Published in final edited form as:

Proc SIAM Int Conf Data Min. 2016 May ; 2016: 261–269. doi:10.1137/1.9781611974348.30.

Binary Classifier Calibration Using an Ensemble of Linear Trend Estimation

Mahdi Pakdaman Naeini[*] and **Gregory F. Cooper**[†]

[*]Intelligent Systems Program, University of Pittsburgh

[†]Department of Biomedical Informatics, University of Pittsburgh

Abstract

Learning accurate probabilistic models from data is crucial in many practical tasks in data mining. In this paper we present a new non-parametric calibration method called *ensemble of linear trend estimation* (ELiTE). ELiTE utilizes the recently proposed ℓ_1 trend ltering signal approximation method [22] to find the mapping from uncalibrated classification scores to the calibrated probability estimates. ELiTE is designed to address the key limitations of the histogram binning-based calibration methods which are (1) the use of a piecewise constant form of the calibration mapping using bins, and (2) the assumption of independence of predicted probabilities for the instances that are located in different bins.

The method post-processes the output of a binary classifier to obtain calibrated probabilities. Thus, it can be applied with many existing classification models. We demonstrate the performance of ELiTE on real datasets for commonly used binary classification models. Experimental results show that the method outperforms several common binary-classifier calibration methods. In particular, ELiTE commonly performs statistically significantly better than the other methods, and never worse. Moreover, it is able to improve the calibration power of classifiers, while retaining their discrimination power. The method is also computationally tractable for large scale datasets, as it is practically $O(N \log N)$ time, where N is the number of samples.

1 Introduction

This paper focuses on developing a new non-parametric calibration method for post-processing the output of commonly used binary classification models to generate accurate probabilities. Obtaining accurate probabilities is crucial in many real world decision making and data mining problems. Decision theory provides a rationale basis for intelligent agents to make decisions [29] in which the utilities and probabilities are combined in determining the actions that maximize expected utility. In many of decision problems, the probabilities need to be well-calibrated in order to achieve the goal of finding the best action. The predicted probabilities of a forecaster are well-calibrated if they are close to the objective probabilities (i.e., the frequency of the events in the long run). More specifically, we say that a classification model is well calibrated if events predicted to occur with probability p do

occur about p fraction of the time, for all p. This concept applies to binary as well as multi-class classification problems.

Producing well-calibrated probabilistic predictions is critical in many areas of science (e.g., determining which experiments to perform), medicine (e.g., deciding which therapy to give a patient), business (e.g., making investment decisions), and many others. In data mining problems, obtaining well-calibrated classification models is crucial not only for decision-making, but also for combining the output of different classification models [3]. It is also useful when we aim to use the output of a classifier not only to discriminate the instances but also to rank them [36, 19, 14]. Research on learning well calibrated models has not been explored in the data mining literature as extensively as has, for example, learning models that have high discrimination (e.g., high accuracy).

There are two main approaches to obtaining well-calibrated classification models. The first approach is to build a classification model that is intrinsically well-calibrated *ab initio*. This approach can restrict the designer of the data mining model by requiring major changes in the objective function (e.g., using a different type of loss function) and can potentially increase the complexity of the associated optimization program that learns the model. The other approach is to rely on existing discriminative data mining models and then calibrate their output using post-processing methods. This approach has the advantage that it is general, flexible, and it frees the designer of a data mining algorithm from modifying the learning procedure and the associated optimization method [25]. However, this approach has the potential to decrease discrimination when increasing calibration, if care is not taken. The method we describe in this paper is shown empirically to improve calibration of different types of classifiers (e.g., LR, SVM, and NB) while maintaining their discrimination performance well.

In general, there are two main applications of postprocessing calibration methods. First, they can be used to convert the outputs of discriminative classification methods with no apparent probabilistic interpretation to posterior class probabilities [27]. An example is an SVM model that learns a discriminative model that does not have a direct probabilistic interpretation. In this paper, we show this use of calibration to map SVM outputs to well-calibrated probabilities. Second, calibration methods can be applied to improve the calibration of predictions of a probabilistic model that is miscalibrated. For example, a naïve Bayes (NB) model is a probabilistic model, but its class posteriors are often miscalibrated due to unrealistic independence assumptions [24]. The method we describe is shown empirically to improve the calibration of NB models without reducing their discrimination. The method can also work well on calibrating models that are less egregiously miscalibrated than are NB models.

2 Related work

Existing post-processing binary-classifier calibration models can be divided into parametric and non-parametric methods. Platt's method is an example of the former; it uses a sigmoid transformation to map the output of a classifier into a calibrated probability [27]. The two parameters of the sigmoid function are learned in a maximum-likelihood framework. The

method was originally developed to transform the output of an SVM model into calibrated probabilities. It has also been used to calibrate other type of classifiers [24]. The method runs in $O(1)$ at test time, and thus, it is fast. Its key disadvantage is the restrictive shape of sigmoid function that rarely fits the true distribution of the predictions [20].

A popular non-parametric calibration method is the equal frequency histogram binning model, which is also known as quantile binning [34]. In quantile binning, predictions are partitioned into B equal frequency bins. For each new prediction y that falls into a specific bin, the associated frequency of observed positive instances will be used as the calibrated estimate for $P(z = 1|y)$, where z is the true label of an instance that is either 0 or 1. Quantile binning can be implemented in a way that allows it to be applied to large scale data mining problems. Its limitations include (1) bins inherently pigeonhole calibrated probabilities into only B possibilities, (2) bin boundaries remain fixed over all predictions, (3) there is uncertainty in the optimal number of the bins to use, (4) predictions are independent within different bins, and (5) estimated probabilities will have abrupt changes at the boundary of the bins [35].

The most commonly used non-parametric classifier calibration method in machine learning and data mining applications is the *isotonic-regression-based calibration* (IsoReg) model [35]. To build a mapping from the uncalibrated output of a classifier to the calibrated probability, IsoReg assumes the mapping is an isotonic (monotonic) mapping following the ranking imposed by the base classifier. The commonly used algorithm for isotonic regression is the *Pool Adjacent Violators Algorithm* (PAVA), which is linear in the number of training data instances [2]. An IsoReg model based on PAVA can be viewed as a histogram binning model [35] where the position of the boundaries are selected by fitting the best monotone approximation to the train data according to the ordering imposed by the classifier. There is also a variation of the isotonic-regression-based calibration method for predicting accurate probabilities with a ranking loss [23]. In addition, an extension to IsoReg combines the outputs generated by multiple binary classifiers to obtain calibrated probabilities [37]. While IsoReg can perform well on some real datasets, the monotonicity assumption it makes can fail in real data mining applications[1] [25].

