Properties and benefits of calibrated classifiers

Ira Cohen and Moises Goldszmidt

Hewlett-Packard Research Laboratories
1501 Page Mill Rd., Palo Alto, CA 94304
{ira.cohen,moises.goldszmidt}@hp.com

Abstract. A calibrated classifier provides reliable estimates of the true probability that each test sample is a member of the class of interest. This is crucial in decision making tasks. Procedures for calibration have already been studied in weather forecasting, game theory, and more recently in machine learning, with the latter showing empirically that calibration of classifiers helps not only in decision making, but also improves classification accuracy. In this paper we extend the theoretical foundation of these empirical observations. We prove that (1) a well calibrated classifier provides bounds on the Bayes error (2) calibrating a classifier is guaranteed not to decrease classification accuracy, and (3) the procedure of calibration provides the threshold or thresholds on the decision rule that minimize the classification error. We also draw the parallels and differences between methods that use receiver operating characteristic (ROC) curves and calibration based procedures that are aimed at finding a threshold of minimum error. In particular, calibration leads to improved performance when multiple thresholds exist.

1 Introduction

In a decision making task, in order to evaluate different courses of action, it is useful to obtain accurate likelihood estimates of the alternatives. Pattern classifiers can be used to provide automated mappings between situations (represented by features) and outcomes (represented by the class membership). Yet, to be applicable to decision making problems, we require a reliable estimate of the true probability of class membership of each test sample. We will use the term *calibrated* to refer to classifiers with reliable estimates of the class membership probabilities. A successful classifier in terms of classification accuracy is not necessarily calibrated, e.g., the Naive Bayes classifier.Procedures for calibrating classifiers have been proposed in different contexts: In weather prediction tasks [1], in game theory [2, 3], and more recently in the context of pattern classification [4, 5]. Zadrozny and Elkan were also the first to notice the need of calibrating classifiers when used as decision making aids.

Our own incentive to study calibration came from applying probabilistic based classifiers to the problem of characterizing and forecasting the I/O response time of large storage arrays given passive observations. As these forecasts are used for scheduling purposes, we also need to accompany each forecast with

an accurate estimate of the probability of the forecast. We applied a variant of the calibration procedure suggested in [1, 4] and noticed that in addition to producing more accurate estimates, the classification accuracy of our induced classifiers increased. While these empirical results agree with those of Zadrozny and Elkan [4, 5], a theoretical guarantee that calibration cannot degrade classification performance was still missing. Our investigation of the calibration produced the following results which we prove in Sections 3 and 4. First, we can bound the Bayes error using the same parameters that result from the calculations needed for calibration. Second, we are *guaranteed* that the classification accuracy of the original classifier does not decrease as a consequence of calibration. Moreover, the classification accuracy can actually increase. Third, using the calibration process we can compute a threshold or thresholds in the decision rule of the classifier that minimize the classification error. We show that when a single threshold is derived from the calibration procedure, the result is equivalent to finding the point of minimum error in an ROC curve [6, 7]. However, when calibration produces multiple thresholds on the decision rule, the error achieved with those is lower than that of any single threshold derived from the ROC based methods. Thus, in addition to producing more accurate estimates of a-posteriori probabilities, calibration obviates the need for using ROC based methods for finding optimal thresholds.

The rest of the paper is organized as follows. Section 2 introduces formally the notions of calibration, refinement and Brier score. Sections 3 and 4 contain the proofs of our main results. Section 5 illustrates the effects of calibration on classifiers induced on real data, observing also the effect of the sample size on the process of calibration. Finally, Section 6 discusses and summarizes the main results.

2 Notation and preliminary definitions

A classifier takes an incoming vector of features $\mathbf{X}$ and maps it to a class label. We will use C to denote the *class variable* the values of which are called *classes*. Throughout this paper we assume a binary classification problem, i.e., one in which C takes one of two values. We will use $(1, 0)$, or $(c, \bar{c})$ to denote a specific instantiation of C. Each instantiation of $\mathbf{X}$, denoted by $\mathbf{x}$ is a *sample*. We assume that all samples are i.i.d.

