Data Mining and Knowledge Discovery 1, 55–77 (1997)

On Bias, Variance, 0/1—Loss, and the Curse-of-Dimensionality

JEROME H. FRIEDMAN
Department of Statistics and Stanford Linear Accelerator Center, Stanford University

Editor: Usama Fayyad

Received June 3, 1996; Revised October 23, 1996; Accepted October 23, 1996

Abstract. The classification problem is considered in which an output variable y assumes discrete values with respective probabilities that depend upon the simultaneous values of a set of input variables $\mathbf{x} = \{x_1, \ldots, x_n\}$. At issue is how error in the estimates of these probabilities affects classification error when the estimates are used in a classification rule. These effects are seen to be somewhat counter intuitive in both their strength and nature. In particular the bias and variance components of the estimation error combine to influence classification in a very different way than with squared error on the probabilities themselves. Certain types of (very high) bias can be canceled by low variance to produce accurate classification. This can dramatically mitigate the effect of the bias associated with some simple estimators like "naive" Bayes, and the bias induced by the curse-of-dimensionality on nearest-neighbor procedures. This helps explain why such simple methods are often competitive with and sometimes superior to more sophisticated ones for classification, and why "bagging/aggregating" classifiers can often improve accuracy. These results also suggest simple modifications to these procedures that can (sometimes dramatically) further improve their classification performance.

Keywords: classification, bias, variance, curse-of-dimensionality, bagging, naive Bayes, nearest-neighbors

1. Introduction

One of the most common and important uses for data is prediction. The purpose is to predict (forecast) the unknown value of some attribute y of a system ("output" or "response" variable), based on the known values of other attributes $\mathbf{x} = (x_1, \ldots, x_n)$ ("input" or "predictor" variables). With the "supervised learning" paradigm the data base provides a "training" sample

$$T = \{\mathbf{x}_i, y_i\}_1^N \tag{1.1}$$

of previously solved cases in which both the values of the inputs and corresponding outputs have been recorded. The goal of a "learning" algorithm is to use this data (1.1) to construct a reliable rule for predicting likely output values y for future data where only the values of the inputs $\mathbf{x}$ have been recorded.

The rule construction strategy can depend upon the nature of the output variable in terms of the types of values it can realize. The two most common data types are orderable $y \in R^1$ ("regression") and categorical $y \in \{y_1, \ldots, y_L\}$ ("classification"). With an orderable variable there is an order relation between every pair of its values $y, y'(y \leq y' \text{ or } y > y')$ and a distance $|y - y'|$ defined between them. For a categorical variable there exists no

order relation nor a continuous distance between its values. Two values are either equal ($y = y'$) or they are unequal ($y \neq y'$).

Historically, both regression and classification have been developed from the common perspective of real valued prediction. In the case of classification the real valued (surrogate) outputs are taken to be the respective probabilities that y realizes each of its individual values, as a function of the input values $\mathbf{x}$. These probabilities are then used in a decision rule to forecast the most likely (categorical) value for the output y. Much research in classification has been devoted to achieving higher accuracy probability estimates under the presumption that this will generally lead to more accurate (categorical) prediction. The (often counter intuitive) results obtained in this paper challenge that presumption. More accurate probability estimates do not necessarily lead to better classification performance and often can make it worse.

This has implications concerning applications of classification as well as for future directions of methodological research. In individual applications one is faced with the choice of which method(s) to consider. Often older simpler techniques are discarded in favor of newly developed sophisticated methods on the grounds that the latter can provide more accurate estimates of the probabilities of the respective output values. Results derived in this paper show that, even if this is the case, it need not result in smaller classification error. Very simple procedures such as "naive" Bayes and nearest neighbor methods generally provide very poor probability estimates especially in high dimensional settings involving many input variables. Never-the-less they often yield lower prediction error than other (newer) methods intended to produce higher accuracy estimates of the probabilities. The results derived in this paper indicate those situations in which this is likely to occur. They also suggest straightforward modifications to these "simple" procedures that can sometimes improve their classification performance even more, by selectively further reducing the accuracy of their probability estimates. In terms of methodological research these results suggest that in situations where the goal is accurate classification, focusing on improved probability estimation may be misguided and a totally different paradigm may be required.

2. Classification

As noted above, in the classification problem the output variable y assumes values on an unordered discrete set $y \in \{y_1, \ldots, y_L\}$. In this paper the special (but common) case in which $L = 2$ is considered. Although many of the concepts generalize to the more general case $L \geq 3$, the derivations and underlying intuition are more straightforward for this special ("two-class") case. As will be seen, even in this restricted setting much of the conventional wisdom concerning the classification problem can be brought into question.

Without loss of generality we take $y \in \{0, 1\}$. The goal of a classification procedure is to predict the output value given the values of a set of "input" variables $\mathbf{x} = \{x_1, \ldots, x_n\}$ simultaneously measured on the same system. It is often the case that at a particular point $\mathbf{x} \in R^n$ the value of y is not uniquely determinable. It can assume both its values with respective probabilities that depend on the location of the point $\mathbf{x}$ in the n-dimensional input space

$$\Pr(y = 1 \mid \mathbf{x}) = 1 - \Pr(y = 0 \mid \mathbf{x}) \doteq f(\mathbf{x}). \tag{2.1}$$

Here $f(\mathbf{x})$ is a single valued deterministic function that at every point $\mathbf{x} \in R^n$ specifies the probability that y assumes its second value.

The role of a classification procedure is to produce a rule that makes a prediction $\hat{y}(\mathbf{x}) \in \{0, 1\}$ for the correct class label y at every input point $\mathbf{x}$. The goal is to choose $\hat{y}(\mathbf{x})$ to minimize inaccuracy as characterized by the misclassification "risk" (expected or average loss)

$$r(\mathbf{x}) = l_1 f(\mathbf{x}) 1(\hat{y}(\mathbf{x}) = 0) + l_0(1 - f(\mathbf{x}))1(\hat{y}(\mathbf{x}) = 1). \tag{2.2}$$

Here l_0 and l_1 are the losses incurred for the respective misclassifications, $f(\mathbf{x})$ is given by (2.1), and $1(\cdot)$ is an indicator function of the truth of its argument

$$1(\eta) = \begin{cases} 1 & \text{if } \eta \text{ is true} \\ 0 & \text{otherwise.} \end{cases} \tag{2.3}$$

The misclassification risk (2.2) is minimized by the ("Bayes") rule

$$y_B(\mathbf{x}) = 1\left(f(\mathbf{x}) \geq \frac{l_0}{l_0 + l_1} \right) \tag{2.4}$$

which (by definition) achieves the lowest possible risk

$$r_B(\mathbf{x}) = \min(l_1 f(\mathbf{x}), l_0(1 - f(\mathbf{x})). \tag{2.5}$$

Note that the rule (2.4) is not necessarily the only minimizer of (2.2). Other rules may also achieve minimum risk (2.5). Also, in the special (but common) case $l_0 = l_1 = 1$, (2.4) reduces to predicting the most probable class and (2.5) represents the fraction of erroneous predictions thereby encountered.

Generally the function $f(\mathbf{x})$ (2.1) characterizing a particular system is unknown. However, data from the system is available in the form of a collection of previously solved cases in which both the input and output variables have been measured. This "training" data set (1.1) is used to "learn" a classification rule $\hat{y}(\mathbf{x} \mid T)$ for (future) prediction. The usual paradigm for accomplishing this is to use the training data T (1.1) to form an approximation (estimate) $\hat{f}(\mathbf{x} \mid T)$ to $f(\mathbf{x})$ (2.1) and substitute this into (2.4)

$$\hat{y}(\mathbf{x} \mid T) = 1\left(\hat{f}(\mathbf{x} \mid T) \geq \frac{l_0}{l_0 + l_1} \right). \tag{2.6}$$

When $\hat{f}(\mathbf{x}) \neq f(\mathbf{x})$ this rule (2.6) may be different than from the Bayes rule (2.4) and thus not achieve the minimal Bayes risk (2.5). It is the purpose of this study to examine the way in which inaccuracies $f(\mathbf{x}) - \hat{f}(\mathbf{x})$ in the function estimate are reflected in misclassification risk.

3. Function estimation

In the usual function estimation setting one assumes that an output variable y is related to a set of input variables $\mathbf{x}$ by

$$y = f(\mathbf{x}) + \varepsilon \tag{3.1}$$

where $f(\mathbf{x})$ ("target function") is a single valued deterministic function of n arguments and ε is a random variable distributed according to some law $\varepsilon \sim L(\varepsilon \mid \mathbf{x})$. By definition its average $E(\varepsilon \mid \mathbf{x}) = 0$ for all $\mathbf{x}$ so that the target function is defined by

$$f(\mathbf{x}) = E(y \mid \mathbf{x}). \tag{3.2}$$

The goal is to obtain an estimate

$$\hat{f}(\mathbf{x} \mid T) = \hat{E}(y \mid \mathbf{x}, T) \tag{3.3}$$

using a training data set T (1.1). Inaccuracy is usually quantified by root-mean-squared prediction error

$$\text{rms}(\mathbf{x}) = E_{\varepsilon}^{1/2}[(y - \hat{f}(\mathbf{x} \mid T))^2] \tag{3.4}$$

where the expected value in (3.4) is taken with respect to the distribution of ε (3.1).