Recently, we introduced a new non-parametric Bayesian binary classifier calibration method called ABB [25]. ABB addresses the main drawbacks of the quantile binning method by considering all possible binning models induced by the training instances. In order to find calibrated probabilities, ABB applies Bayesian averaging over all possible binning models using the K2 Bayesian model scoring [8]. The main drawback of ABB is that it is computationally intractable for most real world applications, as it requires $O(N^2)$ computations for learning the model as well as $O(N^2)$ computations for computing the calibrated estimate for each of the test instances[2]. To address this problem, we introduced a new non-parametric calibration model called BBQ [25]. In order to find calibrated probability estimates, BBQ performs selective Bayesian averaging over a collection of

[1]In the limit, the correctness of the ranking imposed by the base classifier, is equivalent to presuming that the classifier has AUC equal to 1, which rarely happens in real world data mining applications.

[2]Note that the running time for the test instance can be reduced to $O(1)$ in any post-processing calibration model by using a simple caching technique that reduces the numerical precision of calibrated estimates in order to decrease calibration time [26]

different quantile binning models using the BDeU Bayesian scoring function [15]. BBQ requires $O(N \log N)$ computations for learning the calibration model and $O(\log(N))$ computations for computing the calibrated estimate for each of the test instances [25].

All of the above histogram binning-based calibration methods (IsoReg, ABB, and BBQ) have the following restrictions in common: (1) The ultimate generated calibration function that maps the uncalibrated classifier scores to the calibrated probabilities is piecewise constant, (2) The estimated probabilities will have abrupt changes at the boundary of the bins, and (3) Predictions are assumed to be independent within different bins. This paper presents a new binary classifier calibration method called *ensemble of linear trend estimation* (ELiTE) that extends the above histogram binning-based calibration methods by assuming that the calibration function is piecewise linear[3]. ELiTE cast the problem of finding the calibration mapping as a convex optimization problem. The resulting optimization program will be equivalent to the recently proposed ℓ_1 trend filtering signal approximation method [22]. It uses recently proposed alternating direction method of multipliers (ADMM) based optimization method to find a collection of a piecewise linear calibration mappings [28]. Finally, it uses the AICc scoring measure [5] to combine the predictions made by these models to yield more robust calibrated predictions for each of the test instances.

3 Method

In all the classifier calibration methods, the postprocessing step can be seen as a mapping function that transforms the outputs of a classification model to probabilities that are intended to be well-calibrated. In all of the histogram binning-based calibration models—including quantile binning [34], isotonic-regression-based calibration (IsoReg) [35], and our previous Bayesian extensions to the histogram binning, ABB and BBQ, [26, 25] — the generated mapping function will be a piecewise constant function. In this section we introduce the *ensemble of linear trend estimation* (ELiTE) calibration method that has the following three main advantages relative to all the above histogram binning-based calibration methods: (1) ELiTE assumes that the calibration mapping function is piecewise linear while the mapping found by quantile binning, IsoReg, ABB, and BBQ are always piecewise constant, (2) ELiTE removes the restrictive assumption that probability estimates are independent between the neighboring bins, and (3) ELiTE automatically finds the boundary of the bins through an optimization algorithm by trading off the best fit of the training instances for the tendency to follow the same trend in probability estimates. This trade-off will be controlled by a regularization parameter.

Before getting into the details of the method, we define some notation. Let y_i and z_i define respectively an uncalibrated classifier prediction and the true class of the i'th instance. In this paper, we focus on calibrating a binary classifier's output[4], and thus, $z_i \in \{0, 1\}$ and y_i

[3]It is possible to generalize the method to obtain piecewise polynomial calibration functions; however, we have noticed an over fitting to the training data by using piecewise polynomial degrees higher than 1.

[4]For classifiers that output scores that are not in the unit interval (e.g., SVM), we use a simple sigmoid transformation $f(x)=\frac{1}{1+exp(-x)}$ to transform the scores into the unit interval.

$\in [0; 1]$. Without loss of generality, we can assume that the instances are sorted based on the classifier scores y_i, so we have $y_1 < y_2 < \ldots < y_N$, where N is the total number of samples in the training data. Borrowing the term "bin" from the histogram binning literature, we define each bin as the largest interval over the training data with a uniform slope of change. The problem of finding an optimum piecewise linear calibration mapping can be formulated as the following optimization program:

$$\begin{aligned} \hat{\boldsymbol{p}} = \underset{\boldsymbol{p}\in R^N}{argmin} \quad & \tfrac{1}{2}\sum_{i=1}^{N}(p_i - z_i)^2 \\ \text{s.t.} \quad & \|\mathbf{v}\|_0 \leq B - 1 \end{aligned} \tag{3.1}$$

where $\|\mathbf{v}\|_0 = \sum_i 1(v_i \neq 0)$ is ℓ_0 norm defined as the number of nonzero elements of the vector $\mathbf{v}$. Also, the vector $\mathbf{v} \in R^{N-2}$ is defined as the second order finite difference vector associated with the training data [5] $\mathbf{v}_i = \frac{p_{i+2} - p_{i+1}}{y_{i+2} - y_{i+1}} - \frac{p_{i+1} - p_i}{y_{i+1} - y_i}$ and B is an optimization parameter that is defined as the maximum number of bins that we could have over all the training data (Thus, $B - 1$ shows the number of change points or kinks in the calibration mapping function). The above optimization program tries to keep the estimated probability p_i close to z_i, the true class of the corresponding training instance, while the program constrains the number of kinks or change points in the slope of the calibration mapping [6]. Solving the above optimization program is intractable and requires combinatorial optimization methods [22]. A natural convex relaxation of this problem can be obtained by substituting the ℓ_0 norm with the ℓ_1 norm using the sparsity property of the ℓ_1 norm. After relaxing the ℓ_0 norm, it is possible to rewrite the resulting constrained optimization program in the following equivalent Lagrangian form:

$$\hat{\boldsymbol{p}} = \underset{\boldsymbol{p}\in R^N}{argmin} \quad \tfrac{1}{2}\sum_{i=1}^{N}(p_i - z_i)^2 + \lambda\|\mathbf{v}\|_1 \tag{3.2}$$

Where $\hat{\boldsymbol{p}} = (\hat{p}_1, \ldots, \hat{p}_n)$ is the vector of calibrated probability estimates and $\|\mathbf{v}\|_1 = \sum_{i=1}^{N} \left| \frac{p_{i+2} - p_{i+1}}{y_{i+2} - y_{i+1}} - \frac{p_{i+1} - p_i}{y_{i+1} - y_i} \right|$. Also, λ is a positive real number that regulates the trade-off between the complexity of the model and the goodness of fit by penalizing the total variation over the slope of the resulting calibration mapping function. The above optimization program is equivalent to the ℓ_1 (linear) trend filtering signal approximation model [22]. The linear trend filtering itself is a special case of the recently introduced adaptive piecewise polynomial trend filtering model [33] [7].

