On Optimum Choice of k in Nearest Neighbor Classification

Anil K. Ghosh

Theoretical Statistics and Mathematics Unit
Indian Statistical Institute
203, B. T. Road, Kolkata 700108, India.
E-mail : anilkghosh@rediffmail.com

Abstract

A major issue in k-nearest neighbor classification is how to choose the optimum value of the neighborhood parameter k. Popular cross-validation techniques often fail to guide us well in selecting k mainly due to the presence of multiple minimizers of the estimated misclassification rate. This article investigates a Bayesian method in this connection, which solves the problem of multiple optimizers. Utility of the proposed method is illustrated using some benchmark data sets.

Keywords : Accuracy index, Bayesian strength function, cross-validation, misclassification rate, neighborhood parameter, non-informative prior, optimal Bayes risk, posterior probability.

1 Introduction

Nearest neighbor classification (see e.g., Fix and Hodges, 1951; Cover and Hart, 1968; Dasarathy, 1991) is one of the simplest and popular methods for statistical pattern recognition. Like other classifiers, it forms a finite partition $\mathcal{X}_1, \mathcal{X}_2, \ldots, \mathcal{X}_J$ of the sample space $\mathcal{X}$ such that an observation $\mathbf{x}$ is classified into j-th class if $\mathbf{x} \in \mathcal{X}_j$. When the density functions f_j and the prior probabilities π_j of J competing classes are known, the optimal Bayes rule (see e.g., Anderson, 1984) is given by $d(\mathbf{x}) = \arg\max_j \pi_j f_j(\mathbf{x})$. However, in practice, they are usually unknown, and one needs to estimate them using the available data. A natural estimator for π_j is $\hat{\pi}_j = n_j/n$, where n_j is the number of class-j observations in the training sample and $n = \sum_{j=1}^{J} n_j$. Estimates of f_j can be obtained using some parametric (see e.g., Anderson, 1984; McLachlan, 1992) or nonparametric methods (see e.g., Ripley, 1996; Duda, Hart and Stork, 2000; Hastie, Tibshirani and Friedman, 2001). Nearest neighbor density estimation (see e.g., Loftsgaarden and Quesenberry, 1965) is one such nonparametric method for estimating population densities. To estimate f_j at a specific data point $\mathbf{x}$, it assumes f_j to be constant over a neighborhood around $\mathbf{x}$. If v is the volume of the neighborhood and $n_j^{(v)}$ out of n_j observations fall in that region, the nearest neighbor density estimate at $\mathbf{x}$ is given by $\hat{f}_j^{(v)}(\mathbf{x}) = n_j^{(v)}/n_j v$. If the same neighborhood is used for estimation of all population densities, plugging these estimates in the Bayes rule, one gets the usual nearest neighbor classifier, which essentially classifies an observation to the class having the maximum number of representatives in that neighborhood. Usually, a closed ball of radius $r_k(\mathbf{x})$ is taken as this neighborhood, where $r_k(\mathbf{x})$ is the distance between $\mathbf{x}$ and its k-th nearest data point in the training sample, and the resulting classification rule is known as the k-nearest neighbor rule.

Performance of a nearest neighbor classifier depends on the distance function and the value of k as well. Euclidean metric is the most popular choice for this distance function. Of course, if the measurement variables are not of comparable units and scales, it is more meaningful to standardize

the variables before using the Euclidean distance for classification. If an estimate of the pooled dispersion matrix is used for standardization, it essentially leads to classification using Mahalanobis distances (see e.g., Mahalanobis, 1936). However, many other flexible or adaptive metric (see e.g., Friedman, 1994; Hastie and Tibshirani, 1996) can be used as well.