The classification problem can be cast in this function estimation setting by observing that (3.2) holds for y and $f(\mathbf{x})$ in (2.1) so that they can be related by (3.1) with ε distributed as a (centered) binomial distribution with variance $\text{var}(\varepsilon \mid \mathbf{x}) = f(\mathbf{x})(1 - f(\mathbf{x}))$. Thus, regular function estimation technology (3.3) can be applied to obtain the estimate $\hat{f}(\mathbf{x} \mid T)$, which is plugged into (2.6) to form a classification rule. This is the paradigm used by many popular classification methods including neural networks (Lippmann, 1989), decision tree induction methods (Breiman et al., 1984; Quinlan, 1993), projection pursuit (Friedman, 1985), and nearest neighbor methods (Fix and Hodges, 1951).

4. Density estimation

An alternative paradigm for estimating $f(\mathbf{x})$ (2.1) in the classification setting is based on density estimation. Here Bayes theorem

$$f(\mathbf{x}) = \frac{\pi_1 p_1(\mathbf{x})}{\pi_0 p_0(\mathbf{x}) + \pi_1 p_1(\mathbf{x})} \tag{4.1}$$

is applied where $\{p_j(\mathbf{x}) = \Pr(\mathbf{x} \mid y = j)\}_0^1$ are the class conditional probability density functions and $\{\pi_j = \Pr(y = j)\}_0^1$ are the unconditional ("prior") probabilities of each class. The training data (1.1) are partitioned into subsets $T = \{T_0, T_1\}$ with the same class label. The data in each subset are separately used to estimate its respective probability density

$\{\hat{p}_j(\mathbf{x} \mid T_j)\}_0^1$. These estimates are plugged into (4.1) to obtain an estimate $\hat{f}(\mathbf{x} \mid T)$ which is in turn plugged into (2.6) to form a classification rule. Examples of this approach are discriminant analysis (see McLachlan, 1992), kernel discriminant methods (Hand, 1982), Gaussian mixtures (Chow and Chen, 1992), learning vector quantization techniques (Kohonen, 1990), and Bayesian belief networks (Heckerman et al., 1994).

5. Bias, variance, and estimation error

Whether one applies the function estimation (3.3) or density estimation (4.1) approach, the estimated probability $\hat{f}(\mathbf{x} \mid T)$ depends upon the training data T (1.1) used to obtain it. A change in the data (usually) results in a change in the probability estimate at (at least) some of the input points $\mathbf{x}$. In most applications the training data set (1.1) represents a random sampling from the system under study. Even if the target probability function $f(\mathbf{x})$ is everywhere stationary, sampling the system at different times results in (at least somewhat) different training data sets T and thereby different estimates $\hat{f}(\mathbf{x} \mid T)$. For a given set of input points $\{\mathbf{x}_i\}_1^N$ the corresponding outputs $\{y_i\}_1^N$ are random owing to the stochastic component ε (3.1), and generally the input points themselves represent a random sampling (observational study).

The random nature of the training data T implies that the estimate $\hat{f}(\mathbf{x} \mid T)$ is a random variable that assumes a distribution of values at each input point $\mathbf{x}$ governed by some (usually unknown) probability law $\hat{f}(\mathbf{x}) \sim L(\hat{f} \mid \mathbf{x})$ characterized by a probability density function $p(\hat{f} \mid \mathbf{x})$. Estimating $f(\mathbf{x})$ with a particular training data set T gives rise to a particular random realization of $\hat{f}(\mathbf{x} \mid T)$ with relative probability $p(\hat{f} \mid \mathbf{x})$. Two important parameters of any such distribution are its first two moments, the mean (expected value)

$$E\,\hat{f}(\mathbf{x}) = \int_{-\infty}^{\infty} \hat{f}\, p(\hat{f} \mid \mathbf{x})\, d\hat{f} \tag{5.1}$$

and variance

$$\operatorname{var}\hat{f}(\mathbf{x}) = \int_{-\infty}^{\infty} (\hat{f} - E\,\hat{f}(\mathbf{x}))^2\, p(\hat{f} \mid \mathbf{x})\, d\hat{f}. \tag{5.2}$$

Both of these quantities directly affect expected prediction error (3.4) through the well known decompositions

$$E_T[y - \hat{f}(\mathbf{x} \mid T)]^2 = E_T[f(\mathbf{x}) - \hat{f}(\mathbf{x} \mid T)]^2 + E_\varepsilon[\varepsilon \mid \mathbf{x}]^2 \tag{5.3}$$

and

$$E_T[f(\mathbf{x}) - \hat{f}(\mathbf{x} \mid T)]^2 = [f(\mathbf{x}) - E_T\,\hat{f}(\mathbf{x} \mid T)]^2 + E_T[\hat{f}(\mathbf{x} \mid T) - E_T\,\hat{f}(\mathbf{x} \mid T)]^2. \tag{5.4}$$

The left side of (5.3) represents the squared prediction error (at $\mathbf{x}$) averaged over repeatedly realized training samples of the (same) size N from the system under study. The last term

in (5.3) is independent of both the target function and the training sample and reflects the irreducible prediction error due to the random nature of the output variable (2.1), (3.1). The other term in (5.3) is the squared "estimation error" in the target function $f(\mathbf{x})$ averaged over training samples. This depends on $f(\mathbf{x})$ and the method used to obtain $\hat{f}(\mathbf{x} \mid T)$. From (5.4) one sees that this quantity depends only on the mean (5.1) and the variance (5.2) of the distribution of $\hat{f}(\mathbf{x} \mid T)$. The last quantity in (5.4) is just the variance (5.2). The other quantity in (5.4) is the square of the "bias"

$$\text{bias}\ \hat{f}(\mathbf{x}) = f(\mathbf{x}) - E\ \hat{f}(\mathbf{x}). \tag{5.5}$$

The variance (5.2) reflects the sensitivity of the function estimate $\hat{f}(\mathbf{x} \mid T)$ to the training sample T. Less sensitivity means that the estimate will be more stable against changes (sampling variations) in the data and thus be less variable under repeated sampling. The bias (5.5) reflects sensitivity to the target function $f(\mathbf{x})$. It represents how closely on average the estimate is able to approximate the target. From (5.4) one sees that it is desirable to have both low bias-squared and low variance since both contribute to the squared estimation error in equal measure. There is however a tension between these goals (Geman et al., 1992). The purpose of training is to gain information concerning the target function from the data; therefore sensitivity to the training data is essential, and generally more sensitivity results in lower bias. However, this in turn increases variance and so there is a natural "bias-variance trade-off" associated with function approximation.

For a given bias (5.5) the variance (5.2) generally decreases with increasing training sample size N (1.1). Therefore for problems with large training samples the bias can be the dominant contributor to estimation error. Since larger and larger data bases are becoming routinely available, most modern research in learning methodology has focused on increasingly flexible techniques that reduce estimation bias, some with considerable success. From (5.3), (5.4) one can see that this is a reasonable strategy for function approximation (based on root-mean-squared error (3.4)), provided enough attention is paid to the variance ("overfitting"). For classification however this strategy has been less successful in improving performance. Some simple highly biased procedures such as "naive" Bayes (Good, 1965) and nearest neighbor methods (Fix and Hodges, 1951) remain competitive with and sometimes outperform more sophisticated ones, even with moderate to large training samples (Holte, 1993). In the next section we show that this may not be as surprising as it seems. The quantities $E\,\hat{f}(\mathbf{x})$ (5.1) and $\text{var}\,\hat{f}(\mathbf{x})$ (5.2) that characterize the distribution of $\hat{f}(\mathbf{x} \mid T)$ conspire to affect classification error in a very different way when $\hat{f}(\mathbf{x} \mid T)$ is used in a classification rule (2.6).