The piecewise linear trend filtering estimation has the following properties that make it an attractive choice for estimating the calibration mapping function: (1) The final solution to

[5] Note that an element of $\mathbf{v}$ is zero when the slope remains the same between two successively predicted points.
[6] If some of the training instances obtain equal classification scores, they will be replaced by an instance with the target value z that is equal to the average of their corresponding z_i. In this case, we form a weighted objective in the optimization program in Equation 3.1
[7] Note that the adaptive piecewise polynomial trend filtering model is itself a special case of generalized lasso problem [32]

the optimization program $\hat{\mathbf{p}}$ will be a continuous piecewise linear function with the change points occurring on the training data [22], so the final calibration mapping function will be a continuous function of uncalibrated scores y_i, and the estimated probabilities will not have any abrupt changes at the boundary of the bins, (2) Due to shrinkage property of the lasso-based penalties, the final probability estimates in two neighboring bins will shrink toward each other [32], as a result it will relax the restrictive independence assumption made in histogram binning-based calibration models, (3) The solution path to the optimization program in Equation 3.2 is piecewise linear with respect to the regularization parameter λ. This will make it computationally efficient to find the entire path of the solutions to the trend filtering problem for small sample sizes [32].

There are a few different methods to solve the trend filtering optimization problem: It is possible to convert the trend filtering problem into the standard lasso problem and then use the LARS algorithm to find all the solution paths with respect to λ [11, 32]. It is also possible to cast the problem as a special case of the generalized lasso signal approximation [32], and then derive the dual program and utilize the piecewise linearity property of the solution path in the dual program to find the entire path of the solutions [32]. However, these two methods do not scale well for large N [33]. Another approach is to use coordinate descent methods to solve the dual program of the generalized lasso problem [33]. There are two specialized optimization methods designed to solve the trend filtering optimization problem. The first method is based on the specialized interior-point method optimization that was proposed by Kim et al. [22]. The method requires $O(N)$ computations to solve a banded linear system of equations in each iteration of the interior-point optimization algorithm; in the worst-case it will solve the trend filtering for a single value of λ in $O(N^{1.5})$. However, the authors claim that in practice the interior-point method converges in tens of iterations, in which case the general running time for solving the optimization problem will still be $O(N)$. The other specialized optimization method for trend filtering problem is recently proposed by A. Ramdas et al. [28]. They introduced a specialized *alternating direction method of multipliers* (ADMM), and they showed that their method has better scalability and faster convergence rate for large scale problems compared to the interior-point based method, while on the small sample sizes they have similar performance to the interior-point based method proposed by Kim et al. [22]. In our implementation of ELiTE, we use the specialized ADMM optimization method [8]. For the sake of completeness, we briefly describe the method; more detailed information about the algorithm and the derivations can be found in A. Ramdas et al. [28].

In order to solve the trend filtering problem in Equation 3.2, the specialized ADMM method introduces a new auxiliary parameter $\boldsymbol{\alpha}$ to rewrite the unconstrained optimization program in Equation 3.2 as the following constrained optimization program:

$$\hat{\boldsymbol{p}} = \underset{\boldsymbol{p}\in R^N, \alpha\in R^{N-1}}{argmin} \quad \frac{1}{2}\|\boldsymbol{p}-z\|_2^2+\lambda\|D_{N-1}\boldsymbol{\alpha}\|_1 \quad \text{s.t.} \quad \alpha = A\mathbf{p}, \tag{3.3}$$

[8]The specialized ADMM code is publicly available at https://github.com/statsmaths/glmgen

where $D_k \in R^{k-1 \times k}$ is defined as follows:

$$D_k = \begin{bmatrix} -1 & 1 & 0 & \dots & 0 & 0 \\ 0 & -1 & 1 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & -1 & 1 \end{bmatrix},$$

and, $A = diag(\frac{1}{y_2 - y_1}, \frac{1}{y_3 - y_2}, \dots, \frac{1}{y_N - y_{N-1}}) D_N$.

The corresponding augmented Lagrangian of the optimization program in Equation 3.3 will be as $L(\mathbf{p}, \boldsymbol{\alpha}, \mathbf{u}) = \frac{1}{2}\|\boldsymbol{p} - \boldsymbol{z}\|_2^2 + \lambda \|D_{N-1}\boldsymbol{\alpha}\|_1 + \frac{\rho}{2}\|\boldsymbol{\alpha} - A_{\mathbf{p}} + \mathrm{u}\|_2^2 - \frac{\rho}{2}\|\mathbf{u}\|_2^2$ Using the augmented Lagrangian and performing some calculations [4], the ADMM iterations will be as follows:

$$\begin{gathered} \mathbf{p} \leftarrow (I + \rho A^T A)^{-1}(\mathbf{z} + \rho A^T(\boldsymbol{\alpha} + \mathbf{u})) \\ \boldsymbol{\alpha} \leftarrow \underset{\alpha \in R^{N-1}}{argmin} \tfrac{1}{2}\|A\mathbf{p} - \mathbf{u} - \boldsymbol{\alpha}\|_2^2 + \tfrac{\lambda}{\rho}\|D_{N-1}\boldsymbol{\alpha}\|_1 \\ \mathbf{u} \leftarrow \mathbf{u} + \alpha - A\mathbf{p} \end{gathered}$$

.

The tricky part in the above sequential updates is the second equation related to updating the value of $\boldsymbol{\alpha}$. It requires solving an optimization problem that is equivalent to the fused lasso signal approximation [31]. In implementing the specialized ADMM method, Ramdas et al. used a computationally efficient dynamic programming method proposed by N. Johnson [21] that finds the fused lasso solution in $O(N)$. They reported that ADMM iterations converge in a constant number of iterations. As a result, the ultimate time for finding the trend filtering solution will still be $O(N)$ [28].

ELiTE employs the specialized ADMM optimization method just described to generate a collection of trend filtering models (one for each value of λ ranging equally in the log space from λ_{max} to $\lambda_{max} * 10^{-4}$, where λ_{max} is the corresponding value of λ that gives the best affine approximation of the calibration mapping that is $\lambda_{max} = \|(D_{N-1} A A^T D_{N-1}^T)^{-1} D_{N-1} A_{\boldsymbol{z}}\|_\infty$ [22]). It then uses the Akaike information criterion with a correction for finite sample sizes (AICc) [5] to score each of the models [9]. We use the unbiased estimate of the degree of freedom for each linear trend filtering model as the effective number of parameters in computing the scores [33]. Assume ELiTE yields the piecewise linear calibration models M_1, M_2 …, M_T, where T is the total number of generated models by changing λ (in our experiments $T = 50$). For any new classifier output y, the calibrated prediction in the *ELiTE* model is defined using the following weighted averaging [16]:

[9] We also tried BIC and AIC model scoring functions. The AIC scoring shows extreme overfitting to the training data, while BIC results were comparable to AICc scoring. We finally chose AICc since it performed slightly better than BIC in general.

$$P(z=1|y)=\sum_{i=1}^{T}\frac{Score(M_i)}{\sum_{j=1}^{T}Score(M_j)}P(z=1|y,M_i),$$

where $P(z=1|y,M_i)$ is the probability estimate obtained using the trend filter model M_i for the uncalibrated classifier output y. Also, $Score(M_i)$ is obtained using the AICc scoring function [30].