The neighborhood parameter k, which controls the volume of the neighborhood and consequently the smoothness of the posterior estimates, plays an important role on the performance of a nearest neighbor classifier. Existing theoretical results (see e.g., Loftsgaarden and Quesenberry, 1965; Cover and Hart, 1968) suggest that if Euclidean distance is used for classification, one should vary k with n in such a way that $k \to \infty$ and $k/n \to 0$ as $n \to \infty$. The same assertion holds also for Mahalanobis distance if one uses any consistent estimate of the pooled dispersion matrix for standardization. However, for small or moderately large sample sizes, there is no theoretical guideline for choosing the optimum value of k. This optimum value depends on the specific data set and it is to be estimated using the available training sample observations. In this context, one can follow the idea likelihood cross validation (LCV, see e.g., Silverman, 1986) used in kernel methods. It estimates the optimum value of k by maximizing the loglikelihood score $\mathcal{L}(k) = \sum_{t=1}^{n} log\{p_{-t}^{(k)}(c_t \mid \mathbf{x}_t)\}$, where c_t is the class label of the observation $\mathbf{x}_t$ and $p_{-t}^{(k)}(j \mid \mathbf{x}_t)$ is the posterior probability estimate for the j-th class at $\mathbf{x}_t$ (i.e. the proportion of class-j observations among the k nearest neighbors of $\mathbf{x}_t$), when $\mathbf{x}_t$ is not used as a data point. Holmes and Adams (2002, 2003) used a slightly different version this likelihood criterion for aggregating the results of different nearest neighbor classifiers.

In practice, one uses cross-validation methods (see e.g., Lachenbruch and Mickey, 1968, Stone, 1977) to estimate the misclassification rate for different values of k and chooses that one which leads to the lowest estimate of error rate. However, these cross-validation techniques use naive empirical proportions for estimating the misclassification probabilities. As a consequence, often we get two or more values of k as minimizers of estimated misclassification rate, from which it is difficult to choose the optimum one. Ghosh and Chaudhuri (2004) discussed this problem of cross-validation methods in the context of kernel discriminant analysis, where they proposed a smooth estimate for the misclassification probability function for finding the optimal bandwidth parameter. This article proposes one such estimate for misclassification rate of a nearest neighbor classifier and thereby gives a rule for selecting the optimum value of k for a specific data set to be classified.

2 Description of the methodology

In usual nearest neighbor classification, one assumes f_j's and hence the posterior probabilities $[p(j \mid \mathbf{x}) = \pi_j f_j(\mathbf{x}) / \sum_{r=1}^{J} \pi_r f_r(\mathbf{x})]$ to be fixed and non-random over a neighborhood around $\mathbf{x}$. The method we propose in this article differs from this usual notion of nearest neighbor classification. Here, instead of assuming the posterior probabilities to be constant over that neighborhood, we assume a prior distribution there. Since it is quite evident that the calculations have to done at each $\mathbf{x}$ separately, for convenience of our notation we shall denote $p(j \mid \mathbf{x})$ by p_j, and the vector of conditional probabilities $(p_1, p_2, \ldots, p_J)$ $[\sum_{j=1}^{J} p_j = 1]$ will be denoted by $\mathbf{p}$. Suppose that for some given k, $\xi_k(\mathbf{p})$ is the prior distribution of $\mathbf{p}$ in the neighborhood around $\mathbf{x}$, which is a ball of radius $r_k(\mathbf{x})$. If t_{j_k} of these k neighbors come from the j^{th} class, the conditional distribution of $\mathbf{t}_k = (t_{1_k}, t_{2_k}, \ldots, t_{J_k})$ $[\sum_{j=1}^{J} t_{j_k} = k]$

for given $\mathbf{p}$ and k can be expressed as

$$\psi(\mathbf{t}_k \mid \mathbf{p}, k) = \binom{k}{t_{1_k}, t_{2_k}, \ldots, t_{J_k}} \prod_{j=1}^{J} p_j^{t_{j_k}}.$$

Therefore, for some fixed k and $\mathbf{t}_k$, the conditional distribution of $\mathbf{p}$ is given by

$$\zeta(\mathbf{p} \mid k, \mathbf{t}_k) = \xi_k(\mathbf{p})\ \psi(\mathbf{t}_k \mid \mathbf{p}, k) \Big/ \int \xi_k(\mathbf{p})\ \psi(\mathbf{t}_k \mid \mathbf{p}, k)\ d\mathbf{p}.$$

Following the idea and terminology of Ghosh, Chaudhuri and Murthy (2005), we use this conditional distribution to define the Bayesian measure of strength for different populations, where the strength function for the j-th class is given by

$$S(j \mid k) \ = P\{\arg\max_r p_r = j \mid k, \mathbf{t}_k\} \ = \int\limits_{p_j = max\ \{p_1, p_2, \ldots, p_J\}} \zeta(\mathbf{p} \mid k, \mathbf{t}_k)\ d\mathbf{p}.$$