6. Bias, variance, and classification error

For concreteness we take $l_0 = l_1 = 1$ in (2.4), (2.6) so that the threshold in the indicator function is $1/2$ and misclassification risk (2.2) reduces to probability of misclassification $\Pr(\hat{y}(\mathbf{x}) \neq y)$. The first step in uncovering how $E\,\hat{f}(\mathbf{x})$ (5.1) and $\text{var}\,\hat{f}(\mathbf{x})$ (5.2) affect classification error is to decompose it into the irreducible error associated with the random nature of y (2.1), (3.1) and a reducible part that depends on $\hat{f}(\mathbf{x})$ (3.3) in analogy with

(5.3) for squared-error loss. Given a particular training sample T (1.1) the error rate $\Pr(\hat{y}(\mathbf{x} \mid T) \neq y)$ (averaged over all future predictions at $\mathbf{x}$) depends on whether or not the decision (2.6) agrees with that of the Bayes rule (2.4). If it does then its error rate is the irreducible error associated with the Bayes rule (2.5) $\Pr(\hat{y}(\mathbf{x} \mid T) \neq y) = \Pr(y_B(\mathbf{x}) \neq y) = \min[f(\mathbf{x}), 1 - f(\mathbf{x})]$. If not, then it suffers an increased error rate $\Pr(\hat{y}(\mathbf{x} \mid T) \neq y) = \max[f(\mathbf{x}), 1 - f(\mathbf{x})] = |2f(\mathbf{x}) - 1| + \Pr(y_B(\mathbf{x}) \neq y)$. Therefore one has

$$\Pr(\hat{y}(\mathbf{x} \mid T) \neq y) = |2f(\mathbf{x}) - 1|\, 1[\hat{y}(\mathbf{x} \mid T) \neq y_B(\mathbf{x})] + \Pr(y_B(\mathbf{x}) \neq y). \tag{6.1}$$

Averaging over all training samples T (of size N), under the assumption that they are drawn independently of future data to be predicted, one has

$$\Pr(\hat{y} \neq y) = |2f - 1| \Pr(\hat{y} \neq y_B) + \Pr(y_B \neq y). \tag{6.2}$$

Here (6.2) and in what follows all quantities are presumed to be conditioned at a particular point $\mathbf{x}$ in the input space, and that explicit dependence is suppressed for convenience.

From (6.2) one sees that the classification error rate $\Pr(\hat{y} \neq y)$ is linearly proportional to $\Pr(\hat{y} \neq y_B)$ which is the only quantity in (6.2) that involves the probability estimate $\hat{f}$ through (2.6). It can be viewed as a decision "boundary" error in that it represents misestimation of the (optimal) decision boundary separating the two classes in the input space, defined by the set of points (surface) for which $f(\mathbf{x}) = 1/2$. This "boundary error" is the analog of the (squared) estimation error $E[f(\mathbf{x}) - \hat{f}(\mathbf{x})]^2$ in (5.3) and (5.4).

In order to proceed further it is necessary to calculate how the boundary error depends on the distribution of $\hat{f}$, $p(\hat{f})$, induced by the random variations in training data sets T as (repeatedly) sampled from the system under study. This is just the tail area of $p(\hat{f})$ on the opposite side of the value $1/2$ from the true probability f

$$\Pr(\hat{y} \neq y_B) = 1(f < 1/2) \int_{1/2}^{\infty} p(\hat{f})\; d\hat{f} + 1(f \geq 1/2) \int_{-\infty}^{1/2} p(\hat{f})\; d\hat{f}. \tag{6.3}$$

This will depend on the detailed form of the distribution $p(\hat{f})$, and not just on its first two moments (5.1), (5.2), as was the case with squared estimation error (5.4). In order to gain some intuition we approximate $p(\hat{f})$ by a normal distribution

$$p(\hat{f}) = \frac{1}{\sqrt{2\pi \operatorname{var} \hat{f}}} \exp\left(-\frac{1}{2} \frac{(\hat{f} - E\,\hat{f})^2}{\operatorname{var} \hat{f}}\right). \tag{6.4}$$

This approximation is often reasonable since for many procedures the computation of $\hat{f}$ involves a (sometimes complex) averaging process. Even when it is not the case the qualitative conclusions are still generally valid. Assuming (6.4) the boundary error (6.3) becomes

$$\Pr(\hat{y} \neq y_B) = \tilde{\Phi}\left[\operatorname{sign}(f - 1/2) \frac{E\,\hat{f} - 1/2}{\sqrt{\operatorname{var} \hat{f}}}\right] \tag{6.5}$$

where

$$\tilde{\Phi}(z) = \frac{1}{\sqrt{2\pi}} \int_z^\infty e^{-\frac{1}{2}u^2} du \tag{6.6}$$

is the upper tail area of the standard normal distribution.

7. Discussion

Inspection of (6.5) reveals that the boundary error depends upon the true probability f, and the systematic component of the estimate $E\,\hat{f}$, through

$$b(f, E\,\hat{f}) = \text{sign}(1/2 - f)(E\,\hat{f} - 1/2). \tag{7.1}$$

For any (nonzero) value of the random component, $\text{var}\,\hat{f} > 0$, the boundary error rate $\Pr(\hat{y} \neq y_B)$ is monotonically increasing in $b(f, E\,\hat{f})$. In this sense it can be viewed as an analog of the estimation bias (5.5) squared for squared-error loss (5.4). For convenience we refer to $b(f, E\,\hat{f})$ (7.1) as the "boundary bias". (In cases where $p(\hat{f})$ is an asymmetric distribution it is more natural to define boundary bias (7.1) in terms of the median instead of the mean $E\,\hat{f}$.)

Comparison of (6.2), (6.5) with (5.3), (5.4) reveals that the quantities $E\,\hat{f}$ and $\text{var}\,\hat{f}$ affect classification error very differently than they affect estimation error on the probability f itself. For a given $\text{var}\,\hat{f}$, estimation (squared) error (5.4) is proportional to the (squared) distance $(f - E\,\hat{f})^2$ (bias-squared). In classification (6.2), (6.5) the dependence on f is only through the sign of $f - 1/2$, and the relevant quantity is boundary bias (7.1). Therefore, so long as boundary bias is negative $b(f, E\,\hat{f}) < 0$ classification error decreases with increasing $|E\,\hat{f} - 1/2|$ irrespective of the estimation bias $(f - E\,\hat{f})$. For positive boundary bias the classification error increases with the distance of $E\,\hat{f}$ from $1/2$.

For a given value of $E\,\hat{f}$, estimation (squared) error (5.4) is proportional to $\text{var}\,\hat{f}$. For classification error the effect of the $\text{var}\,\hat{f}$ depends mostly on the sign of the boundary bias (7.1). For a negative sign classification error decreases with decreasing variance (though not linearly), whereas for a positive sign the error rate *increases* with decreasing variance. The rate of increase/decrease depends on the absolute boundary bias. With estimation error (5.4) small variance does not necessarily provide small error; the bias-squared might be quite large. For classification, zero variance results in optimal classification (Bayes rule) irrespective of the value of the estimation bias (5.5) provided boundary bias (7.1) is negative. For positive boundary bias, zero variance gives rise to maximal error rate (certain boundary error) at $\mathbf{x}$. Note that imposing the constraint $0 \leq \hat{f}(\mathbf{x}) \leq 1$, while often improving estimation bias, need not improve boundary bias. In fact, it could increase boundary bias and thereby boundary error (6.5) unless a requisite reduction in variance is achieved through the constraint.

The "bias-variance trade-off" is clearly very different for classification error than estimation error on the probability function f itself. The dependence of squared estimation error (5.4) on $E\,\hat{f}$ and $\text{var}\,\hat{f}$ is additive (bias-squared plus variance) whereas for classification error (6.2), (6.5) there is a strong (multiplicative) interaction effect. The effect of boundary bias (7.1) on classification error (6.2), (6.5) can be mitigated by low variance. Similarly, the affect of the variance depends on the value (especially the sign) of the boundary bias.

Therefore low variance (5.2) can be very important for classification but low (estimation) bias (5.5) squared is not. For the most part, all that is required of $E\,\hat{f}$ is to insure that it be on the same side of the value $1/2$ as f (negative boundary bias). This being the case, one can reduce classification error toward its minimal (Bayes) value by reducing variance alone. In this sense variance tends to dominate bias for classification.

This different "bias-variance trade-off" for classification error (6.5) suggests that certain methods that are inappropriate for function estimation because of their very high bias (5.4), (5.5) may none-the-less perform well for classification when their (highly biased) estimates are used in the context of a classification rule (2.6). All that is required is predominately negative boundary bias (7.1) and small enough variance. Among these are procedures for which the bias is caused by "over-smoothing"; the estimate at each point $\mathbf{x}$, $\hat{f}(\mathbf{x})$, tends to be shrunk towards the mean output value

$$\bar{y} = \frac{1}{N}\sum_{i=1}^{N} y_i. \tag{7.2}$$

That is, the result of applying the procedure tends to be

$$\hat{f}(\mathbf{x}) = (1 - \alpha(\mathbf{x}))\, f(\mathbf{x}) + \alpha(\mathbf{x})\bar{y} \tag{7.3}$$

where $0 \leq \alpha(\mathbf{x}) \leq 1$ represents an "over-smoothing" coefficient that usually depends on $\mathbf{x}$. The larger the value for $\alpha(\mathbf{x})$ the more over-smoothing bias. So long as $\bar{y} = 1/2$ (equal number of each class in the training sample) then boundary bias is negative (for all $\mathbf{x}$) and $\text{var}\,\hat{f}(\mathbf{x})$ is likely to dominate classification error for such procedures. (Generalization to $\bar{y} \neq 1/2$ is discussed in Section 10.) The variance is also controlled by the degree of (over) smoothing-more smoothing less variance. Therefore, the optimal amount of smoothing for minimizing classification error (6.2), (6.5) is likely to be much larger than that for estimation error (5.4) since the latter is more strongly affected by estimation bias (5.5).