4 Experimental Setup

This section describes the set of experiments that we performed to evaluate the performance of the ELiTE calibration method in comparison to other commonly used calibration methods. The comparison methods include quantile binning [34], Platt's method[27], isotonic regression[35], and BBQ, which is a Bayesian extension to the quantile binning method [25]. We did not include ABB in our experiments mainly because it is not computationally tractable for datasets that have more than couple of thousands of instances. Moreover, even for small size datasets, we noticed that ABB performs similarly to BBQ.

In order to evaluate the performance of the calibration methods, we use 5 different evaluation measures. We use Accuracy (Acc) and *area under ROC curve* (AUC) to evaluate how well the methods discriminate the positive and negative instances in the feature space. We also utilize the three measures of calibration: *root mean square error* (RMSE), *maximum calibration error* (MCE), and *expected calibration error* (ECE) [10] [25, 26].

MCE and ECE are computed by partitioning the output space of the binary classifier, which is the interval [0; 1], into K fixed number of bins (K = 10 in our experiments). The estimated probability for each instance is located in one of the bins. For each bin we can define the associated calibration error as the absolute difference between the mean value of the predictions and the actual observed frequency of positive instances. The MCE calculates the maximum calibration error among the bins, and ECE calculates expected calibration error over the bins, using empirical estimates. The lower the values of MCE and ECE, the better the calibration of a model [25, 26].

We used three common classifiers, Logistic Regression (LR), Support Vector Machines (SVM), and Naïve Bayes (NB) to evaluate the performance of ELiTE. In the experiments, we used the average over 10 random runs of 10-fold cross validation, and we always used the training data for calibrating the models.

5 Experimental Results

We ran two sets of experiments on 35 binary outcome classification datasets from the UCI and LibSVM repositories[11] [1, 6]. In the first set of experiments we were interested in evaluating if there is experimental support for using ELiTE as a post-processing calibration

[10]Note that, to be more precise, RMSE evaluates both calibration and refinement of the predicted probabilities. Refinement accounts for the usefulness of the probabilities by favoring those that are either close to 0 or 1 [9, 7]

method. Table 1 shows the 95% confidence interval for the mean of the random variable X, which is defined as the percentage of the gain (or loss) of ELiTE with respect to the base classifier:

$$X = \frac{measure_{elite} - measure_{method}}{measure_{method}}, \quad (5.4)$$

where *measure* is one of the evaluation measures AUC, ACC, ECE, MCE, or RMSE. Also, *method* denotes one of the choices of the base classifiers, namely, LR, SVM, or NB. For instance, Table 1 shows that by post-processing the output of SVM using ELiTE, we are 95% confident to gain anywhere from 16% to 30% average improvement in terms of RMSE. This could be a promising result, depending on the application, considering the 95% CI for the AUC which shows that by using ELiTE we are 95% confident not to lose more than 1% of the SVM discrimination power in terms of AUC (Note also that the CI includes zero, which indicates that there is not a statistically significant difference between the performance of SVM and ELiTE in terms of AUC).

Overall, the results in Table 1 show that there is not a statistically meaningful difference between the performance of ELiTE and the base classifiers in terms of AUC. The results support at a 95% confidence level that ELiTE improves the performance of the LR and NB base classifiers in terms of ACC. Furthermore, the results in Table 1 show that by post-processing the output of LR, SVM, and NB using ELiTE, we can make dramatic improvements in terms of calibration measured by RMSE, ECE, and MCE. For instance, the results indicate that at a 95% confidence level, ELiTE improved the average performance of NB in terms of ECE anywhere from 27% to 55%, which could be practically significant in many decision-making and data mining applications.

In the second set of experiments on real data, we are interested in evaluating the performance of ELiTE compared with the base classifier and other calibration methods. To evaluate the performance of models, we used the recommended statistical test procedure by Janez Demsar [10]. More specifically, we used the non-parametric testing method based on the F_F test statistics [18], which is an improved version of Freidman non-parametric hypothesis testing method [13], followed by Holm's step-down procedure [17] to evaluate the performance of ELiTE in comparison with other methods, across the 35 baseline datasets.

The results on real datasets are shown in the Figures 1-5. In these graphs, we indicate the average rank of each method (1 is best) and we connect the methods that are statistically equivalent with our target method ELiTE using a horizontal bar (e.g., in Figure 3a the average rank of ELiTE is 1.89, and it is performing statistically equivalent to isoreg in terms of RMSE; however, its performance in terms of RMSE is statistically superior to Hist,

[11]The datasets used were as follows: spect, adult, breast, page-blocks, pendigits, ad, australian, colon cancer, letter unbalanced, letter balanced, diabetes, duke, fourclass, german numer, gisette scale, heart, ionosphere scale, liver disorders, mushrooms, sonar scale, splice, svmguide1, svmguide3, coil2000, balance, breast cancer, w1a, thyroid sick, scene, uscrime, solar, car34, car4, mamography, satimage.

Platt's method, BBQ, and the base classifier LR). Figure 1 shows the results of comparing the AUC of ELiTE with other methods. As shown, ELiTE performs significantly better than all other calibration methods in terms of AUC at a confidence level of $\alpha = 0.05$. Also, its performance in terms of AUC is always statistically equivalent to the base classifier (LR, SVM, NB). Note that we did not include Platt's method in our statistical test for AUC, since the AUC of the Platt's method would be the same as the AUC of the base classifier; this pattern occurs because Platt's method always uses a monotonic mapping of the base classifier's output as the calibrated score.

Figure 2 shows the results of comparing ACC of ELiTE with the other methods. As shown, ELiTE performs statistically better than histogram binning and Platt's method, as well as the base classifiers NB, and LR. However, ELiTE is statistically equivalent to BBQ and IsoReg, as well as the base classifier SVM, in our experiments over 35 real datasets. Figure 3 shows the results of our experiments in comparing the performance of ELiTE with other calibration methods in terms of RMSE. ELiTE always outperforms the base classifier and all other calibration methods. However, its difference with isotonic regression is not statistically significant, when the base classifier is LR or NB.

Figures 4, and 5 show the results of comparing ELiTE performance with the others in terms of ECE and MCE, respectively. They show that ELiTE performs superior to all other calibration methods and to the base classifier, in terms of ECE and MCE. However, its difference with BBQ is not statistically significant in terms of ECE when the base classifier is SVM or NB. Also, in terms of MCE, the difference between ELiTE and BBQ is not statistically significant when SVM is used as the base classifier.

