Published in final edited form as:
Proc IEEE Int Conf Data Min. 2016 December ; 2016: 360–369. doi:10.1109/ICDM.2016.0047.

Binary Classifier Calibration using an Ensemble of Near Isotonic Regression Models

Mahdi Pakdaman Naeini and
Intelligent Systems Program, University of Pittsburgh, Pittsburgh, USA

Gregory F. Cooper
Department of Biomedical Informatics, University of Pittsburgh, Pittsburgh, USA

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 near isotonic regression* (ENIR). The method can be considered as an extension of BBQ [20], a recently proposed calibration method, as well as the commonly used calibration method based on isotonic regression (IsoRegC) [27]. ENIR is designed to address the key limitation of IsoRegC which is the monotonicity assumption of the predictions. Similar to BBQ, the method post-processes the output of a binary classifier to obtain calibrated probabilities. Thus it can be used with many existing classification models to generate accurate probabilistic predictions.

We demonstrate the performance of ENIR on synthetic and real datasets for commonly applied binary classification models. Experimental results show that the method outperforms several common binary classifier calibration methods. In particular on the real data, ENIR commonly performs statistically significantly better than the other methods, and never worse. 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 $O(N \log N)$ time, where N is the number of samples.

I. Introduction

In many real world data mining applications, intelligent agents often must make decisions under considerable uncertainty due to noisy observations, physical randomness, incomplete data, and incomplete knowledge. Decision theory provides a normative basis for intelligent agents to make rational decisions under such uncertainty. To do so, decision theory combines utilities and probabilities to determine the optimal actions that maximize expected utility [23]. The output in many of the machine learning models that are used in data mining applications is designed to discriminate the patterns in data. However, such output should also provide accurate (calibrated) probabilities in order to be practically useful for rational decision making in many real world applications.

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. Informally, 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. Figure 1 illustrates the binary calibration problem using a reliability curve [6], [19]. The curve shows the probability predicted by the classification model versus the actual fraction of positive outcomes for a hypothetical binary classification problem, where Z is the binary event being predicted. The curve shows that when the model predicts $Z = 1$ to have probability 0.2, the outcome $Z = 1$ occurs in about 0.3 fraction of the time. The curve shows that the model is fairly well calibrated, but it tends to underestimate the actual probabilities. In general, the straight dashed line connecting (0, 0) to (1, 1) represents a perfectly calibrated model. The closer a calibration curve is to this line, the better calibrated is the associated prediction model. Deviations from perfect calibration are very common in practice and may vary widely depending on the binary classification model that is used [20].

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 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 [28], [16], [11]. Research on learning well calibrated models has not been explored in the data mining literature as extensively as, 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 will 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 could potentially increase the complexity and computational cost of the associated optimization program to learn the model. The other approach is to rely on the 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 [20]. However, this approach has the potential to decrease discrimination while 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.

Existing post-processing binary classifier calibration methods include Platt scaling [22], histogram binning [26], isotonic regression [27], and a recently proposed method BBQ which is a Bayesian extension of histogram binning [20]. In all these methods, the post-processing step can be seen as a function that maps the outputs of a prediction model to probabilities that are intended to be well-calibrated. Figure 1 shows an example of such a mapping.

In general, there are two main applications of post-processing calibration methods. First, they can be used to convert the outputs of discriminative classification methods with no apparent probabilistic interpretation to posterior class probabilities [22]. 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 [19]. 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.

II. 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 [22]. The two parameters of the sigmoid function are learned in a maximum-likelihood framework using a model-trust minimization algorithm [10]. 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 [19]. 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 [17].

A popular non-parametric calibration method is the equal frequency histogram binning model which is also known as quantile binning [26]. 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. Histogram 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, and (3) there is uncertainty in the optimal number of the bins to use [27].

The most commonly used non-parametric classifier calibration method in machine learning and data mining applications is the *isotonic regression based calibration* (IsoRegC) model [27]. To build a mapping from the uncalibrated output of a classifier to the calibrated probability, IsoRegC 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 [2]. An IsoRegC model based on PAVA can be viewed as a histogram binning model [27] 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 [18]. In addition, an extension to

IsoRegC combines the outputs generated by multiple binary classifiers to obtain calibrated probabilities [29]. While IsoRegC can perform well on some real datasets, the monotonicity assumption it makes can fail in real data mining applications. This can specifically occur when we encounter large scale data mining problems in which we have to make simplifying assumptions to build the classification models. Thus, there is a need to relax the assumption, which is the focus of the current paper.

Adaptive calibration of predictions (ACP) is another extension to histogram binning [17]. ACP requires the derivation of a 95% statistical confidence interval around each individual prediction to build the bins. It then sets the calibrated estimate to the observed frequency of the instances with positive class among all the predictions that fall within the bin. To date, ACP has been developed and evaluated using only logistic regression as the base classifier [17].

Recently, a new non-parametric calibration model called BBQ was proposed which is a refinement of the histogram-binning calibration method [20]. BBQ addresses the main drawbacks of the histogram binning model by considering multiple different equal frequency histogram binning models and their combination using a Bayesian scoring function [12]. However, BBQ has two disadvantages. First, as a postprocessing calibration method, it does not take advantage of the fact that in real world applications a classifier with poor discrimination performance (e.g., low area under the ROC curve) will seldom be used. Thus, BBQ will usually be applied to calibrate classifiers with at least fair discrimination performance. Second, BBQ still selects the position and boundary of the bins by considering only equal frequency histogram binning models. A Bayesian non-parametric method called ABB addresses the latter problem by considering Bayesian averaging over all possible binning models induced by the training instances [21]. 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[1].