Let $p(C|\mathbf{X})$ be the *true* a-posteriori distribution of the class given the features. The optimal classification rule, that is, the optimal function that maps a sample $\mathbf{x}$ to one of the values of C, under the 0-1 cost function, is the maximum a-posteriori (MAP) rule [8]:

$$g^*(\mathbf{x}) = argmax_{c'=(0,1)}[p(C = c'|\mathbf{x})], \tag{1}$$

The decision rule $g^*(\mathbf{x})$ is called the Bayes optimal decision and

$$\begin{aligned}\mathbf{e}_{\text{B}} &= \sum_{\mathbf{x}} p(g^*(\mathbf{x}) \neq c|C=c)\, p(\mathbf{x}) \\ &= \sum_{\mathbf{x}} min(p(C=1|\mathbf{x})\,, p(C=0|\mathbf{x}))p(\mathbf{x})\,, \end{aligned} \tag{2}$$

is the associated probability of error. This error is known as the Bayes error (or Bayes risk), and it is the minimum probability of error achievable with the given set of features. [1]

Given that $p(C|\mathbf{X})$ is unknown, one strategy for classification is to induce an estimate $\hat{p}(C|X)$ of the a-posteriori probability, and then use a decision rule $\hat{g}(X)$ such that the classification error, given by

$$CE = \sum_{\mathbf{x}} p(\hat{g}(\mathbf{x}) \neq c|C=c)\, p(\mathbf{x}) \tag{3}$$

is minimized. We note that plugging in $\hat{p}(C|X)$ into the decision rule in Eq. 1 may not be optimal [6], since given the errors and biases embedded in the estimate $\hat{p}(C|\mathbf{X})$ the threshold of 0.5, implicit in Eq. 1, may not minimize the error in Eq. 3. We return to this subject in Section 4, where we show the link between calibration and the decision rule that minimizes Eq. 3.

The classification error provides one way to evaluate classifiers. However, when using the classifier output as a basis for decision making, we need a score that takes into account not only the prediction accuracy of the classifier, but also the quality of the estimate $\hat{p}(C|\mathbf{X})$. One such score is the Brier score [9]. The Brier score is one of a class of so-called *proper* scores [1] which are used in evaluating the subjective probability assessment of forecasters. For the binary classification case, the Brier score is given as the average squared difference between the forecaster's probability of $C = 1$ and the true label:

$$BS = \frac{1}{n}\sum_{i=1}^{n}(\hat{p}(C=1|\mathbf{x}_i) - c_i)^2, \tag{4}$$

where n is the number of samples. Among the various intuitive justifications of this score, the following one is based on decision theoretic considerations. Assume that the agent (classifier or forecaster), should pay a price proportional to the confidence with which it asserts its decision. The Brier score uses the probability of the estimate as providing the appropriate penalty. Note that in Eq. 4, if the agent predicts $C = 1$ with high probability but $c_i = 0$ the penalty will be higher than if he predicts $C = 1$ with low probability. Thus, the lower the Brier score, the lower is the penalty assessed to the agent.

[1] Note that the summation over $\mathbf{X}$ implies finite values for the features; for continuous features the summation is replaced by integration. Throughout the paper we maintain the summation over $\mathbf{X}$, but note that the analysis holds for continuous features as well.

The notion of calibration can be derived directly from the Brier score. We need some preliminary definitions. Let $t \in [0,1]$ denote the a-posteriori probability assessment of a forecaster. Following [1], we assume that t takes on a finite number of values on the interval $[0,1]$. We denote by R_t the set of feature values for which the classifier density, $\hat{p}(C=1|\mathbf{x})$, yields a forecast probability t, namely:

$$R_t = \{\mathbf{x} \in \mathbf{X} : \hat{p}(C=1|\mathbf{x}) = t\}. \tag{5}$$

Let $\pi(t)$ be the probability that the forecaster predicts $C=1$ with probability t on a random instance. $\pi(t)$ can also be thought of as the frequency at which the forecaster predicted $C=1$ with probability t on a set of N samples, with $N \to \infty$. As such, given the probability density of the features, $p(\mathbf{x})$, $\pi(t)$ can be expressed as:

$$\pi(t) = \sum_{\mathbf{x} \in R_t} p(\mathbf{x})\,. \tag{6}$$

Let $p(C=1|t)$ be the probability that $C=1$ given that the forecaster predicts $C=1$ with probability t. The Brier score can be rewritten as (see [1] for derivation):

$$BS = \sum_t \pi(t)(t - p(c|t))^2 + \sum_t \pi(t)p(c|t)\,(1 - p(c|t)). \tag{7}$$

The first term is a measure of the *calibration*, and the second term is a measure of the *refinement* of the forecaster, denoted as $\mathbf{R}$. Calibration indicates how close is the probability assessment of the forecaster on $C=1$ to the frequency with which $C=1$ occurs (in reality). Note that for calibration to be 0, t has to be $p(c|t)$ for every t. A *well-calibrated* forecaster is one with calibration equal to 0. The notion of calibration fits our purposes, since the probability assessments of a well-calibrated agent, can be used in decision making as an indication of its confidence of the classification label provided.