8. "Naive" Bayes methods

The naive Bayes approach is surprisingly effective (Titterington et al., 1981; Langley et al., 1992) given the crude nature of its approximation. It uses the density estimation paradigm (Section 4) and approximates each class conditional probability density $\{p_k(\mathbf{x})\}_0^1$ (4.1) by the product of its marginal densities $\{p_j^{(k)}(x_j)\}_{j=1}^n$ on each input variable,

$$\tilde{p}_k(\mathbf{x}) = \prod_{j=1}^{n} p_j^{(k)}(x_j) \tag{8.1}$$

with

$$p_j^{(k)}(x_j) = \int p_k(\mathbf{x}) \prod_{l \neq j} dx_l. \tag{8.2}$$

Data from each class $k \in \{0, 1\}$, and each input variable $j \in \{1, 2, \ldots, n\}$, are separately used to obtain corresponding estimates $\hat{p}_j^{(k)}(x_j)$ of (8.2). These are used in (8.1), which is

in turn plugged into (4.1) to form an estimate of $f(\mathbf{x})$. This is then inserted into (2.6) to produce an output estimate.

Estimates $\hat{f}(\mathbf{x})$ obtained in this manner are clearly biased (5.5) estimates of $f(\mathbf{x})$ (2.1), (3.1), even if the true marginal densities (8.2) are used, unless the input variables for every class happen to be totally independent. Since such total independence is far from being realized in most applications, this bias can be quite large, especially when there are many inputs. Introducing estimates for (8.2) can introduce further bias and, of course, variance as well.

The high degree of bias (5.4), (5.5) associated with the naive Bayes method (8.1), (8.2) makes it generally unsuitable for accurately approximating the target probability function $f(\mathbf{x})$ (2.1), (3.1). However, this bias is generally of the "over-smoothing" variety discussed in Section 7. The approximating densities $\tilde{p}_k(\mathbf{x})$ tend to be much smoother than the corresponding (true) densities $p_k(\mathbf{x})$ from which they are derived. They place substantive mass over broader regions of the input space as evidenced by the fact that the entropy of $\tilde{p}_k(\mathbf{x})$ is (usually much) greater than that of $p_k(\mathbf{x})$. This over-smoothing of the class conditional densities produces an over-smoothed estimate of $f(\mathbf{x})$ when inserted into (4.1), producing (usually large) estimation bias (5.5), and therefore error (5.4). However, as discussed in Section 7, the boundary bias (7.1) produced by this mechanism is likely to remain negative over much of the input space so that low variance estimates of the marginal densities (8.2) can produce low boundary error (6.5). This fact may explain why the naive Bayes method has seen so much success in classification despite its "naive" approach.

9. *K*-nearest neighbor methods

Another class of highly biased estimation procedures are those based on K-nearest neighbors. A local subregion $R(\mathbf{x}) \subset R^n$ of the input space, centered at the estimation point $\mathbf{x}$, is constructed and the target function estimate is taken to be the average of the training sample output values (1.1) in that region

$$\hat{f}(\mathbf{x}) = \text{ave}_{\mathbf{x}_i \in R(\mathbf{x})} y_i . \tag{9.1}$$

The predicting region $R(\mathbf{x})$ is defined to be the subregion of the input space containing the K closest training points to $\mathbf{x}$

$$R(\mathbf{x}) = \{\mathbf{x}' \mid \|\mathbf{x} - \mathbf{x}'\| \leq d_{(K)}\} \tag{9.2}$$

where $d_{(K)}$ is the Kth order statistic of $\{\|\mathbf{x} - \mathbf{x}_i\|\}_1^N$. This method requires the definition of a distance $\|\mathbf{x} - \mathbf{x}'\|$ on the input space. This is usually taken to be a (weighted) l_q distance

$$\|\mathbf{x} - \mathbf{x}'\| = \left[\sum_{j=1}^{n} |w_j (x_j - x'_j)|^q \right]^{1/q} \tag{9.3}$$

with $q = 2$ (Euclidean distance) the most common choice. The weights $\{w_j\}_1^n$ are usually chosen to be inversely proportional to the (global) scales of the respective input variables so as to give each input equal influence in defining the region (9.2).

For K-nearest neighbor procedures the bias-variance trade-off associated with estimation error generally is driven by the bias (5.5) in high dimensional settings (many inputs). This is due to the geometry of Euclidean spaces; the radius of a region varies as the nth root of its volume, whereas the number of training points in the region K varies roughly linearly with the volume. Thus, even the smallest possible volume ($K = 1$) gives rise to large regions in terms of radius. This can already produce high bias even for the largest variance ($K = 1$). This phenomenon is referred to as the "curse-of-dimensionality" (Bellman, 1961).

Like naive Bayes (Section 8), the bias (5.5) associated with K-nearest neighbor procedures is produced by over-smoothing. In fact, as $K \to N$, $\hat{f}(\mathbf{x}) \to \bar{y}$ (7.2) for all $\mathbf{x}$. Provided $\bar{y} = 1/2$ the boundary bias generally tends to be negative, and decreasing the variance can have dramatic impact on reducing boundary error (6.5).

This is illustrated with a simple example. The input space is taken to be the n-dimensional unit hypercube $\mathbf{x} \in [0, 1]^n$. The class densities are

$$p_0(\mathbf{x}) = 2 \cdot 1(x_1 < 1/2), \quad p_1(\mathbf{x}) = 2 \cdot 1(x_1 \geq 1/2) \tag{9.4}$$

so that the target probability function is

$$f(\mathbf{x}) = 1(x_1 \geq 1/2). \tag{9.5}$$

The prior probabilities (4.1) are taken to be equal ($\pi_0 = \pi_1$). This target (9.5) is a simple function of x_1 only, so having additional inputs serves to increase K-nearest neighbor bias (5.5) at the maximal rate since these inputs contain no additional information. Note that the irreducible squared-prediction error $E_\varepsilon[\varepsilon \mid \mathbf{x}]^2$ (5.3) for this problem (9.5) is zero and thus the minimal (Bayes) error rate (2.5) is also zero.

Table 1 shows the values of average squared estimation error (Column 2) and classification error (Column 4) as a function of training sample size N (first column) along with the corresponding optimal values (K_e and K_c, respectively) of the number of nearest neighbors (third and fifth columns) for this example (9.4), (9.5) at $n = 20$ dimensions. One sees that classification error is decreasing at a much faster rate than squared estimation error as N increases. The optimal value of K for squared estimation error (third column) is seen to be very slowly increasing with N. As the training sample is increased the additional

Table 1. Error rates and optimal K as a function of N for $n = 20$.

N	Estimation^2	K_e	Classification	K_c
100	.165	10	.165	57
200	.144	11	.127	67
400	.132	13	.089	205
800	.120	14	.060	417
1600	.108	15	.039	773
3200	.099	17	.029	1651
6400	.091	15	.018	2029
12800	.083	17	.013	7953

Table 2. Estimation and classification error as a function of n for $N = 12800$.

n	Estimation[2]	Classification
2	.0023	.0022
3	.0079	.0041
5	.0213	.0055
10	.0467	.0091
20	.0829	.0130

data are being used to reduce the radius of the regions in a (not very successful) attempt to reduce the impact of the bias (5.5) contribution to squared estimation error (5.4). For classification error, the optimal value for K (last column) is much larger and increases more rapidly (almost linearly) with increasing N. The additional data are being used to reduce variance of the estimates $\hat{f}(\mathbf{x})$. Because of its interaction (6.5) with (boundary) bias (7.1) reducing variance has a much bigger impact on classification error. This results in much faster decrease in classification error with increasing N. These (and all following) results were obtained through Monte Carlo simulation using 20 replications at each training sample size and 20000 independent (test) observations.

Table 2 shows the relationship of both squared estimation and classification error with dimension n for the largest training sample size ($N = 12800$) considered here. One sees that classification error is not completely immune to the tendency of K-nearest neighbor methods to degrade as irrelevant inputs are included. But whereas the squared estimation error degrades by over a factor of 35 as the number of irrelevant inputs is increased by a factor of 20, the corresponding increase in classification error is less than a factor of six. In this sense one can say that dimensionality is a "problem" here for classification, whereas for estimation error it definitely qualifies as a "curse".

