.634**	0.569†	0.471‡
nursery	**6.652**	5.346	4.795	6.448†	5.302	4.737	6.509‡	4.802	4.525
obesity	8.897‡	7.651†	6.462	9.519‡	8.496‡	6.202	**9.556**	7.408†	5.835
abalone	0.532	0.709	0.179	1.741	**2.179**	0.533	1.550	2.092†	0.560
letter	7.898	9.737‡	6.591	8.865†	10.622‡	7.162	9.026†	**10.927**	6.988
digits	14.667‡	12.600	13.519†	**14.990**	12.801†	13.002†	**14.990**	11.093	12.190
academic-success	2.618‡	2.225	2.026	3.245‡	2.525	2.258	**3.248**	2.485	2.153
wine-quality	1.003†	1.335‡	0.586	1.191	**1.555**	0.620	1.232	1.459‡	0.645
dry-bean	15.381‡	11.433	10.507	**15.443**	10.450	10.729	15.305‡	9.536	9.055
Average	5.441	4.907†	4.356	**5.944**	5.222‡	4.391	5.937‡	4.782†	4.081

(b) Goodness results for multiclass experiments using PACC.

	Population-based			Model-based			Combined		
	CI	CE	CT	CI	CE	CT	CI	CE	CT
hcv	0.269	0.270	0.413†	0.553‡	0.592‡	**0.623**	0.521‡	0.577‡	0.602‡
image_seg	5.789	8.620	6.977	12.047‡	11.290‡	6.720	**12.176**	11.175‡	6.328
phishing	1.207†	1.246	1.312	2.010†	1.880†	1.590†	**2.094**	1.818	1.651†
page_block	0.794	0.872	1.239	2.428‡	**2.430**	2.305‡	2.070†	1.924‡	2.059‡
chess	1.459	2.533	0.608	2.884	4.716‡	0.915	2.890	**5.059**	1.017
mhr	**2.783**	2.286	1.907	2.412	2.289	1.599	2.368	2.063	1.575
connect-4	**3.258**	2.618	2.253	3.173	2.674	2.046	3.158	2.478	2.030
poker_hand	0.209	0.000	0.081	0.496‡	0.528†	**0.636**	0.473‡	0.527	0.615‡
molecular	**3.955**	3.204	3.284	3.839	3.233	3.310	3.843	3.248	3.149
waveform-v1	**4.607**	3.640	3.691	4.531	3.585	3.464	4.500	3.451	3.534
isolet	12.572	14.429†	10.281	13.217	14.628‡	9.229	13.378	**15.647**	8.036
cmc	0.336	0.506	0.446	2.011‡	1.776	1.175	**2.039**	1.828	1.069
shuttle	2.707†	3.388‡	2.376	3.920‡	**4.226**	2.027	3.922‡	4.042‡	1.928
satellite	**11.398**	7.900	6.295	11.298‡	7.606	6.186	10.745‡	7.825	6.034
hand_digits	14.990‡	12.887	11.921	14.990‡	12.745	11.212	**15.312**	11.964	10.099
yeast	1.130‡	0.929	0.771	1.382‡	1.128	0.859	**1.475**	1.117	0.761
nursery	6.370†	5.077	4.709	6.585‡	5.247	4.581	**6.630**	5.152	4.600
obesity	7.177	7.373	5.901	**10.495**	8.746‡	5.909	10.475‡	7.937	5.718
abalone	0.484	0.658	0.342	1.800	2.586†	0.826	1.661	**2.641**	0.790
letter	2.740	4.770‡	3.128†	3.707	**5.271**	3.687‡	3.707	4.804‡	2.895†
digits	**14.990**	11.491	11.114	**14.990**	13.771‡	9.554	**14.990**	12.914†	8.142
academic-success	3.318	2.626	2.301	3.394	2.817	2.407	**3.435**	2.641	2.300
wine-quality	0.394	0.376	0.482	1.062‡	**1.140**	0.702†	0.963	1.117‡	0.710†
dry-bean	14.822‡	10.276	10.484	**15.075**	10.881	9.579	14.730‡	9.984	8.756
Average	4.907	4.499	3.846	**5.762**	5.241‡	3.798	5.731‡	5.080‡	3.517

(c) Goodness results for multiclass experiments using DM.

Table 8: Goodness results for multiclass ACC, PACC, and DM in terms of Log-Ratio Goodness (LRG).