Overall, in terms of discrimination measured by AUC and ACC, the results show that the proposed non-parametric calibration method either outperforms the other calibration methods or has a performance that is not statistically significantly different from the other methods and the base classifier. In terms of calibration performance, ELiTE is often statistically superior to the other methods and is never statistically significantly worse.

6 Conclusion

In this paper, we presented a new non-parametric binary classifier calibration method called *ensemble of linear trend estimation* (ELiTE) [12] that generalizes all the histogram binning-based calibration methods. ELiTE assumes that the calibration mapping function is piecewise linear while the mapping found by quantile binning, IsoReg, ABB, and BBQ are always piecewise constant. The method is computationally tractable, as it runs in $O(N \log N)$ for N training instances. It can be used to calibrate many different types of binary classifiers, including logistic regression, support vector machines, naïve Bayes, and others. Our experiments show that by post-processing the output of classifiers using ELiTE, we can gain high calibration improvement in terms of RMSE, ECE, and MCE, without losing any statistically meaningful discrimination performance. Moreover, our experimental evaluation

[12] An implementation of ELiTE method will be made publicly available at the following address: https://github.com/pakdaman/calibration.git

on a broad range of real datasets shows that ELiTE outperforms other commonly used binary classifier calibration methods as well as BBQ (our recently introduced Bayesian extension to the quantile binning method) [25].

An important advantage of ELiTE over BBQ is that it can be naturally extended to multi-class and multi-label calibration models, similar to what has been done for the standard IsoReg [35]. This is an area of our current research. We also plan to investigate theoretical properties of ELiTE. We are interested to utilize the minimax properties of the piecewise polynomial trend filtering method [33] to find theoretical guarantees regarding the discrimination and calibration performance of ELiTE, similar to what has been proved for the AUC guarantees of IsoReg [12].

Acknowledgments

Research reported in this publication was supported by grant U54HG008540 awarded by the National Human Genome Research Institute through funds provided by the trans-NIH Big Data to Knowledge (BD2K) initiative. The content is solely the responsibility of the authors and does not necessarily represent the official views of the National Institutes of Health.

References

1. Bache K, Lichman M. UCI Machine Learning Repository. 2013
2. Barlow, Richard E., Bartholomew, David J., Bremner, JM., Brunk, H Daniel. Statistical Inference Under Order Restrictions: The Theory and Application of Isotonic Regression. Wiley; New York: 1972.
3. Bella, Antonio, Ferri, Cèsar, Hernández-Orallo, José, Ramírez-Quintana, María José. On the effect of calibration in classifier combination. Applied Intelligence. 2013; 38(4):566–585.
4. Boyd, Stephen, Parikh, Neal, Chu, Eric, Peleato, Borja, Eckstein, Jonathan. Distributed optimization and statistical learning via the alternating direction method of multipliers. Foundations and Trends® in Machine Learning. 2011; 3(1):1–122.
5. Cavanaugh, Joseph E. Unifying the derivations for the Akaike and corrected Akaike information criteria. Statistics & Probability Letters. 1997; 33(2):201–208.
6. Chang, Chih-Chung, Lin, Chih-Jen. Libsvm: A library for support vector machines. ACM Transactions on Intelligent Systems and Technology. 2011; 2(3):27.
7. Cohen, Ira, Goldszmidt, Moises. Properties and benefits of calibrated classifiers. Knowledge Discovery in Databases: PKDD 2004. 2004:125–136.
8. Cooper, Gregory F., Herskovits, Edward. A Bayesian method for the induction of probabilistic networks from data. Machine learning. 1992; 9(4):309–347.
9. DeGroot MH, Fienberg SE. The comparison and evaluation of forecasters. The Statistician. 1983:12–22.
10. Demšar, Janez. Statistical comparisons of classifiers over multiple data sets. The Journal of Machine Learning Research. 2006; 7:1–30.
11. Efron, Bradley, Hastie, Trevor, Johnstone, Iain, Tibshirani, Robert. Least angle regression. The Annals of Statistics. 2004; 32(2):407–499.
12. Fawcett, Tom, Niculescu-Mizil, Alexandru. PAV and the ROC convex hull. Machine Learning. 2007; 68(1):97–106.
13. Friedman, Milton. The use of ranks to avoid the assumption of normality implicit in the analysis of variance. Journal of the American Statistical Association. 1937; 32(200):675–701.
14. Hashemi, Homa B., Yazdani, Nasser, Shakery, Azadeh, Naeini, Mahdi Pakdaman. Application of ensemble models in web ranking. 5th International Symposium on Telecommunications; 2010. p. 726-731.

15. Heckerman D, Geiger D, Chickering DM. Learning Bayesian networks: The combination of knowledge and statistical data. Machine Learning. 1995; 20(3):197–243.
16. Hoeting, Jennifer A., Madigan, David, Raftery, Adrian E., Volinsky, Chris T. Bayesian model averaging: a tutorial. Statistical Science. 1999:382–401.
17. Holm, Sture. A simple sequentially rejective multiple test procedure. Scandinavian Journal of Statistics. 1979:65–70.
18. Iman, Ronald L., Davenport, James M. Approximations of the critical region of the friedman statistic. Communications in Statistics-Theory and Methods. 1980; 9(6):571–595.
19. Jiang, Liangxiao, Zhang, Harry, Su, Jiang. Learning k-nearest neighbor Naïve Bayes for ranking. Advanced Data Mining and Applications. 2005:175–185.
20. Jiang X, Osl M, Kim J, Ohno-Machado L. Calibrating predictive model estimates to support personalized medicine. Journal of the American Medical Informatics Association. 2012; 19(2): 263–274. [PubMed: 21984587]
21. Johnson, Nicholas A. A dynamic programming algorithm for the fused lasso and ℓ_0 segmentation. Journal of Computational and Graphical Statistics. 2013; 22(2):246–260.
22. Kim, Seung-Jean, Koh, Kwangmoo, Boyd, Stephen, Gorinevsky, Dimitry. ℓ_1 trend filtering. SIAM Review. 2009; 51(2):339–360.
23. Menon, Aditya, Jiang, Xiaoqian, Vembu, Shankar, Elkan, Charles, Ohno-Machado, Lucila. Predicting accurate probabilities with a ranking loss. International Conference on Machine Learning; 2012. p. 703-710.
24. Niculescu-Mizil, A., Caruana, R. Predicting good probabilities with supervised learning. International Conference on Machine Learning; 2005. p. 625-632.
25. Naeini, Mahdi Pakdaman, Cooper, Gregory, Hauskrecht, Milos. Obtaining well calibrated probabilities using Bayesian binning. Twenty-Ninth AAAI Conference on Artificial Intelligence; 2015.
26. Naeini, Mahdi Pakdaman, Cooper, Gregory F., Hauskrecht, Milos. Binary classifier calibration using a Bayesian non-parametric approach. SIAM Data Mining (SDM). 2015
27. Platt, John C. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. Advances in Large Margin Classifiers. 1999; 10(3):61–74.
28. Ramdas, Aaditya, Tibshirani, Ryan J. Fast and flexible admm algorithms for trend filtering. arXiv preprint arXiv:1406 2082. 2014
29. Russell, Stuart Jonathan, Norvig, Peter, Davis, Ernest, Russell, Stuart Jonathan, Russell, Stuart Jonathan. Artificial intelligence: A Modern Approach. Vol. 2. Prentice hall; Englewood Cliffs: 2010.
30. Schwarz, Gideon, et al. Estimating the dimension of a model. The Annals of Statistics. 1978; 6(2): 461–464.
31. Tibshirani, Robert, Saunders, Michael, Rosset, Saharon, Zhu, Ji, Knight, Keith. Sparsity and smoothness via the fused lasso. Journal of the Royal Statistical Society: Series B (Statistical Methodology). 2005; 67(1):91–108.
32. Tibshirani, Ryan, Taylor, Jonathan. The solution path of the generalized lasso. The Annals of Statistics. 2011; 39(3):1335–1371.
33. Tibshirani, Ryan J. Adaptive piecewise polynomial estimation via trend filtering. The Annals of Statistics. 2014; 42(1):285–323.
34. Zadrozny, B., Elkan, C. Obtaining calibrated probability estimates from decision trees and naïve Bayesian classifiers. International Conference on Machine Learning; 2001. p. 609-616.
35. Zadrozny, B., Elkan, C. Transforming classifier scores into accurate multiclass probability estimates. SIGKDD International Conference on Knowledge Discovery and Data Mining; 2002. p. 694-699.
36. Zhang, Harry, Su, Jiang. Naïve Bayesian classifiers for ranking. Machine Learning: ECML 2004. 2004:501–512.
37. Zhong, Leon Wenliang, Kwok, James T. Accurate probability calibration for multiple classifiers. Twenty-Third International Joint Conference on Artificial Intelligence; 2013. p. 1939-1945.