An important aspect contributing to the successful resistance of classification error to the curse-of-dimensionality is the choice of a good value for the number of nearest neighbors K. The discussion in Section 7 suggests that this should be typically larger for classification than for estimation error. This is verified in Table 1 for our simple example (9.5). Figure 1 shows plots of the typical dependence of both squared estimation error (upper frame) and classification error (lower frame) on K (here $n = 20$, $N = 3200$). One sees that choice of number of nearest neighbors is less critical for classification error so long as K is neither too small nor too large (here $500 \leq K \leq 2000$). However, it must be substantially larger than the optimal value for estimation error (Table 1) in order to obtain near optimal classification performance. Quite often when K-nearest neighbors are compared to other classification methods a small value ($K = 1$ or $K = 5$, for example) is used. The simple example examined here suggests that, at least in some situations, this may underestimate the performance achievable with the K-nearest neighbor approach. This was dramatically demonstrated by Rosen et al. (1995), and noted by Henley and Hand (1996), in the context of specific (real data) problems.

The example (9.4), (9.5) studied here is a very simple one intended to illustrate the concepts involved. It was specifically designed to be highly susceptible to the effects of the curse-of-dimensionality. It may well not be representative of many classification problems,

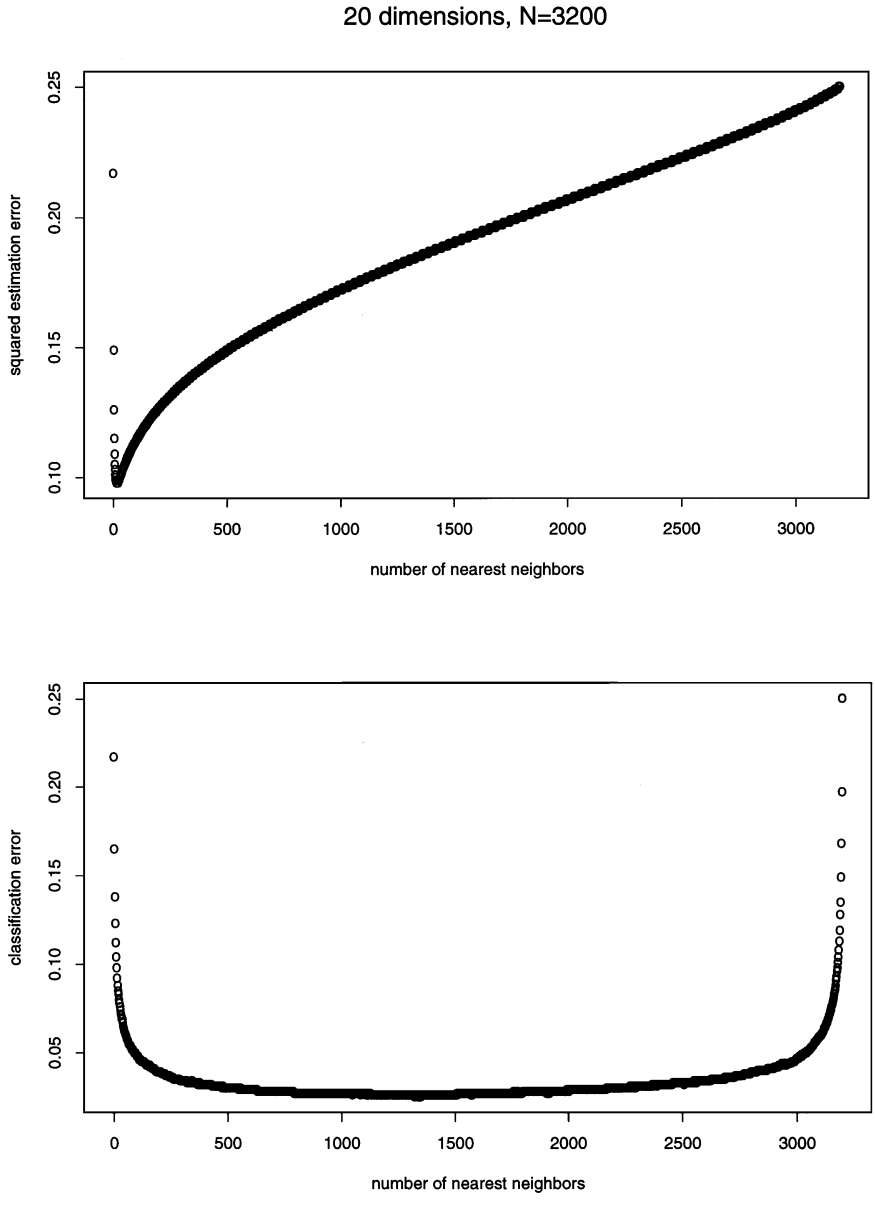

Figure 1. Squared estimation error (upper) and classification error (lower) as a function of number of nearest neighbors K, for $n = 20$ dimensions and training sample size $N = 3200$.

especially those with very complicated decision boundaries. It does however illustrate the different nature of the bias-variance trade-off in classification, and suggests that much of the conventional wisdom, derived from intuition based on (function) estimation, may not be directly applicable to the classification problem.

Another potential limitation of the study presented here is that only dimensionalities up to $n = 20$ were considered. In many problems, especially those involving signals and images, there may be hundreds or even thousands of input variables. However, in the context of nearest neighbor methods the number of inputs is not the relevant factor. The important quantity is the (local) *intrinsic* dimensionality of the joint distribution of input values as characterized by the number of its singular values that are not small. Especially when there are many inputs there is usually a high degree of association among them so that the corresponding intrinsic dimensionality is fairly moderate. In such cases the results presented here will likely be relevant.

10. Boundary bias

An important ingredient contributing to the success of both naive Bayes and K-nearest neighbor procedures is negative boundary bias (7.1). So long as $b(f(\mathbf{x}), E\hat{f}(\mathbf{x})) < 0$ they can use decreasing variance to overcome its (increasing) effect on boundary error (6.5) to produce accurate classification at $\mathbf{x}$. It is the (over-smoothing) nature of the (large) bias inherent in these methods that leads to predominately negative boundary bias at most input points $\mathbf{x}$, and thereby good overall classification performance. Non-negative boundary bias on the other hand devastates classification performance. In this case the boundary error is greater than $1/2$ and decreasing variance *increases* that error. At such points $\mathbf{x}$ the classification procedure has no alternative but to try to reduce estimation bias (5.5) in an attempt to bring boundary bias down to a negative value. This generally involves an increase in variance and the favorable trade-off produced by their multiplicative interaction effect (6.5) is lost.

The devastating effect of positive boundary bias in the context of K-nearest neighbor procedures is illustrated by a simple example. This example is the same as that used in Section 9 (9.4), (9.5) but with a modification to the prior probabilities (4.1). Here we take them to be unequal, specifically $\pi_1 = 3\pi_0$, so that the value of the output mean (7.2) is $\bar{y} = 3/4$. Table 3 shows the values of average squared estimation error (second column), classification error (fourth column), along with their respective optimal number of nearest neighbors K (columns 3 and 5) as a function of sample size N (first column), at $n = 20$

Table 3. Error rates and optimal K as a function of sample size N for $n = 20$ with $\pi_1 = 3\pi_0$.

N	Est.[2]	K_e	Class	K_c	Class($t = -1/4$)	$K_c(t = -1/4)$
100	.136	11	.195	5	.154	44
200	.125	11	.176	5	.110	96
400	.116	10	.164	5	.081	192
800	.109	14	.154	7	.060	372
1600	.100	15	.141	7	.043	772
3200	.092	17	.129	7	.029	1488
6400	.086	13	.118	7	.024	3292
12800	.078	14	.105	9	.016	4400

dimensions. For squared estimation error one sees similar results to that shown in Table 1 for equal priors ($\pi_0 = \pi_1$). Error is large and decreases slowly with increasing N. For classification error however one sees a quite different result for unequal priors ($\pi_1 = 3\pi_0$). Classification error is here much larger than for equal priors and decreases very slowly with increasing N at a rate similar to that for squared estimation error. The number of nearest neighbors that minimize classification error is also very different for unequal priors; they are even smaller than those for squared estimation error and increase very slowly with increasing N. For unequal priors ($\bar{y} \neq 1/2$) classification error is suffering from the curse-of-dimensionality in the same way as squared estimation error.