This paper presents a new binary classifier calibration method called *ensemble of near isotonic regression* (ENIR) that can post process the output generated by a wide variety of classification models. The essential idea in ENIR is to use the prior knowledge that the scores to be calibrated are in fact generated by a well-performing classifier in terms of discrimination. IsoRegC also uses such prior knowledge; however, it is biased by constraining the calibrated scores to obey the ranking imposed by the classifier. In the limit, this is equivalent to presuming the classifier has AUC equal to 1, which rarely happens in real world biomedical applications. In contrast, BBQ does not make any assumptions about the correctness of classifier rankings. ENIR provides a balanced approach that spans between IsoRegC and BBQ. In particular, ENIR assumes that the mapping from uncalibrated scores to calibrated probabilities is a near isotonic (monotonic) mapping; it allows violations of the ordering imposed by the classifier and then penalizes them through the use of a regularization term. ENIR utilizes the path algorithm *modified pool adjacent*

[1] 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 calibration precision in order to decrease calibration time [21]

violators algorithm (mPAVA) that can find the solution path to a near isotonic regression problem in $O(N \log N)$, where N is the number of training instances [25]. Finally, it uses the BIC scoring measure to combine the predictions made by these models to yield more robust calibrated predictions.

We perform an extensive set of experiments on a large suite of real datasets, to show that ENIR outperforms both IsoRegC and BBQ. Our experiments show that the near-isotonic assumption made by ENIR is a realistic assumption about the output of classifiers, and unlike the isotonicity assumption that is made by IsoReg, it is not biased. Moreover, our experiments show that by post processing the output of classifiers using ENIR, we can gain high calibration improvement, without losing any statistically meaningful discrimination performance.

The remainder of this paper is organized as follows. Section III introduces the ENIR method. Section IV describes a set of experiments that we performed to evaluate ENIR and other calibration methods. Finally, Section V states conclusions and describes several areas for future work.

III. Method

In this section we introduce the *ensemble of near isotonic regression* (ENIR) calibration method. ENIR utilizes the near isotonic regression method that seeks a nearly monotone approximation for a sequence of data $y_1,\dots,y_n$ [25]. The proposed calibration method extends the commonly used isotonic regression-based calibration by an approximate selective Bayesian averaging of a set of nearly isotonic regression models. The set includes the isotonic regression model as an extreme member. From another viewpoint, ENIR can be considered as an extension to a recently introduced calibration model BBQ [20] by relaxing the assumption that probability estimates are independent inside the bins and finding the boundary of the bins automatically through an optimization algorithm.

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 ith instance. In this paper, we focus on calibrating a binary classifier's output[2], and thus, $z_i \in \{0, 1\}$ and $y_i \in [0, 1]$. Let $\mathcal{D}$ define the set of all training instances (y_i, z_i). Without loss of generality, we can assume that the instances are sorted based on the classifier scores y_i, so we have $y_1 \le y_2 \le \dots \le y_N$, where N is the total number of samples in the training data.

The standard isotonic regression-based calibration model finds the calibrated probability estimates by solving the following optimization problem:

[2] 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.

$$\begin{array}{lc} \hat{\boldsymbol{p}}_{\text{iso}} = \underset{p \in R^N}{\operatorname{argmin}} & \frac{1}{2}\sum_{i=1}^{N}(p_i - z_i)^2 \\ \text{s. t.} & p_1 \leq \ldots \leq p_N \\ & 0 \leq p_i \leq 1 \forall i \in \{1, \ldots, N\}, \end{array} \quad (1)$$

where $\hat{\boldsymbol{p}}_{iso}$ is the vector of calibrated probability estimates. The rationale behind this model is to assume that the base classifier ranks the instances correctly. To find the calibrated probability estimates, it seeks the best fit of the data that is consistent with the classifier's ranking. A unique solution to the above convex optimization program exists and can be obtained by an iterative algorithm called *pool adjacent violators algorithm* (PAVA) that runs in $O(N)$. Note, however, that isotonic regression calibration still needs $O(N \log N)$ computations due to the fact that instances are required to be sorted based on the classifier scores y_i. PAVA iteratively groups the consecutive instances that violate the ranking constraint and uses their average over z (frequency of positive instances) as the calibrated estimate for all the instances within the group. We define the set of these consecutive instances that are located in the same group and attain the same predicted calibrated estimate as a bin. Therefore, an isotonic regression-based calibration can be viewed as a histogram binning method [27] where the position of boundaries are selected by fitting the best monotone approximation to the training data according to the ranking imposed by the classifier.

One can show that the second constraint in the optimization given by Equation 1 is redundant, and it is possible to rewrite the equation in the following equivalent form:

$$\begin{array}{lc} \hat{\boldsymbol{p}}_{\text{iso}} = \underset{\boldsymbol{p} \in R^N}{\operatorname{argmin}} & \frac{1}{2}\sum_{i=1}^{N}(p_i - z_i)^2 + \lambda \sum_{i=1}^{N-1}(p_i - p_{i+1})\nu_i \\ \text{s. t.} & \lambda = +\infty, \end{array} \quad (2)$$

where $\nu_i = \mathbb{1}(p_i > p_{i+1})$ is the indicator function of ranking violation. Relaxing the equality constraint in the above optimization program leads to a new convex optimization program as follows:

$$\hat{\boldsymbol{p}}_{\lambda} = \underset{\boldsymbol{p} \in R^N}{\operatorname{argmin}} \frac{1}{2}\sum_{i=1}^{N}(p_i - z_i)^2 + \lambda \sum_{i=1}^{N-1}(p_i - p_{i+1})\nu_i, \quad (3)$$

where λ is a positive real number that regulates the tradeoff between the monotonicity of the calibrated estimates with the goodness of fit by penalizing adjacent pairs that violate the ordering imposed by the base classifier. The above optimization problem is equivalent to the near-isotonic regression problem [25]. It yields a unique solution $\hat{p}_\lambda$, where the use of the subscript λ emphasizes the dependency of the final solution to the value of λ.