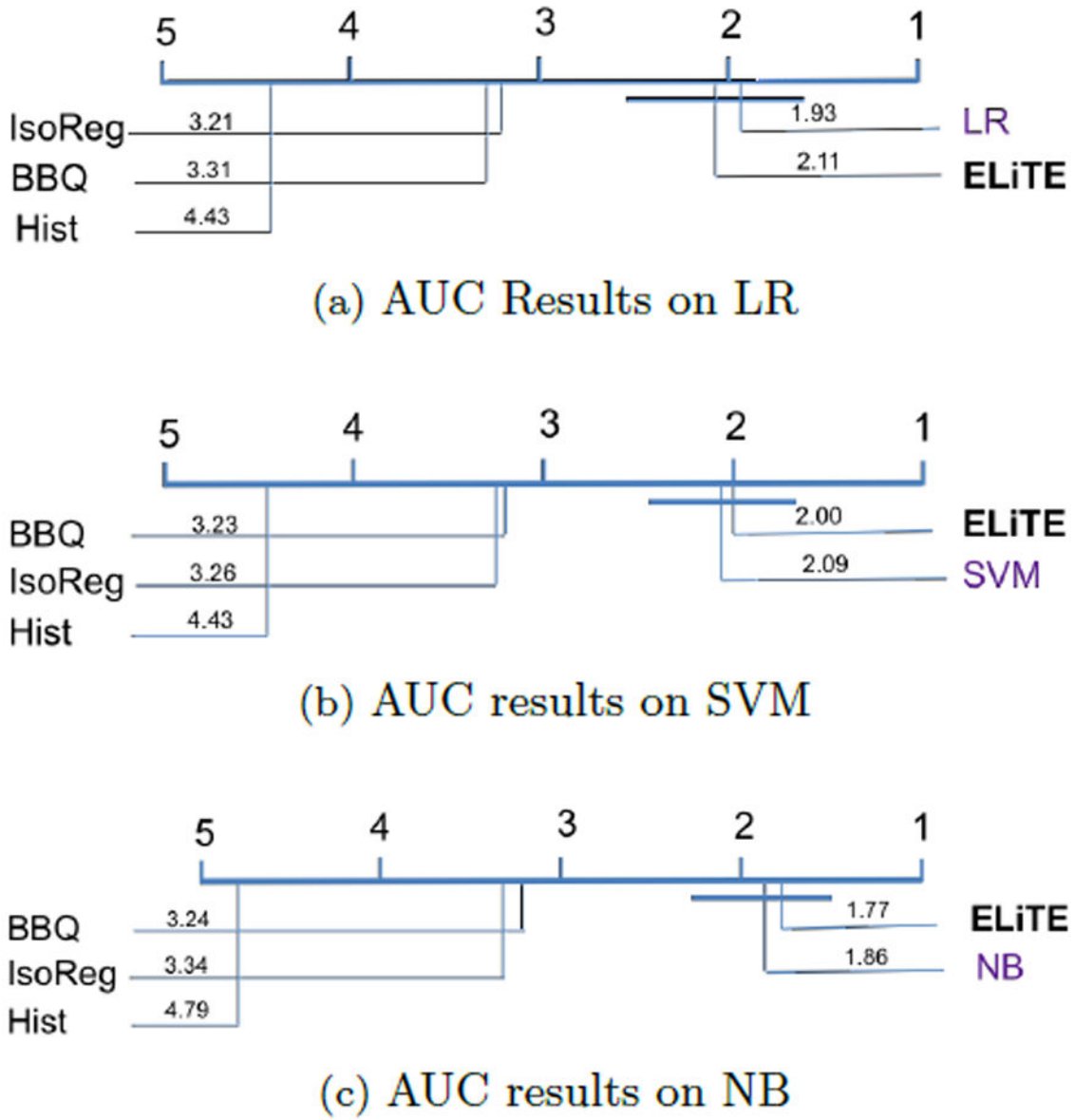

Figure 1.

Performance of each method in terms of average rank of AUC on the real datasets. All the methods which are not connected to ELiTE by the horizontal bar are statistically significantly worse than ELiTE (using an improved Friedman test followed by Holm's step-down procedure at a 0.05 significance level).