Refinement scores the *usefulness* of each forecast. As an illustration, assume that we live in a place that rains 50% of the time. Thus a forecaster that always announces rain with 50% confidence is calibrated, yet not very useful in helping to plan a picnic for the following day. Ideally we would like estimates that are close to certainty. The more concentrated $p(c|t)$ is towards 0 or 1, the more refined the classifier. To minimize the overall Brier score, a forecaster has to be both well-calibrated and refined. Thus, if two classifiers are well-calibrated, the one with the lower Brier score is also more refined. We describe the relationship between bias, Bayes error, calibration and refinement in the next section.

3 The Brier score, bias, and the Bayes error

In the following we show that being well-calibrated is a weaker condition than being unbiased. Loosely speaking, a well-calibrated classifier is an "on average" unbiased classifier. We also show that we can use the notion of refinement (second term in Eq. 7) as a bound on the Bayes error. In particular, twice the refinement

of a well-calibrated classifier is an upper bound on the Bayes error; and, in the case that the classifier is unbiased, then its refinement is a lower bound on the Bayes error. Section 5 illustrates the practical implications of the various approximations made when calibrating in practice.

3.1 The bias/calibration relationship

Being well-calibrated requires that $t = p(c|t)$. Using Bayes rule we write $p(c|t)$ as $\frac{p(c,t)}{\pi(t)}$ which can be further rewritten as:

$$p(c|t) = \frac{\sum_{\mathbf{x}\in R_t} p(\mathbf{x})\, p(c|\mathbf{x})}{\sum_{\mathbf{x}\in R_t} p(\mathbf{x})}, \tag{8}$$

where $\pi(t)$ in the denominator is replaced using Eq. 6. The numerator states that the probability of the joint event that the class variable takes its c value and that the classifier states this with probability t, is the result of summing over these precise events in feature space (i.e, for $\mathbf{x} \in R_t$). Given our assumption regarding the i.i.d. nature of the samples, this holds. We can now state the following:

Proposition: An unbiased classifier is also well-calibrated.

Proof. For an unbiased classifier, $\lim_{n\to\infty} \hat{p}(c|\mathbf{x}) = p(c|\mathbf{x})$ for every $\mathbf{x}$, where n is the number of samples. Therefore, as $n \to \infty$, for every t: $\forall \mathbf{x} \in R_t$, $t = p(c|\mathbf{x})$. Replacing $p(c|\mathbf{x})$ with t in Eq.(8) yields $p(c|t) = t$, which is the condition for calibration to be 0.□

However, a well-calibrated classifier might not be unbiased. We see from Eq.(8) that for a well-calibrated classifier, its forecast, $\hat{p}(c|x) = t$ for $\mathbf{x} \in R_t$, is a normalized average of the true a-posteriori probability in the region defined by R_t. Clearly, one can construct cases where the classifier is biased, but $p(c|t) = t$ for all t: for example, suppose we have $X = \{1, 2\}$, $p(X = 1) = 0.5$ and $p(c|X = 1) = 0.2$ and $p(c|X = 2) = 0.6$. Suppose also that the classifier always predicts c with $\hat{p}(c|X) = 0.4$ for any X (hence on $t = 0.4$ has non-zero probability). Obviously, the classifier is biased. However, from Eq. 8 we have that $p(c|t) = 0.4$ and the classifier is well-calibrated.

3.2 The Bayes error-refinement relationship

We start by defining a t dependent error measure:

$$e_t = \sum_t \pi(t) min(t, 1 - t). \tag{9}$$

e_t essentially mirrors the Bayes error formula of Eq.2, but as we will see, e_t upper bounds the Bayes error. We are now ready to state the following result:

Theorem 1 *Given a well-calibrated classifier, whose forecasts are $\hat{p}(c|x)$, and given the true a-posteriori probability $p(c|x)$ with corresponding Bayes error rate $\mathbf{e}_{\text{B}}$, the following holds: $\mathbf{e}_{\text{B}} \leq e_t \leq 2\mathbf{R}$.*