The entire path of solutions for any value of λ of the near isotonic regression problem can be found using a similar algorithm to PAVA which is called *modified pool adjacent violators algorithm* (mPAVA) [25]. mPAVA finds the whole solution path in $O(N \log N)$, and needs $O(N)$ memory space. Briefly, the algorithm works as follows: It starts by constructing N bins, each bin containing a single instance of the train data. Next, it finds the solution path by starting from the saturated fit $p_i = z_i$, that corresponds to setting $\lambda = 0$, and then increasing λ iteratively. As the λ increases the calibrated probability estimates $\hat{p}_{\lambda,i}$, for each bin, will change linearly with respect to λ until the calibrated probability estimates of two consecutive bins attain equal value. At this stage, mPAVA merges the two bins that have the same calibrated estimate to build a larger bin, and it updates their corresponding estimate to a common value. The process continues until there is no change in the solution for a large enough value of λ that corresponds to finding the standard isotonic regression solution. The essential idea of mPAVA is based on a theorem stating that if two adjacent bins are merged on some value of λ to construct a larger bin, then the new bin will never split for all larger values of λ [25].

mPAVA yields a collection of nearly isotonic calibration models, with the over fitted calibration model at one end ($\hat{p}_{\lambda=0} = z$) and the isotonic regression solution at the other ($\hat{p}_{\lambda=\lambda_\infty} = \hat{\boldsymbol{p}}_{iso}$), where λ_∞ is a large positive real number. Each of these models can be considered as a histogram binning model where the position of boundaries and the size of bins are selected according to how well the model trades off the goodness of fit with the preservation of the ranking generated by the classifier, which is governed by the value of λ, (As λ increases the model is more concerned to preserving the original ranking of the classifier, while for the small λ it prioritizes the goodness of fit.)

ENIR employs the approach just described to generate a collection of models (one for each value of λ). It then uses the Bayesian Information Criterion (BIC) to score each of the models [3]. Assume mPAVA yields the binning models $M_1, M_2, \ldots, M_T$, where T is the total number of models generated by mPAVA. For any new classifier output y, the calibrated prediction in the *ENIR* model is defined using selective Bayesian model averaging [13]:

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

where $P(z = 1|y, M_i)$ is the probability estimate obtained using the binning model M_i for the uncalibrated classifier output y. Also, $Score(M_i)$ is defined using the BIC scoring function [4] [24].

Next, for the sake of completeness, we briefly describe the mPAVA algorithm; more detailed information about the algorithm and the derivations can be found in [25].

[3] Note that we exclude the highly overfitted model that corresponds to $\lambda = 0$ from the set of models in ENIR

[4] Note that, as it is recommended in [25], we use the expected degree of freedom of the nearly isotonic regression models, which is equivalent to the number of bins, as the number of parameters in the BIC scoring function.

A. The modified PAV algorithm

Suppose at a value of λ we have N_λ bins, B_1, B_2,…, B_{N_λ}. We can represent the unconstrained optimization program given by Equation 3 as the following loss function that we seek to minimize :

$$\mathscr{L}_{B,\lambda}(\boldsymbol{z},\boldsymbol{p})=\frac{1}{2}\sum_{i=1}^{N_\lambda}\sum_{j\in B_i}(p_{B_i}-z_j)^2+\lambda\sum_{i=1}^{N_{\lambda-1}}(p_{B_i}-p_{B_{i+1}})\nu_i, \qquad (4)$$

where p_{B_i} defines the common estimated value for all the instances located at the bin B_i. The loss function $\mathscr{L}_{B,\lambda}$ is always differentiable with respect to p_{B_i} unless two calibrated probabilities are just being joined (which only happens if $p_{B_i} = p_{B_{i+1}}$ for some i). Assuming that $\hat{p}_{B_i}(\lambda)$ is optimal, the partial derivative of $\mathscr{L}_{B,\lambda}$ has to be 0 at $\hat{p}_{B_i}(\lambda)$, which implies:

$$|B_i|\hat{p}_{B_i}(\lambda)-\sum_{j\in B_i}z_j+\lambda(\nu_i-\nu_{i-1})=0 \text{ for } i=1,\dots,N_\lambda \qquad (5)$$

Rewriting the above equation, the optimum predicted value for each bin can be calculated as:

$$\hat{p}_{B_i}(\lambda)=\frac{\sum_{j\in B_i}z_j-\lambda\nu_i+\lambda_{\nu_{i-1}}}{|B_i|}\text{for } i=1,\dots,N_\lambda \qquad (6)$$

While PAVA uses the frequency of instances in each bin as the calibrated estimate, Equation 6 shows that mPAVA uses a shrunken version of the frequencies by considering the estimates that are not following the ranking imposed by the base classifier. In Equation 5, taking derivatives with respect to λ yields:

$$\frac{\partial\hat{p}_{B_i}}{\partial\lambda}=\frac{\nu_{i-1}-\nu_i}{|B_i|},\text{for } i=1,\dots,N_\lambda, \qquad (7)$$

where we set $\nu_0 = \nu_N = 0$ for notational convenience. As we noted above, it has been proven that the optimal values of the instances located in the same bin are tied together and the only way that they can change is to merge two bins as they can never split apart as λ increases [25]. Therefore, as we make changes in λ, the bins B_i, and hence the values ν_i remain constant. This implies the term $\frac{\partial\hat{p}_{B_i}}{\partial\lambda}$ is a constant in Equation 7. Consequently, the solution path remains piecewise linear as λ increases, and the breakpoints happen when two bins merge together. Now, using the piecewise linearity of the solution path and assuming that the two bins B_i and B_{i+1} are the first two bins to merge by increasing λ, the value of $\lambda_{i,i+1}$ at which the two bins B_i and B_{i+1} will merge is calculated as:

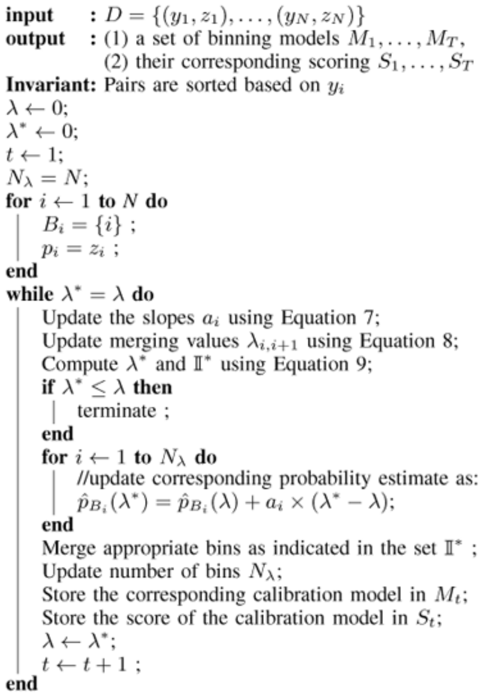
$$\lambda_{i,i+1}=\frac{\hat{p}_{B_i}(\lambda)-\hat{p}_{B_{i+1}}(\lambda)}{a_{i+1}-a_i}+\lambda\text{for } i=1,\ldots N_\lambda-1, \qquad (8)$$

where $a_i=\frac{\partial \hat{p}_{B_i}}{\partial \lambda}$ is the slope of the changes of $\hat{p}_{B_i}$ with respect to λ according to Equation 7. Using the above identity, the λ at which the next breakpoint occurs is obtained using the following equation:

$$\begin{gathered}\lambda^*=\min_i \lambda_{i,i+1}\\ {}^*=\{i|\lambda_{i,i+1}=\lambda^*\},\end{gathered} \qquad (9)$$

where $\mathbb{I}^*$ indicates the set of the indexes of the bins that will be merged by their consecutive bins changing the λ[5]. If $\lambda^* < \lambda$ then the algorithm will terminate since it has obtained the standard isotonic regression solution, and by increasing λ none of the existing bins will ever merge. Having the solutions of the near isotonic regression problem in Equation 3 at the breakpoints, and using the piecewise linearity property of the solution path, it is possible to recover the solution for any value of λ through interpolation. However, the current implementation of ENIR only uses the near isotonic regression based calibration models that corresponds to the value of λ at the breakpoints. The sketch of the algorithm is shown as Algorithm [1].

[5] Note that there could be more than one bin achieving the minimum in Equation 9, so they should be all merged with the bins that are located next to them.



```
input      : $D = \{(y_1, z_1), \ldots, (y_N, z_N)\}$
output     : (1) a set of binning models $M_1, \ldots, M_T$,
             (2) their corresponding scoring $S_1, \ldots, S_T$
Invariant: Pairs are sorted based on $y_i$
$\lambda \leftarrow 0$;
$\lambda^* \leftarrow 0$;
$t \leftarrow 1$;
$N_\lambda = N$;
for $i \leftarrow 1$ to $N$ do
    $B_i = \{i\}$ ;
    $p_i = z_i$ ;
end
while $\lambda^* = \lambda$ do
    Update the slopes $a_i$ using Equation 7;
    Update merging values $\lambda_{i,i+1}$ using Equation 8;
    Compute $\lambda^*$ and $\mathbb{I}^*$ using Equation 9;
    if $\lambda^* \leq \lambda$ then
        terminate ;
    end
    for $i \leftarrow 1$ to $N_\lambda$ do
        //update corresponding probability estimate as:
        $\hat{p}_{B_i}(\lambda^*) = \hat{p}_{B_i}(\lambda) + a_i \times (\lambda^* - \lambda)$;
    end
    Merge appropriate bins as indicated in the set $\mathbb{I}^*$ ;
    Update number of bins $N_\lambda$;
    Store the corresponding calibration model in $M_t$;
    Store the score of the calibration model in $S_t$;
    $\lambda \leftarrow \lambda^*$;
    $t \leftarrow t + 1$ ;
end
```

Algorithm 1: The *modified pool adjacent violators algorithm* (mPAVA) that yields a set of near-isotonic-regression-based calibration models $M_1, \ldots, M_T$

IV. Empirical Results

This section describes the set of experiments that we performed to evaluate the performance of ENIR in comparison to Isotonic Regression based Calibration (IsoRegC) [27], and a recently introduced binary classifier calibration method called BBQ [20]. We use IsoRegC because it is one of the most commonly used calibration models showing promising performance on real world applications [19], [27]. Moreover ENIR is an extension of IsoRegC, and we are interested in evaluating whether it performs better than IsoRegC. We also include BBQ as a state-of-the-art binary classifier calibration model, which is a Bayesian extension of the simple histogram binning model [20]. We did not include Platt's method since it is a simple and restricted parametric model and there are prior works showing that IsoRegC and BBQ perform superior to Platt's method [19], [27], [20]. We also did not include the ACP method since it requires not only probabilistic predictions, but also a statistical confidence interval (*CI*) around each of those predictions, which makes it tailored to specific classifiers, such as LR [17]; this is counter to our goal of developing post-processing methods that can be used with any existing classification models. Finally, we did not include ABB in our experiments mainly because it is not computationally tractable for real datasets that have more than a couple of thousand instances. Moreover, even for small size datasets, we have observed that ABB performs quite similarly to BBQ.