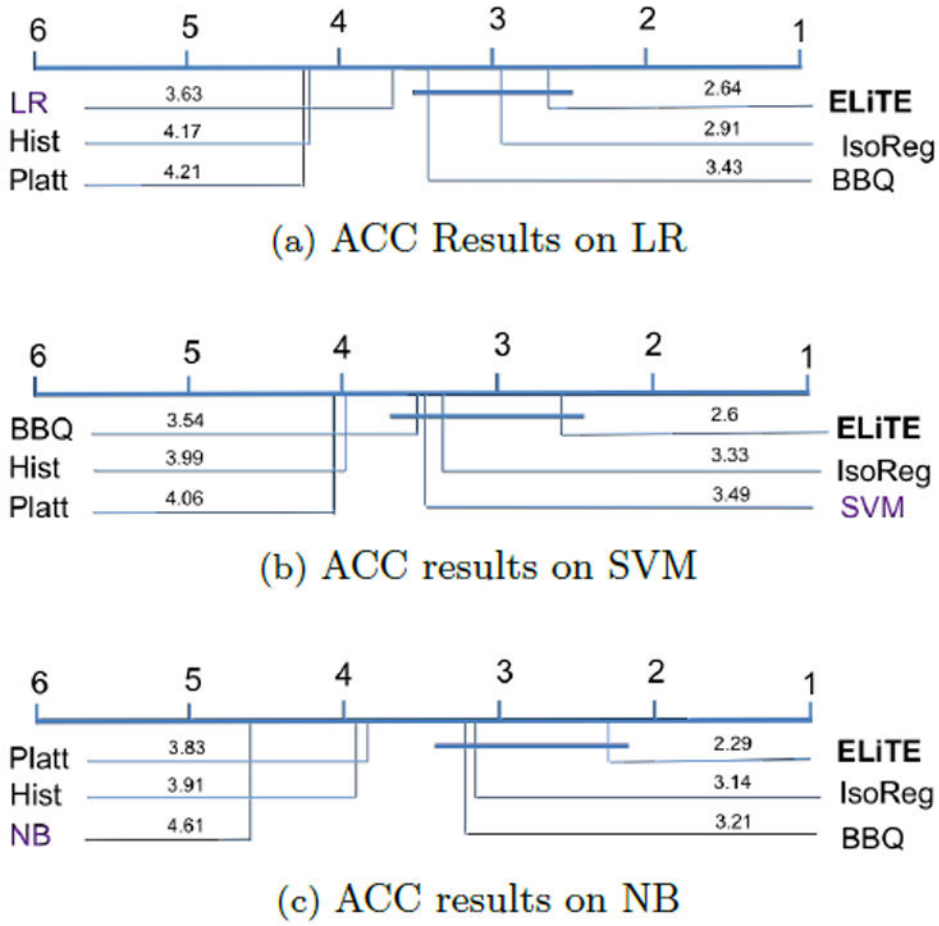

Figure 2.

Performance of each method in terms of average rank of ACC on the real datasets. All the methods which are not connected to ELiTE by the horizontal bar are statistically significantly worse than ELiTE (using an improved Friedman test at a 0.05 significance level).

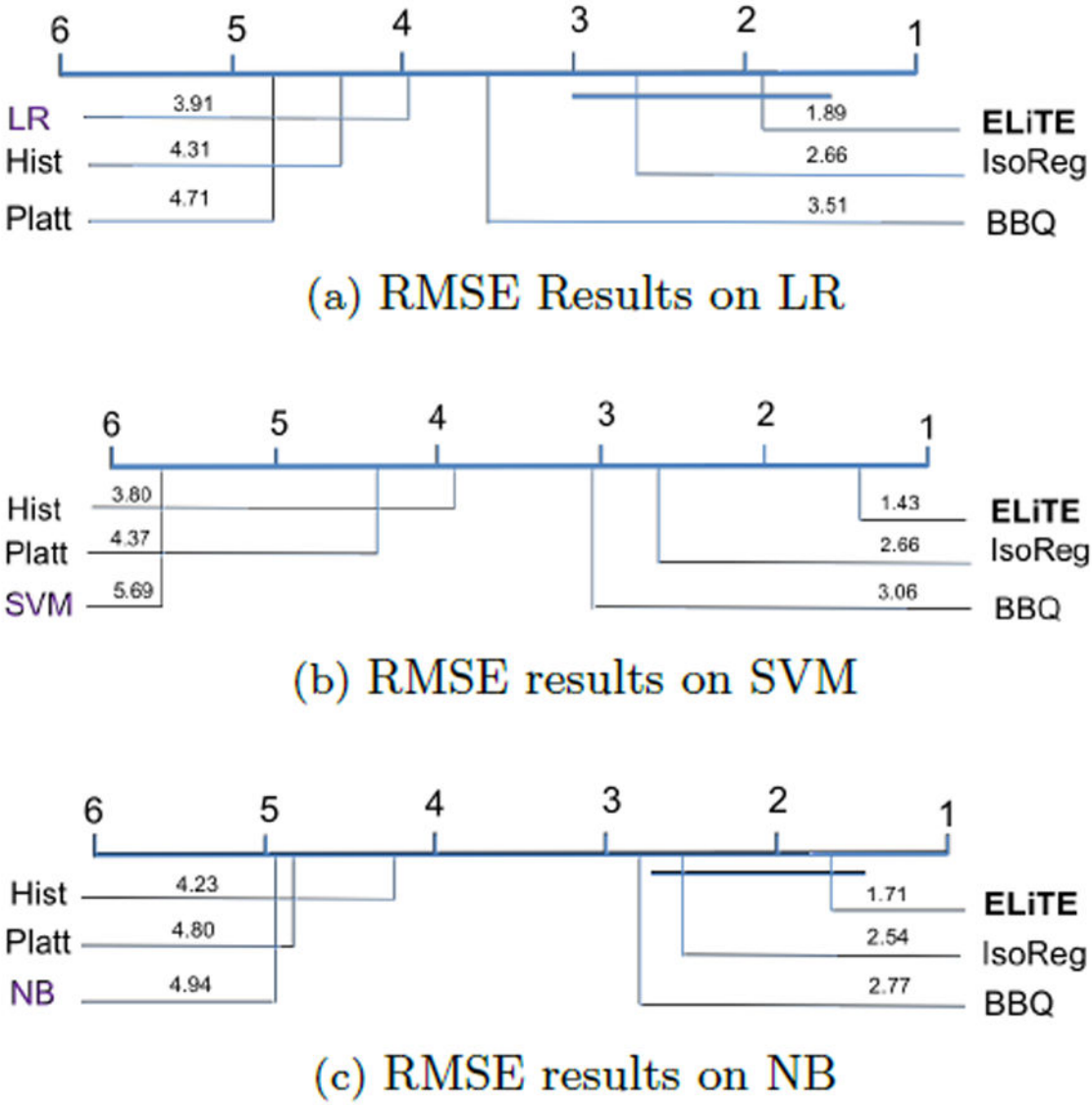

Figure 3.

Performance of each method in terms of average rank of RMSE on the real datasets. All the methods which are not connected to ELiTE by the horizontal bar are statistically significantly worse than ELiTE (using an improved Friedman test followed by Holm's step-down procedure at a 0.05 significance level).