Proof. Recall that for a well-calibrated classifier, $t = p(c|t)$. Making the appropriate substitution in the second term of Eq. 7, the refinement $\mathbf{R}$ can be written as:$\mathbf{R} = \sum_t \pi(t)t(1-t)$. It is easy to show that for $0 \leq t \leq 1$, $min(t, 1-t) \leq 2 \cdot t(1-t)$, from which follows that, $e_t \leq 2\mathbf{R}$.Now we have to show $\mathbf{e}_{\text{B}} \leq e_t$. We rewrite the expressions for the Bayes error in terms of t and R_t:

$$\mathbf{e}_{\text{B}} = \sum_t \sum_{\mathbf{x} \in R_t} p(\mathbf{x})\, min(p(c|\mathbf{x})\,, 1 - p(c|\mathbf{x})). \tag{10}$$

We use Eq. 6 to substitute the $\pi(t)$ term in Eq. 9 and obtain:

$$e_t = \sum_t \sum_{\mathbf{x} \in R_t} p(\mathbf{x})\, min(t, 1-t). \tag{11}$$

With this reformulation, all we have to show is that for every $\mathbf{x}$ in every R_t, $p(\mathbf{x})\, min(p(c|\mathbf{x})\,, 1-p(c|\mathbf{x})) \leq p(\mathbf{x})\, min(t, 1-t)$. We have two cases, when $t \leq 0.5$ and when $t > 0.5$. We proceed with the proof for the first case. The proof for the second case is completely analogous. For the case where $t \leq 0.5$ we can write:

$$\sum_{\mathbf{x} \in R_t} p(\mathbf{x})\, min(t, 1-t) = t \cdot \sum_{\mathbf{x} \in R_t} p(\mathbf{x})$$

Using Eq. 8 and the fact that the classifier is well calibrated we replace t in the right hand side of the above equation to get:

$$\begin{aligned} t \sum_{\mathbf{x} \in R_t} p(\mathbf{x}) &= \frac{\sum_{\mathbf{x} \in R_t} p(\mathbf{x})\, p(c|\mathbf{x})}{\sum_{\mathbf{x} \in R_t} p(\mathbf{x})} \cdot \sum_{\mathbf{x} \in R_t} p(x) \\ &= \sum_{\mathbf{x} \in R_t} p(\mathbf{x})\, p(c|\mathbf{x})\,. \end{aligned} \tag{12}$$

In going over all $\mathbf{x} \in R_t$, we have two cases, depending on whether $p(c|\mathbf{x}) < 0.5$ or $p(c|\mathbf{x}) \geq 0.5$.[2] Let $\mathbf{x}^-$ be such that $p(c|\mathbf{x}^-) < 0.5$. Thus we get that $min(p(c|\mathbf{x}^-)\,, 1 - p(c|\mathbf{x}^-)) = p(c|\mathbf{x}^-)$. It follows then that Eqs. 10 and Eq. 12 are equal for all such cases. Let now $\mathbf{x}' \in R_t$ be such that $p(c|\mathbf{x}') > 0.5$. For that $\mathbf{x}'$, $min(p(c|\mathbf{x}')\,, 1 - p(c|\mathbf{x}')) = 1 - p(c|\mathbf{x}')$. So, while e_t sums over $p(c|\mathbf{x}')$, as in Eq. 12, the Bayes error adds the smaller term, $1 - p(c|\mathbf{x}')$. It follows that $\mathbf{e}_{\text{B}} \leq e_t$. □

From the proof, we see that 'looseness' in the upper bound on the Bayes error occurs whenever for certain $\mathbf{x} \in R_t$, $p(c|\mathbf{x})$ is on the other side of $1/2$ with respect to t. For t's that are close to 0 or 1, there is less of a chance for such $\mathbf{x}$'s to exist (see Eq. 8), while t close to $1/2$ has higher chances of occurrence for such cases. Therefore, a well-calibrated classifier with $\pi(t)$ that has mass close to 0 and 1 is not only more refined, but also provides a tighter bound on the Bayes error.

If the classifier is unbiased, we can provide a stronger result. In this case we know that asymptotically, $t = p(c|\mathbf{x})$ for every $\mathbf{x} \in R_t$ and we have that $R \leq \mathbf{e}_{\text{B}}$. This follows from the fact that $\mathbf{e}_{\text{B}} = e_t$ when the classifier is unbiased, and from the fact that for $0 \leq t \leq 1$, the relation $t(1-t) \leq min(t, 1-t)$ holds.

[2] Recall that these $\mathbf{x}$ samples are placed in R_t according to the value of $\hat{p}(c|\mathbf{x})$.