A. Evaluation Measures

In order to evaluate the performance of the calibration models, 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 three measures of calibration, namely, *root mean square error* (RMSE)[6], *maximum calibration error* (MCE), and *expected calibration error* (ECE) [20], [21]. MCE and ECE are two simple statistics of the reliability curve (Figure 1 shows a hypothetical example of such curve) 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 will be located in one of the bins. For each bin we can define the associated calibration error as the absolute difference between the expected value of 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 as follows:

$$\text{MCE}=\max_{k=1}^{K}(|o_k - e_k|)$$
$$\text{ECE}=\sum_{k=1}^{K} P(k)\cdot|o_k - e_k|,$$

where $P(k)$ is the empirical probability or the fraction of all instances that fall into bin k, e_k is the mean of the estimated probabilities for the instances in bin k, and o_k is the observed fraction of positive instances in bin k. The lower the values of *MCE* and *ECE*, the better is the calibration of a model.

B. Simulated Data

For the simulated data experiments, we used a binary classification dataset in which the outcomes were not linearly separable. The scatter plot of the simulated dataset is shown in Figure 2. We developed this classification problem to illustrate how IsoRegC can suffer from a violation of the isotonicity assumption, and to compare the performance of IsoRegC with our new calibration method that assumes approximate isotonicity. In our experiments, the data are divided into 1000 instances for training and calibrating the prediction model, and 1000 instances for testing the models. We report the average results of 10 random runs for the simulated dataset.

To conduct the experiments with the simulated data, we used two extreme classifiers: *support vector machines* (SVM) with linear and quadratic kernels. The choice of SVM with a linear kernel allows us to see how ENIR perform when the classification model makes an over simplifying (linear) assumption. Also, to achieve good discrimination on the circular configuration data in Figure 2, SVM with a quadratic kernel is a reasonable choice (as is also evidenced qualitatively in Figure 2 and quantitatively in Table Ib). So, the experiment using quadratic kernel SVM allows us to see how well ENIR performs when we use models that should discriminate well.

[6] 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 [6], [5]

As seen in Table I, ENIR generally outperforms IsoRegC on the simulation dataset, especially when the linear SVM method is used as the base learner. This is due to the monotonicity assumption of IsoRegC which presumes the best calibrated estimates will match the ordering imposed by the base classifier. When we use SVM with a linear kernel, this assumption is violated due to the non-linearity of the data. Consequently, IsoRegC only provides limited improvement of the calibration and discrimination performance of the base classifier. ENIR performs very well in this case since it is using the ranking information of the base classifier, but it is not anchored to it. The violation of the monotonicity assumption can happen in real data as well, especially in large scale data mining problems in which we use simple classification models due to the computational constraints. As shown in Table Ib, even when we apply a highly appropriate SVM classifier to classify the instances for which IsoRegC is expected to perform well (and indeed does so), ENIR performs as well or better than IsoRegC.

C. Real Data

We ran two sets of experiments on 40 randomly selected baseline datasets from the UCI and LibSVM repositories[7] [1], [4]. Five summary statistics of the size of the datasets and the percentage of the minority class are shown in Table III. We used three common classifiers, Logistic Regression (LR), Support Vector Machines (SVM), and Naïve Bayes (NB) to evaluate the performance of the proposed calibration method. In both sets of experiments on real data, we used 10 random runs of 10-fold cross validation, and we always used the train data for calibrating the models.

In the first set of experiments on real data, we were interested to evaluate if there is experimental support for using ENIR as a post-processing calibration method. Table II 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 ENIR with respect to the base classifier:

$$X=\frac{\text{measure}_{\text{ENIR}}-\text{measure}_{\text{method}}}{\text{measure}_{\text{method}}}, \qquad (10)$$

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 II shows that by post-processing the output of SVM using ENIR, we are 95% confident to gain anywhere from 17.6% to 31% average improvement in terms of RMSE. This could be a significant improvement, depending on the application, considering the 95% CI for the AUC which shows that by using ENIR we are 95% confident not to lose more than 1% of the SVM discrimination power in terms of AUC (Note, however, that the CI includes zero, which indicates that there is not a statistically significant difference between the performance of SVM and ENIR in terms of AUC).

[7] The datasets used were as follows: spect, adult, breast, pageblocks, pendigits, ad, mamography, satimage, australian, code rna, colon cancer, covtype, letter unbalanced, letter balanced, diabetes, duke, fourclass, german numer, gisette scale, heart, ijcnn1, ionosphere scale, liver disorders, mushrooms, sonar scale, splice, svmguide1, svmguide3, coil2000, balance, breast cancer, leu, w1a, thyroid sick, scene, uscrime, solar, car34, car4, protein homology.

Overall, the results in Table II show that there is not a statistically meaningful difference between the performance of ENIR and the base classifiers in terms of AUC. The results support at a 95% confidence level that ENIR improves the performance of LR and NB in terms of ACC. Furthermore, the results in Table II show that by post-processing the output of LR, SVM, and NB using ENIR, we can obtain dramatic improvements in terms of calibration measured by RMSE, ECE, and MCE. For instance, the results indicate that at a 95% confidence level, ENIR improved the average performance of NB in terms of MCE anywhere from 30.5% to 55.2%, which could be practically significant in many decision-making and data mining applications.

In the second set of experiments on real data, we were interested to evaluate the performance of ENIR compared with other calibration methods. To evaluate the performance of models, we used the recommended statistical test procedure by Janez Demsar [7]. More specifically, we used the non-parametric testing method based on the F_F test statistic [15], which is an improved version of Freidman non-parametric hypothesis testing method [9], followed by Holm's step-down procedure [14] to evaluate the performance of ENIR in comparison with other methods, across the 40 baseline datasets.