It is easy to see that the problem with K-nearest neighbors in this setting is positive boundary bias over much of the input space. The over-smoothed nature of the estimates causes them to be shrunk towards the output mean $\bar{y}$ (7.3) and in this case $\bar{y}$ is not equal to the classification threshold (2.6), here $1/2$. The boundary bias is non-negative $b(f(\mathbf{x}), E\,\hat{f}(\mathbf{x})) \geq 0$ at all input points $\mathbf{x}$ for which $x_1 < 1/2$ ($f(\mathbf{x}) = 0$) and $1/3$ or more of the volume of the K-nearest neighborhood overlaps the class one region $x_1 \geq 1/2$ ($E\,\hat{f}(\mathbf{x}) \geq 1/2$). As the dimension n increases the average radius of the regions (9.2), (9.3) increases (for fixed K) so that the portion of the input space with positive boundary bias also increases. The only way to mitigate this effect is to reduce the value of K and thereby average region radius. For increasing n this strategy becomes less effective owing to the curse-of-dimensionality; average radius varies as the nth root of K. At high dimensions there is considerable positive boundary bias even for $K = 1$. Therefore, one sees slow decrease for average classification error with increasing N, typical of that associated with the curse-of-dimensionality.

For this particular example there is a simple remedy for this problem. One can simply apply the procedure as if the prior probabilities were equal, even though there are three times as many class ones as class zeros in both the training data and future data to be classified. This involves weighting each class zero training observation with three times the mass of each class one in the average leading to the computation of $\hat{f}(\mathbf{x})$ (9.1). This simple trick causes $\bar{y} = 1/2$ (7.2) and thereby produces negative boundary bias everywhere in the input space for this problem. Applying such a weighting scheme is equivalent to modifying the estimate $\hat{f}(\mathbf{x})$ by the transformation

$$\tilde{f}(\mathbf{x}) = \hat{f}(\mathbf{x}) + t \tag{10.1}$$

before inserting it into the output estimate (2.6) (here $t = -1/4$).

The sixth column of Table 3 shows the corresponding classification error using the "bias adjustment" (10.1) with $t = -1/4$, and the seventh column its corresponding optimal number of nearest neighbors. Applying the bias adjustment $t = -1/4$ (10.1) causes the boundary bias associated with $\tilde{f}(\mathbf{x})$ to be everywhere negative and allows decreasing variance (increasing K) to maximally exploit their interaction effect (6.5) to dramatically reduce classification error.

The bias adjustment (10.1) changes both estimation (5.5) and boundary (7.1) bias everywhere in the input space. The optimal value of t for estimation error is $t_e = \text{ave}_{\mathbf{x}} f(\mathbf{x}) - \text{ave}_{\mathbf{x}} \hat{f}(\mathbf{x})$. Since these two averages tend to have similar values for most estimation procedures (especially those that over-smooth) there is seldom much to be gained by employing (10.1). In the case of boundary bias (7.1) the modification (10.1) decreases its value over

half of the input space ($x_1 < 1/2$) and increases it by the same amount at each point $\mathbf{x}$ in the other half ($x_1 \geq 1/2$). Therefore average boundary bias is (substantially) increased since the (pooled) distribution of the input values places three times as much mass in the latter half space ($x_1 \geq 1/2$). However the interaction between variance and boundary bias occurs separately at each individual point $\mathbf{x}$, and the choice $t = -1/4$ (10.1) here provides the right balance so that the boundary bias is negative at all $\mathbf{x}$.

In this example a good bias adjustment value t (10.1) could be determined since the true target function (9.5) and priors ($\pi_1 = 3\pi_0$) were known. This is seldom the case in practice. Even when they are known however a good choice may not be obvious. Consider the case

$$p_0(\mathbf{x}) = (4/3) \cdot 1(x_1 < 3/4), \quad p_1(\mathbf{x}) = 4 \cdot 1(x_1 \geq 3/4) \tag{10.2}$$

with equal prior probabilities ($\pi_0 = \pi_1$), again on the hypercube $\mathbf{x} \in [0, 1]^n$. The target probability function (3.2) is

$$f(\mathbf{x}) = 1(x_1 \geq 3/4). \tag{10.3}$$

Here the response mean (7.2) is $\bar{y} = 1/2$ but there is positive boundary bias over much of the input space, caused by the higher density of class ones near the decision boundary.

Table 4 shows (for $n = 20$ dimensions) values of average squared estimation error (second column), classification error for $t = 0$ (fourth column), and classification error using the optimal value $t = t^*$ (sixth column) along with their corresponding optimal number of nearest neighbors K (Columns 3, 5, and 7, respectively). The last column of Table 4 shows the corresponding optimal value t^* of the bias adjustment t (10.1). Without the bias adjustment classification error converges to zero at roughly the same rate as squared estimation error. The bias adjustment dramatically speeds up convergence to small classification error. At $N = 12800$ classification error is more than five times smaller with the adjustment than without it. The optimal adjustment values t^* however are here much smaller (in absolute value) than in the previous example ($t^* = -0.25$). In fact, using $t = -0.25$ in this case (Table 4) produces higher classification error than no adjustment at all ($t = 0$). Thus one sees similar results to that of the previous example (Table 3). Without the bias adjustment (10.1) classification error suffers from the curse-of-dimensionality in the same manner as

Table 4. Error rates, optimal K, and t as a function of N for $n = 20$.

N	Est.[2]	K_e	Class($t = 0$)	$K_c(t = 0)$	Class($t = t^*$)	$K_c(t = t^*)$	t^*
100	.161	10	.166	53	.161	53	−.025
200	.147	11	.139	79	.121	97	−.025
400	.135	12	.120	85	.086	255	−.025
800	.126	15	.115	83	.061	442	−.025
1600	.117	18	.103	125	.046	985	−.025
3210	.108	17	.101	75	.032	2121	−.025
6400	.101	17	.100	81	.027	1949	−.05
12800	.094	20	.096	43	.018	8714	−.05

Table 5. Optimal bias adjustment value t^* for various values of dimension n and sample size N.

n	$N = 100$	$N = 400$	$N = 1600$	$N = 6400$	$N = 12800$
2	−.250	−.250	−.250	−.225	−.225
3	−.125	−.150	−.225	−.225	−.225
5	−.025	−.075	−.150	−.200	−.225
10	−.025	−.050	−.050	−.075	−.075
20	−.025	−.025	−.025	−.050	−.050

squared estimation error (here slightly worse). With the bias adjustment (convergence rate) immunity to the curse is restored.

Table 5 shows the optimal bias adjustment value t^* for selected sample sizes N (columns 2–6) as a function of dimension n (first column) for this example (10.2), (10.3). One sees that t^* depends on both n and N for this (fixed) target (10.3). At all sample sizes the absolute value of t^* tends to decrease with increasing dimension. At fixed dimension n, the absolute value tends to increase with sample size (except for $n = 2$).

As with the previous examples, this one is especially simple and may not be a close reflection of reality in many classification problems. It does illustrate that even in cases where the true target probability function and the priors are known, the best choice of bias adjustment level (10.1) may not be obvious. In reality neither are generally known so (in any case) model selection techniques such as cross-validation must be employed to estimate good joint values of t and K. Note that this requires very little added computation over that of estimating K alone. As these examples illustrate there may be considerable gains associated with such a strategy, especially in cases where the effects of the curse-of-dimensionality are hindering classification performance.

In the naive Bayes approach applying a bias adjustment (10.1) is equivalent to altering the relative prior probabilities $\{\pi_j\}_0^1$ (4.1) from those values that would be optimal when used in conjunction with the true class conditional densities $\{p_j(\mathbf{x})\}_0^1$. Since the density estimates associated with naive Bayes (8.1), (8.2) are generally highly biased (over-smoothed) estimates of the true densities, bias adjustment may be highly beneficial with it as well. Generally an optimal value for this adjustment will not be known in practice and model selection techniques (such as cross-validation) must be used to obtain an estimate.

11. Bias plus variance in classification

There has been a flurry of recent activity (Dietterich and Kong, 1995; Kohavi and Wolpert, 1996; Breiman, 1996; Tibshirani, 1996) also directed at the goal of attempting to understand the relative influence of the systematic and random components of classification error. These efforts have concentrated on developing an additive decomposition in direct analogy with the (seductively simple) form for squared estimation error (5.3), (5.4). In this section these decompositions are reviewed and related to the definitions and concepts derived in this paper.