4 Calibration, classification error and ROC curves

As discussed in Section 2, in order to minimize the classification error given by Eq. 3, we need to find the appropriate decision rule. This, in turn, translates to finding a probabilistic threshold α, so that we can classify a sample $\mathbf{x}$ as belonging to class c, when $\hat{p}(c|\mathbf{x}) \geq \alpha$. In this section we provide a procedure for finding α in terms of calibration. The intuition is as follows. If we had the real density $p(C|\mathbf{X})$, the optimal decision rule is given by Eq. 1, which in turn implies that $\alpha = 0.5$. Now, the process of calibrating may be seen as the process of bringing $\hat{p}(C|X)$ closer to the real density. Calibrating a classifier is a mapping from $\hat{p}(c|x)$ to $p(c|t)$. In fact the procedures proposed in [4, 5] essentially implement this mapping. Thus, under certain conditions we outline below the optimal threshold α^* of the original classifier is one where in the calibration mapping $p(c|\alpha^*) = 0.5$.

To formalize this intuitions we need to express the classification error in terms of the calibration mapping density. Suppose that our decision function on t is such that we say $C = 0$ if $t \leq \alpha$ and $C = 1$ if $t > \alpha$, where $0 \leq \alpha \leq 1$ (note that for the plug-in decision rule $\alpha = 0.5$). Given the density of $\pi(t)$ on t, the classification error is a function of α and is written as:

$$P_{error}(\alpha) = \int_0^{\alpha} p(C = 1|t)\, p(t)\, dt + \int_{\alpha}^{1} (1 - p(C = 1|t))p(t)\, dt, \qquad (13)$$

where now t takes any value on the interval $[0, 1]$, and is not limited to a discrete set as in the previous section. The first integral in Eq. 13 is the (weighted) area under the calibration map, $p(C = 1|t)$, for which we predict class 0; this area is proportional to the probability that we missed instances that had label of 1. The second integral provides the proportion of the error for which we predict 1, but the actual class label was 0. Borrowing terms from signal detection theory, the first term is proportional to the probability of missed detection (detection of class 1), and the second integral is proportional to the probability of false detection (or false alarm). These areas are illustrated as the marked regions in Figures 1(a) and (b). We can now state the following:

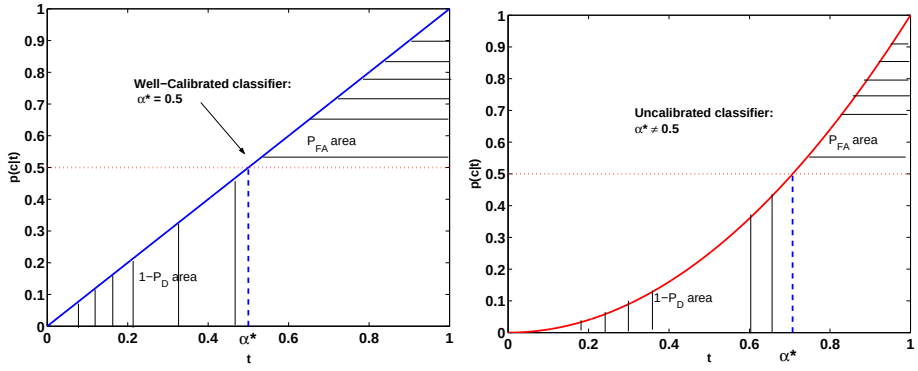
Theorem 2 *Given a classifier with a-posteriori probabilities t, density $\pi(t)$ and a calibration map $p(C = 1|t)$, where $p(C = 1|t)$ does not cross $1/2$ more than once, the threshold α on t which achieves minimum probability of error, i.e., $\alpha^* = \arg\min_\alpha P_{error}(\alpha)$ is given as α s.t. $p(C = 1|t = \alpha) = 0.5$.*

Proof. Taking the derivative of $P_{error}(\alpha)$ with respect to α yields:

$$\frac{dPerror}{d\alpha} = 2 \cdot p(C = 1|t = \alpha) - 1. \qquad (14)$$

Setting the derivative to 0 yields $p(C = 1|\alpha^*) = 1/2$.□

The reason why the calibration map provides the optimal threshold on t for minimizing the probability of error is quite simple: the calibration map can be thought of as a new well-calibrated classifier, with a single feature t - thus the threshold of $1/2$ on this new classifier is optimal. Because we require that the



(a) (b)

Fig. 1. (a) Illustration of the calibration map of a well-calibrated (diagonal line) and a non-calibrated classifier in (b).

calibration map does not cross $1/2$ more then once, there is a (single) threshold on our "feature" t that achieves the minimum error.