Tables [IV,V,VI] show the results of the performance of ENIR in comparison with IsoRegC and BBQ. In these tables, we show the average rank of each method across the baseline datasets, where boldface indicates the best performing method. In these tables, the marker */⊛ indicates whether ENIR is statistically superior/inferior to the compared method using the improved Friedman test followed by Holm's step-down procedure at a 0.05 significance level. For instance, Table V shows the performance of the calibration models when we use SVM as the base classifier; the results show that ENIR achieves the best performance in terms of RMSE by having an average rank of 1.675 across the 40 baseline datasets. The result indicates that in terms of RMSE, ENIR is statistically superior to BBQ; however, it is not performing statistically differently than IsoRegC.

Table IV shows the results of comparison when we use LR as the base classifier. As shown, the performance of ENIR is always superior to BBQ and IsoRegC except for MCE in which BBQ is superior to ENIR; however, this difference is not statistically significant. The results show that in terms of discrimination based on AUC, there is not a statistically significant difference between the performance of ENIR compared with BBQ and IsoRegC. However, ENIR performs statistically better than BBQ in terms of ACC. In terms of calibration measures, ENIR is statistically superior to both IsoRegC and BBQ in terms of RMSE. In terms of MCE, ENIR is statistically superior to IsoRegC.

Table V shows the results when we use SVM as the base classifier. As shown, the performance of ENIR is always superior to BBQ and IsoRegC except for MCE in which BBQ performs better than ENIR; however, the difference is not statistically significant for MCE. The results show that although ENIR is superior to IsoRegC and BBQ in terms of discrimination measures, AUC and ACC, the difference is not statistically significant. In terms of calibration measures, ENIR performs statistically superior to BBQ in terms of RMSE and it is statistically superior to IsoRegC in terms of MCE.

Table VI shows the results of comparison when we use NB as the base classifier. As shown, the performance of ENIR is always superior to BBQ and IsoRegC. In terms of discrimination, for AUC there is not a statistically significant difference between the performance of ENIR compared with BBQ and IsoRegC; however, in terms of ACC, ENIR is statistically superior to BBQ. In terms of calibration measures, ENIR is always statistically superior to IsoRegC. ENIR is also statistically superior to BBQ in terms of ECE and RMSE.

Overall, in terms of discrimination measured by AUC and ACC, the results show that the proposed calibration method either outperforms IsoRegC and BBQ, or has a performance that is not statistically significantly different. In terms of calibration measured by ECE, MCE, and RMSE, ENIR either outperforms other calibration methods, or it has a statistically equivalent performance to IsoRegC and BBQ.

Finally, Table VII shows a summary of the time complexity of different binary classifier calibration methods in learning for N training instances and the test time for only one instance.

V. Conclusion

In this paper, we presented a new non-parametric binary classifier calibration method called *ensemble of near isotonic regression* (ENIR) [8] to build accurate probabilistic prediction models. The method generalizes the isotonic regression-based calibration method (IsoRegC) [27] in two ways. First, ENIR makes a more realistic assumption compared to IsoRegC by assuming that the transformation from the uncalibrated output of a classifier to calibrated probability estimates is approximately (but not necessarily exactly) a monotonic function. Second, ENIR is an ensemble model that utilizes the BIC scoring function to perform selective model averaging over a set of near isotonic regression models that indeed includes IsoRegC as an extreme member. 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 ENIR, we can gain high calibration improvement in terms of RMSE, ECE, and MCE, without losing any statistically meaningful discrimination performance. Moreover, our experimental evaluation on a broad range of real datasets showed that ENIR outperforms IsoRegC and BBQ (i.e. a state-of-the-art binary classifier calibration method [20]).

An important advantage of ENIR over BBQ is that it can be extended to a multi-class and multi-label calibration models similar to what has done for the standard IsoRegC method [27]. This is an area of our current research. We also plan to investigate theoretical properties of ENIR. In particular, we are interested to investigate theoretical guarantees regarding the discrimination and calibration performance of ENIR, similar to what has been proved for the AUC guarantees of IsoRegC [8].

[8] An R package “enir” that implements our calibration method will be made available on the Comprehensive R Archive Network (CRAN) website by the time of conference.

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. It was also supported in part by NIH grants R01GM088224 and R01LM012095. The content is solely the responsibility of the authors and does not necessarily represent the official views of the NIH. This research was also supported by grant #4100070287 from the Pennsylvania Department of Health. The Department specifically disclaims responsibility for any analyses, interpretations, or conclusions

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. Chang, Chih-Chung, Lin, Chih-Jen. Libsvm: A library for support vector machines. ACM Transactions on Intelligent Systems and Technology (TIST). 2011; 2(3):27.
5. Cohen, Ira, Goldszmidt, Moises. Properties and benefits of calibrated classifiers. Knowledge Discovery in Databases: PKDD 2004. 2004:125–136.
6. DeGroot MH, Fienberg SE. The comparison and evaluation of forecasters. The Statistician. 1983:12–22.
7. Demšar, Janez. Statistical comparisons of classifiers over multiple data sets. The Journal of Machine Learning Research. 2006; 7:1–30.
8. Fawcett, Tom, Niculescu-Mizil, Alexandru. Pav and the roc convex hull. Machine Learning. 2007; 68(1):97–106.
9. 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.
10. Gill, Philip E., Murray, Walter, Wright, Margaret H. Practical optimization. Vol. 5. Academic press; London: 1981.
11. Hashemi, Homa B., Yazdani, Nasser, Shakery, Azadeh, Naeini, Mahdi Pakdaman. 5th International Symposium on Telecommunications (IST). IEEE; 2010. Application of ensemble models in web ranking; p. 726-731.
12. Heckerman D, Geiger D, Chickering DM. Learning bayesian networks: The combination of knowledge and statistical data. Machine Learning. 1995; 20(3):197–243.
13. Hoeting, Jennifer A., Madigan, David, Raftery, Adrian E., Volinsky, Chris T. Bayesian model averaging: a tutorial. Statistical Science. 1999:382–401.
14. Holm, Sture. A simple sequentially rejective multiple test procedure. Scandinavian Journal of Statistics. 1979:65–70.
15. 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.
16. Jiang, Liangxiao, Zhang, Harry, Su, Jiang. Advanced Data Mining and Applications. Springer; 2005. Learning k-nearest neighbor naive bayes for ranking; p. 175-185.
17. 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]
18. Menon, Aditya, Jiang, Xiaoqian, Vembu, Shankar, Elkan, Charles, Ohno-Machado, Lucila. Predicting accurate probabilities with a ranking loss; Proceedings of the International Conference on Machine Learning; 2012. p. 703-710.
19. Niculescu-Mizil, A., Caruana, R. Predicting good probabilities with supervised learning; Proceedings of the International Conference on Machine Learning; 2005. p. 625-632.
20. Naeini, Mahdi Pakdaman, Cooper, Gregory, Hauskrecht, Milos. Obtaining well calibrated probabilities using bayesian binning; Twenty-Ninth AAAI Conference on Artificial Intelligence; 2015.