The formulation of Kohavi and Wolpert (1996) is somewhat different than that of this paper. Central to their decomposition (for the two-class case) is the probability at $\mathbf{x}$ that the

classification procedure (2.6) predicts $\hat{y}(\mathbf{x}) = 1$

$$P_1(\mathbf{x}) = \Pr[1(\hat{f}(\mathbf{x}) \geq 1/2) = 1] = \Pr[\hat{f}(\mathbf{x}) \geq 1/2] \tag{11.1}$$

which under the Gaussian assumption (6.4) becomes

$$P_1(\mathbf{x}) = \tilde{\Phi}\left(\frac{1/2 - E\,\hat{f}(\mathbf{x})}{\sqrt{\operatorname{var}\hat{f}(\mathbf{x})}}\right). \tag{11.2}$$

Kohavi and Wolpert (1996) define the classification "bias-squared" at $\mathbf{x}$ as

$$\operatorname{bias}^2_{\mathrm{KW}}(\mathbf{x}) = [f(\mathbf{x}) - P_1(\mathbf{x})]^2, \tag{11.3}$$

the "variance" as

$$\operatorname{var}_{\mathrm{KW}}(\mathbf{x}) = P_1(\mathbf{x})[1 - P_1(\mathbf{x})], \tag{11.4}$$

the "irreducible error-squared" as

$$\sigma^2(\mathbf{x}) = f(\mathbf{x})[1 - f(\mathbf{x})], \tag{11.5}$$

and show that

$$\Pr(\hat{y}(\mathbf{x}) \neq y) = \operatorname{bias}^2_{\mathrm{KW}}(\mathbf{x}) + \operatorname{var}_{\mathrm{KW}}(\mathbf{x}) + \sigma^2(\mathbf{x}). \tag{11.6}$$

The last quantity $\sigma^2(\mathbf{x})$ is the variance of the error term $E_\varepsilon[\varepsilon^2 \mid \mathbf{x}]$ in (3.1). The definitions (11.3), (11.4) each involve both the systematic $E\,\hat{f}(\mathbf{x})$ (5.1) and random $\operatorname{var}\hat{f}(\mathbf{x})$ (5.2) components of the estimate $\hat{f}(\mathbf{x})$, and the irreducible error is not defined as the Bayes error rate (2.5). However this decomposition does have the desirable property that $\operatorname{var}_{\mathrm{KW}}(\mathbf{x}) \geq 0$ at all $\mathbf{x}$.

The formulations of Dietterich and Kong (1995), Breiman (1996), and Tibshirani (1996) are more similar to the approach adopted in this paper. Dietterich and Kong (1995) define the "statistical bias" of a classification procedure as

$$\operatorname{bias}_{\mathrm{DK}}(\mathbf{x}) = 1[\Pr(\hat{y}(\mathbf{x}) \neq y) \geq 1/2]. \tag{11.7}$$

With this definition a procedure has unit bias at an input point $\mathbf{x}$ if it makes the wrong decision there half of the time or more, as averaged over training sets T (1.1), and has zero bias otherwise. The "statistical variance" is defined as the difference between the error rate (at $\mathbf{x}$) and the statistical bias

$$\operatorname{var}_{\mathrm{DK}}(\mathbf{x}) = \Pr(\hat{y}(\mathbf{x}) \neq y) - \operatorname{bias}_{\mathrm{DK}}(\mathbf{x}) \tag{11.8}$$

so that one obtains the decomposition

$$\Pr(\hat{y}(\mathbf{x}) \neq y) = \operatorname{bias}_{\mathrm{DK}}(\mathbf{x}) + \operatorname{var}_{\mathrm{DK}}(\mathbf{x}). \tag{11.9}$$

From (6.2), (6.5), (7.1) one sees that $\text{bias}_{\text{DK}}(\mathbf{x})$ is

$$\text{bias}_{\text{DK}}(\mathbf{x}) = 1[b(f(\mathbf{x}), E\,\hat{f}(\mathbf{x})) \geq 0] \tag{11.10}$$

so that it is just an indicator of positive boundary bias (7.1) at $\mathbf{x}$. The quantity $\text{var}_{\text{DK}}(\mathbf{x})$ (11.8) involves both the systematic and random components $(E\,\hat{f}(\mathbf{x}), \text{var}\,\hat{f}(\mathbf{x}))$ of the estimate $\hat{f}(\mathbf{x})$, as well as the Bayes error rate $\Pr(y_B(\mathbf{x}) \neq y)$ (2.5), in a fairly complicated way, and can assume negative values.

Breiman (1996) defines bias and variance in terms of the "reducible" error rate

$$r(\mathbf{x}) = \Pr(\hat{y}(\mathbf{x}) \neq y) - \Pr(y_B(\mathbf{x}) \neq y) \tag{11.11}$$

where $\Pr(y_B(\mathbf{x}) \neq y)$ is the Bayes error rate (2.5), and in terms of an "aggregated" classifier which in the notation of this paper is

$$y_A(\mathbf{x}) = 1(E\,\hat{f}(\mathbf{x}) \geq 1/2). \tag{11.12}$$

(In the case $p(\hat{f} \mid \mathbf{x})$ is asymmetric the median replaces $E\,\hat{f}(\mathbf{x})$.) The "bias" is defined to be

$$\text{bias}_B(\mathbf{x}) = 1[y_A(\mathbf{x}) \neq y_B(\mathbf{x})]r(\mathbf{x}) \tag{11.13}$$

and the "variance" as

$$\text{var}_B(\mathbf{x}) = 1[y_A(\mathbf{x}) = y_B(\mathbf{x})]r(\mathbf{x}). \tag{11.14}$$

Thus at a given point $\mathbf{x}$ the classifier has either bias or variance (but not both) depending upon whether or not the aggregated classifier (11.12) disagrees with the Bayes rule (2.4) there. By construction $r(\mathbf{x}) = \text{bias}_B(\mathbf{x}) + \text{var}_B(\mathbf{x})$ so that the decomposition

$$\Pr(\hat{y}(\mathbf{x}) \neq y) = \text{bias}_B(\mathbf{x}) + \text{var}_B(\mathbf{x}) + r_B(\mathbf{x}) \tag{11.15}$$

is produced.

In terms of the concepts developed in this paper one has

$$1[y_A(\mathbf{x}) \neq y_B(\mathbf{x})] = 1[b(f(\mathbf{x}), E\,\hat{f}(\mathbf{x})) \geq 0] \tag{11.16}$$

so that the reducible error (11.11) is called "bias" in regions of positive boundary bias (7.1), and "variance" in regions of negative boundary bias.

Tibshirani (1996) also defines "bias" and "variance" in terms of the aggregated classifier (11.12). From the point of view of this paper these definitions reduce to

$$\text{bias}_T(\mathbf{x}) = |2f(\mathbf{x}) - 1|\ \ 1[b(f(\mathbf{x}), E\,\hat{f}(\mathbf{x})) \geq 0] \tag{11.17}$$

and

$$\text{var}_T(\mathbf{x}) = |P_1(\mathbf{x}) - 1/2|\ \ (1 - |2P_1(\mathbf{x}) - 1|) \tag{11.18}$$

where $P_1(\mathbf{x})$ (11.1) is the probability that the classifier (2.6) predicts $\hat{y}(\mathbf{x}) = 1$ at $\mathbf{x}$. This definition of variance has a similar flavor to that of Kohavi and Wolpert (1996) (11.4); it has zero value when $P_1(\mathbf{x})$ assumes its extreme values (0, 1) and is non-negative over the entire range. However $\text{var}_T(\mathbf{x})$ (11.18) achieves its maximum value at $P_1(\mathbf{x}) = 1/4$ and $P_1(\mathbf{x}) = 3/4$ and has the value zero at $P_1(\mathbf{x}) = 1/2$ where $\text{var}_{\text{KW}}(\mathbf{x})$ (11.4) takes on its maximum value. Using the definitions (11.17), (11.18) does not lead to an additive decomposition of classification error in a form similar to that of (11.6), (11.9), or (11.15).

All of these additive decompositions are quite useful in providing insight into the nature of classification error. The bias definitions (11.10), (11.13), (11.16), and (11.17) all suggest (from different perspectives) the importance of the concept of boundary bias (7.1) developed in this paper. All emphasize the contribution of variability to the error rate of a classifier. This latter contribution especially (as noted by the authors) has often been overlooked in the development of machine learning procedures. To the extent that the development in this paper makes an additional contribution, it is that for classification error (unlike squared estimation error) the systematic and random components *interact* in a multiplicative and highly nonlinear way, and this interaction effect can sometimes be exploited to reduce error.

12. "Aggregated" classifiers

A principal motivation for proposing the additive decompositions discussed in Section 11 was to explain the apparent success of variance reduction techniques based on aggregation methods. From the perspective developed in this paper these methods can be viewed as obtaining an estimate of $E\,\hat{f}(\mathbf{x})$

$$\hat{f}_A(\mathbf{x}) = \hat{E}\,\hat{f}(\mathbf{x}) \tag{12.1}$$

and using it in place of $\hat{f}(\mathbf{x})$ (3.3) for function estimation (3.2) and classification (11.12). Examples of (12.1) are "bagging" (Breiman, 1995) which uses the "bootstrap smoothed" estimate of Efron and Tibshirani (1995) and "arcing" (Breiman, 1996) which includes other alternatives based on "boosting".