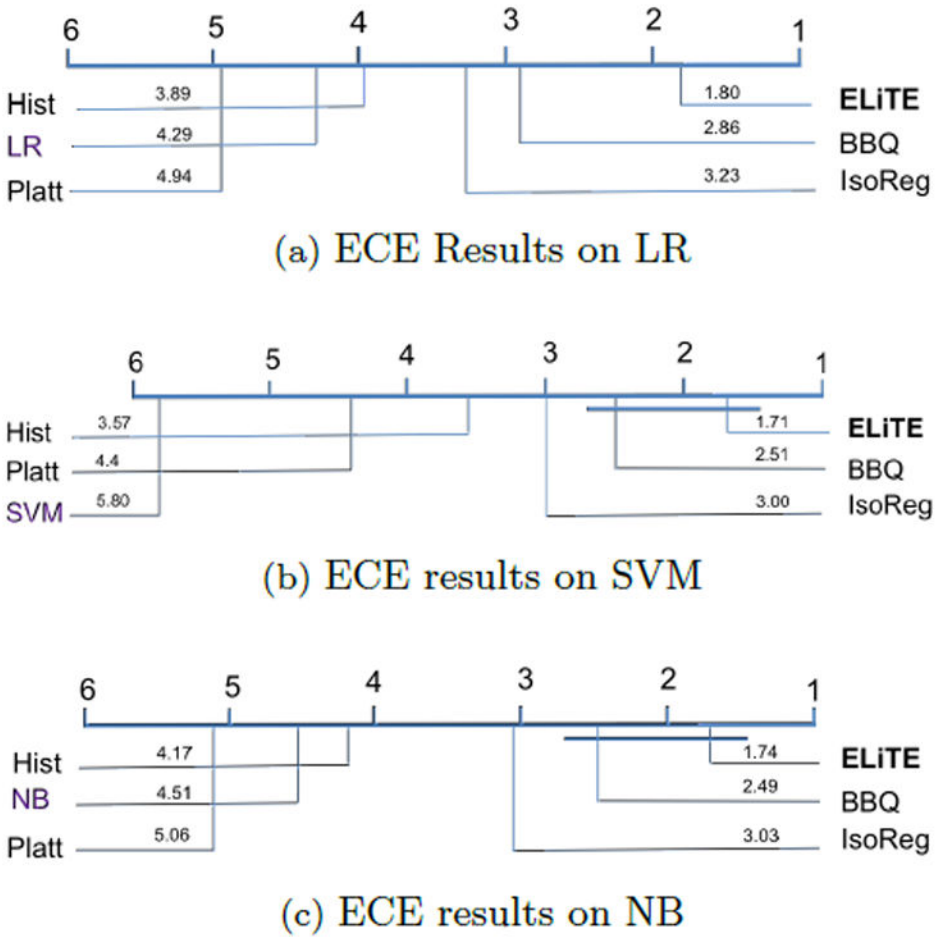

Figure 4.

Performance of each method in terms of average rank of ECE on the benchmark datasets. All the methods which are not connected to ELiTE by the horizontal bar are statistically significantly worse than ELiTE (using an improved Friedman test followed by Holm's step-down procedure at a 0.05 significance level).

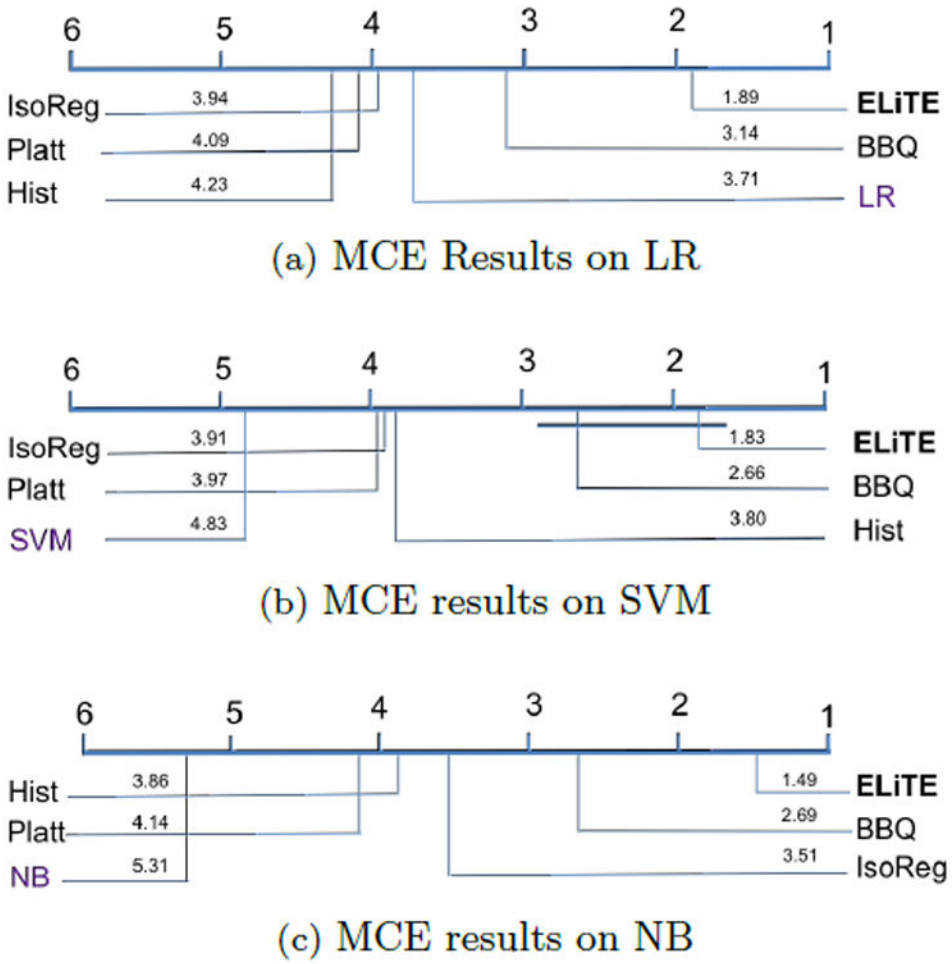

Figure 5.

Performance of each method in terms of average rank of MCE on the benchmark datasets. ELiTE is almost always statistically superior to all other competing methods (using the improved Friedman test followed by Holm's step-down procedure at a 0.05 significance level).

Table 1

The 95% confidence interval for the average percentage of improvement over the base classifiers (LR, SVM, NB) by using the ELiTE method for post-processing. Positive entries for AUC and ACC mean ELiTE is on average performing better discrimination than the base classifiers. Negative entries for RMSE, ECE, and MCE mean that ELiTE is on average performing better calibration than the base classifiers.

	LR	SVM	NB
AUC	[-0.01 , 0.01]	[-0.01 , 0.01]	[-0.01 , 0.01]
ACC	[0.00 , 0.02]	[0.00 , 0.01]	[0.02 , 0.08]
RMSE	[-0.14 , -0.02]	[-0.30 , -0.16]	[-0.22 , -0.11]
ECE	[-0.40 , -0.18]	[-0.76 , -0.56]	[-0.55 , -0.27]
MCE	[-0.35 , -0.12]	[-0.58 , -0.33]	[-0.62 , -0.39]