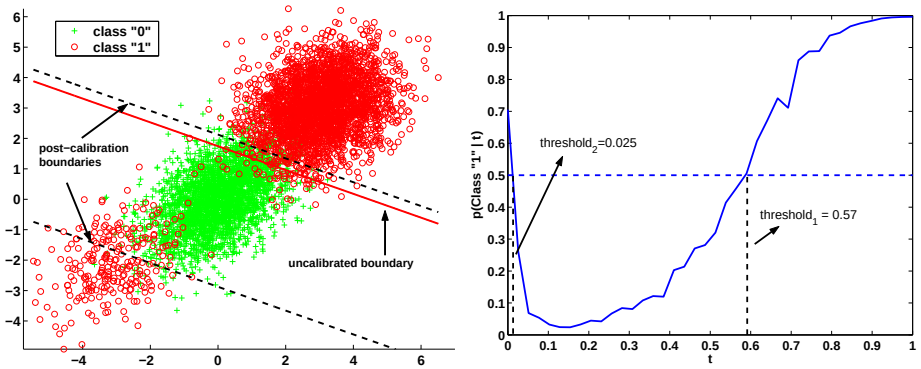
The function $p(C = 1|t)$ can also be used to create ROC curves. To see this, recall that an ROC curve plots the probability of detection, $P_D = P(Predict\ C = 1|Truth\ is\ C = 1)$, against the probability of false alarm, $P_{FA} = P(Predict\ C = 1|Truth\ is\ C = 0)$, created by varying a threshold (e.g., likelihood ratio). The threshold is varied so that we start from perfect detection, but maximum false alarm, to no false alarms, but minimum detection. We already stated that the two integrals composing P_{error} in Eq. 13 are directly related to P_D and P_{FA}, and to put it more accurately:

$$P_D(\alpha) = \frac{1 - \int_0^\alpha p(C = 1|t)\, p(dt)}{p(C = 1)}$$
$$P_{FA}(\alpha) = \frac{\int_\alpha^1 (1 - p(C = 1|t))p(dt)}{1 - p(C = 1)}, \tag{15}$$

thus by varying the threshold α, we can generate the entire ROC curve using the calibration map. At this point it is clear that methods that find the threshold of minimum error from ROC curves [6, 7] produce the exact same result as the calibration procedure, when the calibration map does not cross $1/2$ more than once.

However, the calibration procedure generalizes more than what can be achieved with the ROC method. Theorem 2 can be extended to the case where the calibration map crosses $1/2$ more than once, requiring multiple thresholds on the original decision function for minimizing the error: given multiple thresholds on the decision function we can rewrite equation 13 (splitting the integral based on the number of needed thresholds) and find that minimizing the probability of error for any number of thresholds still occurs when the calibration map is $1/2$. Such cases could occur with classifiers that output a-posteriori probabilities that

are ranked incorrectly. For example, suppose that one class is split into several clusters in space, and a classifier (for example, a linear one) separates well some clusters, leaving other clusters far from the decision boundary. The resultant calibration map of such classifiers would cross 0.5 at several places, but the point of minimum error is still at $p(C = 1|t) = 0.5$. Thus, inverting the calibration map when at that point provides several thresholds on the decision rule. We illustrate the above with a two dimensional example, shown in Figure 2(a). The class marked with circles (class "1") consists of two clusters which are divided by the class marked with x's (class "0"). Learning a Logistic regression classifier on the data leads to a single linear boundary (shown in the figure), which does well at separating one cluster, but leaves the second one very far from the boundary. Thus, data from that cluster have higher probability of belonging to class "0" than data from class "0" itself. Figure 2(b) shows the calibration map of the Logistic regression classifier. The map crosses 0.5 at two values, thus leading to two decision boundaries (with the same slope of the original, but two different intersects). With these boundaries, both clusters of class "1" are well separated, and the resultant error is significantly lower, reducing from 10% with the original boundary to 5.5% with the new boundaries.



(a) (b)

Fig. 2. Example of calibration finding multiple thresholds on the decision rule. (a) Decision boundaries before and after calibration superimposed on data. (b) The calibration map of the original linear classifier.