In the ideal limit

$$\hat{f}_A(\mathbf{x}) \to f_A(\mathbf{x}) \doteq E\,\hat{f}(\mathbf{x}) \tag{12.2}$$

this aggregation approach will reduce estimation error (5.4) since the bias (5.5) of $f_A(\mathbf{x})$ (12.2) is the same as that of $\hat{f}(\mathbf{x})$ but $\text{var}\, f_A(\mathbf{x}) = 0$. The degree of this reduction will depend on the relative importance of $\text{var}\,\hat{f}(\mathbf{x})$ (5.2) as compared to $\text{bias}^2\,\hat{f}(\mathbf{x})$ (5.5). For classification also, the (boundary) biases (7.1) are the same $b(f(\mathbf{x}),\ f_A(\mathbf{x})) = b(f(\mathbf{x}), E\,\hat{f}(\mathbf{x}))$, but in this case there is a multiplicative *interaction* effect with variance (6.5) and $\text{var}\, f_A(\mathbf{x}) = 0$. Therefore, using $f_A(\mathbf{x})$ in place of $\hat{f}(\mathbf{x})$ in (2.6) will produce zero boundary error (6.2), (6.5), and the minimal Bayes error rate (2.5), at $\mathbf{x}$ provided $b(f(\mathbf{x}), E\,\hat{f}(\mathbf{x})) < 0$. On the other hand, if $b(f(\mathbf{x}), E\,\hat{f}(\mathbf{x})) > 0$ this approach will produce certain boundary error $\Pr(y_A(\mathbf{x}) \neq y_B(\mathbf{x})) = 1$ and *increased* error rate over using $\hat{f}(\mathbf{x})$. As noted by Breiman (1996) and observed by Tibshirani (1996), aggregation can make a good classifier better

but can make a bad classifier worse. Clearly, this effect occurs separately at each individual prediction point $\mathbf{x}$, so success of aggregation for classification depends on the relative size of the portion of the input space with negative boundary bias.

As discussed in Section 10 the success of variance reduction techniques for classification can be enhanced by the use of a bias adjustment (10.1) to $\hat{f}(\mathbf{x})$. This is a consequence of the (boundary) bias-variance multiplicative interaction effect at each $\mathbf{x}$. For the same reason it seems likely that such an adjustment

$$\tilde{f}_A(\mathbf{x}) = \hat{f}_A(\mathbf{x}) + t \tag{12.3}$$

will be beneficial in the context of aggregated classification as well. Bias adjustment (10.1), (12.3) can (sometimes dramatically) reduce the proportion of the input space with positive boundary bias. As with other methods of variance reduction a good adjustment value t is not likely to be known in any particular situation, and therefore it will have to be estimated through some model selection technique such as cross-validation.

13. Limitations and future work

The most serious limitation of the work presented here is the restriction to the two-class problem. Intuition suggests that many of the concepts developed in this context may have analogs in the $L \geq 3$ class case, but the detailed development will be more complicated. In particular, there will likely be analogs to the notion of boundary bias and its interaction with the variances of the estimates of the L target probability functions $\{f_l(\mathbf{x})\}_1^L$. Also the concept of bias adjustment(s) may also be helpful in the multi-class problem. This is left for future work.

Another limitation is the use of the Gaussian approximation (6.4). This is clearly not crucial to the qualitative results obtained. For example, the distribution of K-nearest neighbor estimates is not strictly Gaussian but, as seen in Section 9, its behavior closely follows that suggested by (6.5). As noted, the median of $p(\hat{f} \mid \mathbf{x})$ should replace the mean $E\,\hat{f}(\mathbf{x})$ in the definition of boundary bias (7.1) in the case of asymmetry, and an appropriate measure of its spread (variability of $\hat{f}(\mathbf{x})$) would substitute for $\sqrt{\text{var}\,\hat{f}(\mathbf{x})}$ in deriving a boundary error analog to (6.5). Clearly, these two quantities would strongly interact in whatever detailed form emerged from the derivation.

The illustrative examples presented were intensionally chosen to be quite simple so that one could easily understand the geometry of the decision boundaries, and thus the nature of the boundary bias (7.1) associated with the classification methods studied here. Actual decision boundaries for specific problems encountered in practice may of course be quite different, as could the nature of the boundary bias associated with other classification methods. Thus the gains associated with bias adjustment (10.1) may not be the same in other situations. All of this is problem dependent and can only be determined through experimentation in each specific case.

The goal of the work presented here is to illustrate that classification error responds to error in the target probability estimates in a much different (and perhaps less intuitive) way than squared estimation error. This helps explain why improvements to the latter do

not necessarily lead to improved classification performance, and why simple methods such as naive Bayes, K-nearest neighbors, and others remain competitive, even though they usually provide very poor estimates of the true underlying probabilities. Good probability estimates are not necessary for good classification; similarly, low classification error does not imply that the corresponding class probabilities are being estimated (even remotely) accurately. An understanding of these issues may help improve the chance of success of future methodological developments.

Acknowledgments

Enlightening discussions with Trevor Hastie, Art Owen, and David Rosen are gratefully acknowledged. Work supported in part by the Department of Energy under contract number DE-AC03-76SF00515 and by the National Science Foundation under grant number DMS-9403804.

References

Bellman, R.E. 1961. Adaptive Control Processes. Princeton University Press.

Breiman, L. 1995. Bagging predictors. Dept. of Statistics, University of California, Berkeley, Technical Report.

Breiman, L. 1996. Bias, variance, and arcing classifiers. Dept. of Statistics, University of California, Technical Report (revised).

Breiman, L., Friedman, J.H., Olshen, R.A., and Stone, C.J. 1984. Classification and Regression Trees. Wadsworth.

Chow, W.S. and Chen, Y.C. 1992. A new fast algorithm for effective training of neural classifiers. Pattern Recognition, 25:423–429.

Dietterich, T.G. and Kong, E.B. 1995. Machine learning bias, statistical bias, and statistical variance of decision tree algorithms. Dept. of Computer Science, Oregon State University Technical Report.

Efron, B. and Tibshirani, R. 1995. Cross-validation and the bootstrap: Estimating the error rate of a prediction rule. Dept. of Statistics, Stanford University Technical Report.

Fix, E. and Hodges, J.L. 1951. Discriminatory analysis—nonparametric discrimination: Consistency properties. Randolf Field Texas: U.S. Airforce School of Aviation Medicine Technical Report No. 4.

Friedman, J.H. 1985. Classification and multiple response regression through projection pursuit. Dept. of Statistics, Stanford University Technical Report LCS012.

Geman, S., Bienenstock, E., and Doursat, R. 1992. Neural networks and the bias/variance dilemma. Neural Comp., 4:1–48.

Good, I.J. 1965. The Estimation of Probabilities: An Essay on Modern Bayesian Methods. M.I.T. Press.

Hand, D.J. 1982. Kernel discriminant analysis. Chichester: Research Studies Press.

Heckerman, D., Geiger, D., and Chickering, D. 1994. Learning Bayesian networks: the combination of knowledge and statistical data. In Proceedings of the Tenth Conference on Uncertainty in Artificial Intelligence, pp. 293–301, AAAI Press and M.I.T. Press.

Henley, W.E. and Hand, D.J. 1996. A k-nearest neighbour classifier for assessing consumer credit risk. The Statistician, 45:77–95.

Holte, R.C. 1993. Very simple classification rules perform well on most commonly used data sets. Machine Learning, 11:63–90.

Kohavi, R. and Wolpert, D.H. 1996. Bias plus variance decomposition for zero-one loss functions. Dept. of Computer Science, Stanford University Technical Report.

Kohonen, T. 1990. The self-organizing map. Proceedings of the IEEE, 78:1464–1480.

Langley, P., Iba, W., and Thompson, K. 1992. An analysis of Bayesian classifiers. In Proceedings of the Tenth National Conference on Artificial Intelligence, pp. 223–228, AAAI Press and M.I.T. Press.

Lippmann, R. 1989. Pattern classification using neural networks. IEEE Communications Magazine, 11:47–64.

McLachlan, G.J. 1992. Discriminant Analysis and Statistical Pattern Recognition. Wiley.
Quinlan, J.R. 1993. C4.5: Programs for Machine Learning. Morgan Kaufmann.
Rosen, D.B., Burke, H.B., and Goodman, P.H. 1995. Local learning methods in high dimension: Beating the bias-variance dilemma via recalibration. Workshop Machines That Learn—Neural Networks for Computing, Snowbird Utah.
Tibshirani, R. 1996. Bias, variance and prediction error for classification rules. Dept. of Statistics, University of Toronto Technical Report.
Titterington, D.M., Murray, G.D., Murray, L.S., Spiegelhalter, D.J., Skene, A.M., Habbema, J.D.F., and Gelpke, G.J. 1981. Comparison of discrimination techniques applied to a complex data set of head injured patients. J. Roy. Statist. Soc. A, 144:145–175.

Jerome H. Friedman received the Ph.D. degree in Physics from the University of California, Berkeley. He is presently Professor of Statistics at Stanford University. He has been engaged in research on techniques for the analysis of large complex data sets for over 20 years.