21. Naeini, Mahdi Pakdaman, Cooper, Gregory F., Hauskrecht, Milos. Binary classifier calibration using a bayesian non-parametric approach. SIAM Data Mining (SDM). 2015

22. 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.

23. 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.

24. Schwarz, Gideon, et al. Estimating the dimension of a model. The annals of statistics. 1978; 6(2): 461–464.

25. Tibshirani, Ryan J., Hoefling, Holger, Tibshirani, Robert. Nearly-isotonic regression. Technometrics. 2011; 53(1):54–61.

26. Zadrozny, B., Elkan, C. Obtaining calibrated probability estimates from decision trees and naive bayesian classifiers; International Conference on Machine Learning; 2001. p. 609-616.

27. Zadrozny, B., Elkan, C. Transforming classifier scores into accurate multiclass probability estimates; Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining; 2002. p. 694-699.

28. Zhang, Harry, Su, Jiang. Machine Learning: ECML 2004. Springer; 2004. Naive bayesian classifiers for ranking; p. 501-512.

29. Zhong, Leon Wenliang, Kwok, James T. Proceedings of the Twenty-Third international joint conference on Artificial Intelligence. AAAI Press; 2013. Accurate probability calibration for multiple classifiers; p. 1939-1945.

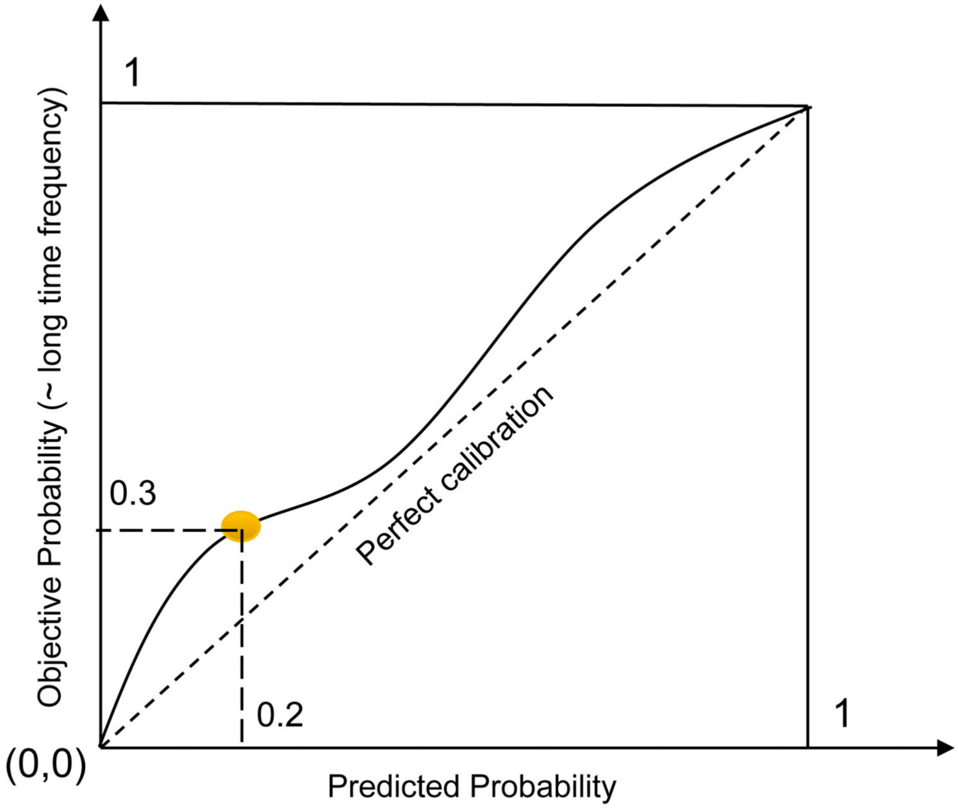


Fig 1.

The solid line shows a calibration (reliability) curve for predicting $Z = 1$. The dotted line is the ideal calibration curve.