The results above can also be extended from the 0-1 loss to the general loss function, for which c_{01} is the cost of predicting class 0 when the true class is 1 and c_{10} is the cost of predicting class 1 and the true class is 0. The Bayes decision rule minimizing the *risk* under this loss function calls for classifying a sample $\mathbf{x}$ as 1 if $p(C = 1|\mathbf{x}) > \frac{c_{10}}{c_{10}+c_{01}}$ [10].As with the classification error under the 0-1 loss, applying the threshold $\frac{c_{10}}{c_{10}+c_{01}}$ on the *estimated* classifier, $\hat{p}(C = 1|\mathbf{x})$, may not minimize the risk under the generalized loss. However, using the same arguments given in Theorem 2, finding the thresholds which minimize the generalized loss function for a given classifier is the value on t for which $p(C = 1|t) = \frac{c_{10}}{c_{10}+c_{01}}$.

5 Calibration with finite data

With finite data sets, we want to estimate $p(C = 1|\hat{p}(C = 1|x))$ reliably. A procedure for this estimation was provided in [4, 5], where $\hat{p}(C = 1|x)$ is binned on the interval $[0, 1]$ and the calibration map is estimated by counting the number of samples that fall into each bin. The procedure was originally suggested as a method for calibrating Naive Bayes classifiers, but is applicable to any classifier that outputs probabilities, or a distance measure that can be converted to probabilities (e.g., Tree-augmented Naive Bayes [11], Logistic regression, mixture models, and SVMs). The empirical success of calibration on various (typically large sized data sets) has been shown in previous works – in this section we aim at providing insight to the finite sample effects that can arise with calibration.

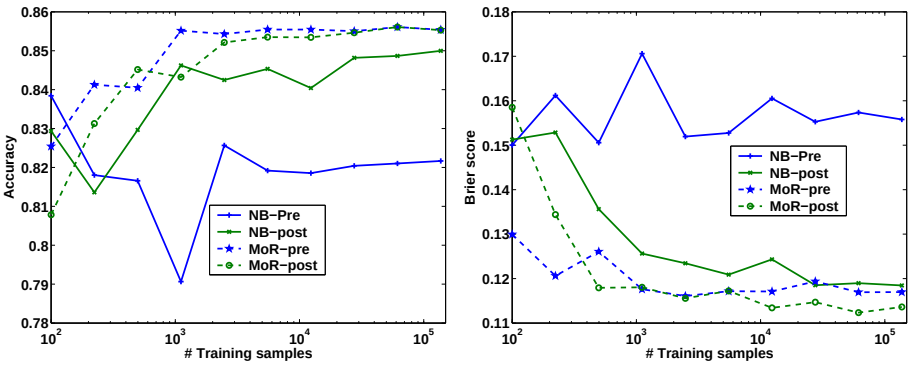
Estimating the calibration map involves learning a function from a scalar input ($\hat{p}(C = 1|\mathbf{x})$ to a scalar output. Thus, it is insensitive to the number of features in the classifier. The estimation is sensitive though to the sample size and to the number of bins used in the estimation procedure. We evaluate the effect of the sample size on the calibration procedure, thus we use learning curves, showing the various performance metrics before and after calibration.

We use the calibration procedure for prediction of I/O response time of individual requests to an enterprize storage array. Our data is based on an anonymized month-long trace of requests to an Hewlett Packard XP 512 storage array collected by the Storage Systems group at Hewlett-Packard Laboratories between 27 September and 27 October 2002. The raw traces are transformed to 10 features that describe queue lengths, locality and sequentiality, as measured by the server issuing the I/O request to the storage array. The problem is transformed to a binary classification problem by determining that any response time faster or equal to 1.5 msec is considered *fast*, while any response time slower than 1.5 msec is considered *slow*.

The data consists of 686091 training data and 343046 test data. We build two competing models to predict the correct class for the I/O request. The first is the Naive Bayes classifier, with Gaussian conditional distribution for the numerical feature and multinomial distribution for a locality feature. The second is a mixture of regression (MoR) classifier. The MoR model finds a mixture of regressors between the features and response time, which provides a distribution of response time for each value of the features, from which we can compute the a-posteriori probability of the response time being fast or slow. With the full training data, the Naive Bayes model achieves 82.18% accuracy before calibration and 85.60% after calibration, a significant improvement. The MoR model improves from 85.50% to 86.16%, a more modest improvement, to be expected from a model that is more naturally calibrated compared to Naive Bayes. The learning curves, both of accuracy and the Brier score, are shown in Figure 3.

For generating the learning curves we fix the number of bins used in the calibration procedure at 20, and average the results measured on the test set of 5 trials for each point on the curve. We see that for the Naive Bayes classifier, calibration improves accuracy and the Brier score early on the curve (already at 200 training samples), while for the already almost calibrated MoR, the calibration

procedure does not produce a significant benefit to performance until fairly large training sets are available. Observing the changes in the Brier score, it appears that both models achieve near convergence to a calibrated classifier as early as after 1000 samples. It is also important to note that for the MoR and sample sizes smaller than 400, the calibration procedure slightly degrades performance because of overfitting. These experiments illustrate that models that are far from being calibrated benefit from calibration even with few data; for classification, any change in the decision boundary in the right direction has a large effect. However, models that are close to being calibrated are more sensitive to noise in the calibration map, and are more prone to overfitting with small data sets. We discuss possible ways to overcome these effects in the summary.



(a) Classification accuracy (b) Brier score

Fig. 3. Learning curves of Naive Bayes and MoR for the I/O prediction data.

6 Summary

In this paper we characterize the mathematical relation between calibration and bounds on the Bayes error and the use of calibration to find thresholds in the decision rule minimizing a classifier's error. These theoretical results, coupled with mounting empirical evidence in the literature, illustrate the importance and value of calibrating classifiers for classification and decision making.

The result relating calibration with the decision rule that minimizes the classification error, produces an effective procedure for finding optimal thresholds in this decision rule. This also establishes a direct relationship with ROC curves, a relationship which was informally alluded to in [6], and is formalized in this paper. As with any learning algorithm, finite sample effects have to be considered; our learning curve experiments show that a simple calibration procedure performs well with large training sets, but can cause overfitting with small training sets. Reducing the possibility of overfitting can be done by smoothing of the calibration map or estimating a smooth function (such as the sigmoid) as the calibration map [5]. As a note, the number of thresholds on the decision function would depend on the smoothing function used, e.g., with a sigmoid, only one

threshold can be found, which might not be always desirable. We also observe that calibration is more beneficial, even at small sample sizes, for classifiers that are inherently not calibrated (such as Naive Bayes), compared to calibration of classifiers that are more naturally calibrated (such as logistic regression).

Future work includes providing bounds on how the estimation error of the calibration map affects the estimation of the optimal thresholds for classification and the payoff in terms of decision making.Such bounds could help avoid overfitting, especially with small sample sizes. Extending the method beyond binary classification problems is another research direction; similar to methods extending ROC curves beyond binary classification [7]. We are also exploring the use of calibration in semi-supervised learning, helping eliminate the possibility of performance degradation when using unlabeled data to learning classifiers, a phenomenon that occurs with biased models that output uncalibrated a-posteriori probabilities [12].

Acknowledgments

We thank Terence Kelly both for his help and suggestions and his work on the I/O response time prediction. We also thank Kim Keeton for providing the I/O data, Tom Fawcett for his comments on ROC curves, George Forman and Charles Elkan for providing feedback on the paper.

References

1. DeGroot, M., Fienberg, S.: The comparison and evaluation of forecasters. The statistician **32** (1983) 12–22
2. Fundenberg, D., Levine, D.: An easier way to calibrate. Games and economic behavior **29** (1999) 131–137
3. Foster, D., Vohra, R.V.: Asymptotic calibration. Biometrika **85** (1998) 379–390
4. Zadrozny, B., Elkan, C.: Obtaining calibrated probability estimates from decision trees and naive Bayesian classifiers. In: ICML. (2001)
5. Zadrozny, B., Elkan, C.: Transforming classifier scores into accurate multiclass probability estimates. In: Knowledge Discovery and Data Mining. (2002)
6. Fawcett, T.: ROC graphs: Notes and practical considerations for data mining representation. Technical Report HPL-2003-4, Hewlett-Packard Labs, Palo Alto, CA (2003)
7. Lachiche, N., Flach, P.: Improving accuracy and cost of two-class and multi-class probabilistic classifiers using ROC curves. In: ICML. (2003) 416–423
8. Devroye, L., Gyorfi, L., Lugosi, G.: A Probabilistic Theory of Pattern Recognition. Springer Verlag, New York (1996)
9. Brier, G.: Verification of forecasts expressed in terms of probability. Monthly weather review **78** (1950) 1–3
10. Duda, R.O., Hart, P.E., Stork, D.: Pattern Classification. John Wiley and Sons, New York (2001)
11. Friedman, N., Geiger, D., Goldszmidt, M.: Bayesian network classifiers. Machine Learning **29** (1997) 131–163
12. Cozman, F.G., Cohen, I., Cirelo, M.: Semi-supervised learning of mixture models. In: ICML. (2003) 99–106