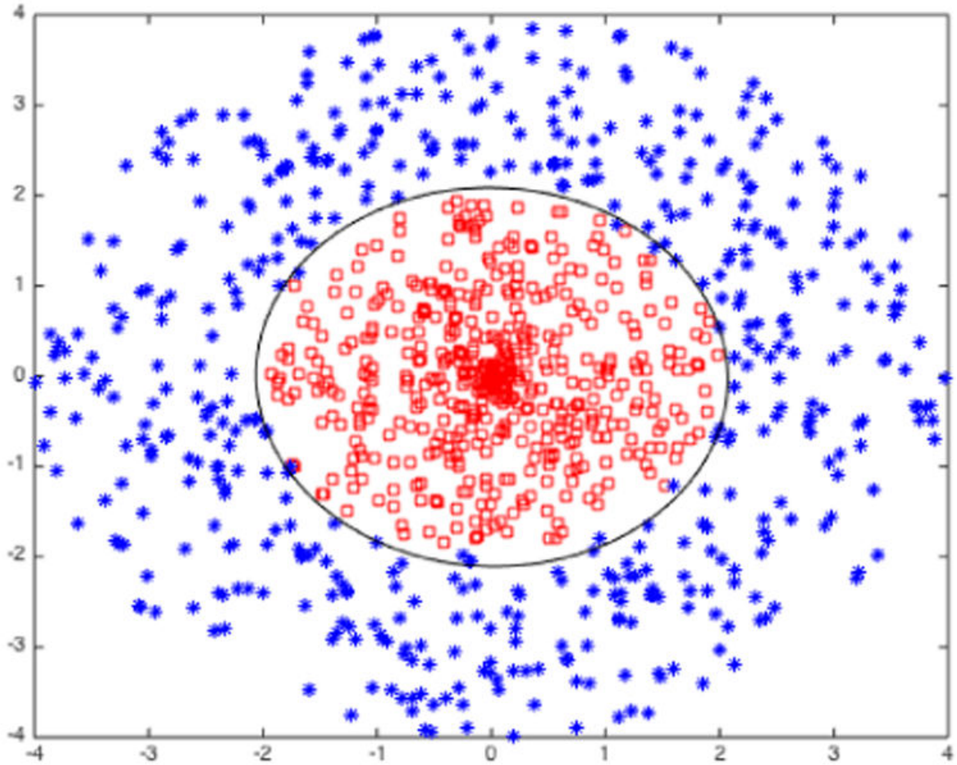


Fig 2.

Scatter plot of the simulated data. The black oval indicates the decision boundary found using SVM with a quadratic kernel.

Table I

Experimental Results on a simulated dataset

	SVM	**IsoRegC**	**BBQ**	**ENIR**
AUC	0.52	0.65	0.85	0.85
ACC	0.64	0.64	0.78	0.79
RMSE	0.52	0.46	0.39	0.38
ECE	0.28	0.35	0.05	0.05
MCE	0.78	0.60	0.13	0.12

(a) SVM Linear Kernel

	SVM	IsoRegC	BBQ	ENIR
AUC	1.00	1.00	1.00	1.00
ACC	0.99	0.99	0.99	0.99
RMSE	0.21	0.09	0.10	0.09
ECE	0.14	0.01	0.01	0.00
MCE	0.36	0.04	0.05	0.03

(b) SVM Quadratic Kernel

Table II

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

	LR	SVM	NB
AUC	[-0.008, 0.003]	[-0.010, 0.003]	[-0.010, 0.000]
ACC	[0.002, 0.016]	[-0.001, 0.010]	[0.012, 0.068]
RMSE	[-0.124, -0.016]	[-0.310, -0.176]	[-0.196, -0.100]
ECE	[-0.389, -0.153]	[-0.768, -0.591]	[-0.514, -0.274]
MCE	[-0.313, -0.064]	[-0.591, -0.340]	[-0.552, -0.305]

Table III

Summary statistics of the size of the real datasets and the percentage of the minority class. Q1 and Q3 defines the first quartile and thirds quartile respectively.

	Min	Q1	Median	Q3	Max
Size	42	683	1861	8973	581012
Percent	0.009	0.076	0.340	0.443	0.500

Table IV

Average rank of the calibration methods on the benchmark datasets using LR as the base classifier. Marker */⊛ indicates whether ENIR is statistically superior/inferior to the compared method (using an improved Friedman test followed by Holm's step-down procedure at a 0.05 significance level).

	IsoRegC	**BBQ**	**ENIR**
AUC	1.963	2.225	**1.813**
ACC	1.675	2.663*	**1.663**
RMSE	1.925*	2.625*	**1.450**
ECE	2.125	1.975	**1.900**
MCE	2.475*	**1.750**	1.775

Table V

Average rank of the calibration methods on the benchmark datasets using SVM as the base classifier. Marker */⊛ indicates whether ENIR is statistically superior/inferior to the compared method (using an improved Friedman test followed by Holm's step-down procedure at a 0.05 significance level).

	IsoRegC	**BBQ**	**ENIR**
AUC	1.988	2.025	**1.988**
ACC	2.000	2.150	**1.850**
RMSE	1.850	2.475*	**1.675**
ECE	2.075	2.025	**1.900**
MCE	2.550*	**1.625**	1.825

Table VI

Average rank of the calibration methods on the benchmark datasets using NB as the base classifier. Marker */⊛ indicates whether ENIR is statistically superior/inferior to the compared method (using an improved Friedman test followed by Holm's step-down procedure at a 0.05 significance level).

	IsoRegC	BBQ	ENIR
AUC	2.150	1.925	**1.925**
ACC	1.963	2.375*	**1.663**
RMSE	2.200*	2.375*	**1.425**
ECE	2.475*	2.075*	**1.450**
MCE	2.563*	1.850	**1.588**

Table VII

Note that N and B are the size of training sets and the number of bins found by the method, respectively. T is the number of iterations required for convergence of the Platt method and M is defined as the total number of models used in the associated ensemble model.

	Training Time	Testing Time
Platt	$O(NT)$	$O(1)$
Hist	$O(N \log N)$	$O(\log B)$
IsoRegC	$O(N \log N)$	$O(\log B)$
ACP	$O(N \log N)$	$O(N)$
ABB	$O(N^2)$	$O(N^2)$
BBQ	$O(N \log N)$	$O(M \log N)$
ENIR	$O(N \log N)$	$O(M \log B)$