It is quite transparent from the definition that $0 \leq S(j \mid k) \leq 1$ and $\sum_{j=1}^{J} S(j \mid k) = 1$ if $\xi_k(\mathbf{p})$ is the probability density function of a continuous distribution. The value of the strength function depends on the prior distribution $\xi_k(\mathbf{p})$ as well, and one has to choose it appropriately. Uniform prior is the easiest one to handle with. Not only it makes the computations simpler, it is non-informative and gives no preference to any of the classes. An interesting property of the Bayesian strength function is given in the following theorem.

Theorem 2.1 : *If $p_1, p_2, \ldots, p_J$ are interchangeable in $\xi_k(\mathbf{p})$, $S(j \mid k) > S(i \mid k)$ iff $t_{jk} > t_{ik}$. Further, if $\xi_k(\mathbf{p})$ is uniform, given the value of other t_{rk}'s ($r \neq i, j$), $S(j \mid k) - S(i \mid k)$ is monotonically increasing in $t_{jk} - t_{ik}$.*

Note that t_{jk}/k is nothing but the usual k-nearest neighbor estimate for the posterior probability of the j-th class. If an observation originally comes from the j-th class, one should expect to have higher values of t_{jk}, and the above theorem tells that the value $S(j \mid \mathbf{x})$ should also be high in that case. Therefore, a good classifier is expected to produce higher values of strength function in favor of the true class, or in other words, the average strength of the true classes can be viewed as a measure of the accuracy of a classifier. We call this measure as the accuracy index (α), and this is given by

$$\alpha(k) = \sum_{j=1}^{J} \pi_j \int S(j \mid k)\ f_j(\mathbf{x})\ d\mathbf{x}.$$

The value of k that leads to the highest value of α can be considered as the optimum value of the neighborhood parameter.

The quantity $1 - \alpha(k)$ can be viewed as an alternative measure for misclassification probability of a k-nearest neighbor classifier. From the existing results (see e.g., Loftsgaarden and Quesenberry, 1965, Cover and Hart, 1968), we know that in nearest neighbor classification, if $k \to \infty$ and $k/n \to 0$ as $n \to \infty$, estimated posterior probabilities tend to true posteriors. The following theorem on accuracy index shows that under the same condition, $1 - \alpha(k)$ converges to the optimal Bayes risk.

Theorem 2.2 : *Suppose that $\{k_n : n \geq 1\}$ is a sequence of positive integers such that as $n \to \infty$, k_n tends to infinity and k_n/n tends to 0. Further assume that the population densities f_j are*

continuous and for all $n \geq 1$, $p_1, p_2, \ldots, p_J$ *are interchangeable in* $\xi_{k_n}(\mathbf{p})$. *Then as* $n \to \infty$, $1 - \alpha(k_n)$ *converges (in probability) to the optimal Bayes risk.*

For data analytic purpose, we approximate α by $\hat{\alpha}(k) = \sum_{t=1}^{n} S(c_t \mid \mathbf{x}_t)/n$ and select k_0 as the optimum neighborhood parameter if $\hat{\alpha}(k_0) > \hat{\alpha}(k)\ \forall k \neq k_0$. Note that for computing $S(c_t \mid \mathbf{x}_t)$ we adopt the leave-one-out strategy and do not consider $\mathbf{x}_t$ as a training data point. The value of k_0 depends on the choice of prior distributions $\xi_1(\mathbf{p}), \xi_2(\mathbf{p}), \ldots, \xi_{n-1}(\mathbf{p})$ as well. Since, these priors are used to estimate the accuracy indices and the optimum value of k, instead of using different priors for different k, it is more meaningful to use the same prior distribution for all values of neighborhood parameter. Throughout this article, we have used uniform prior distribution for this purpose.

It should be noted that usual cross-validation techniques only count the number of correct classifications and misclassifications to select the optimal neighborhood parameter. As a result, often the value of k_0 obtained from cross-validation is not unique, and these different choices of k_0 may lead to significantly different error rates for classifying future observations. But our proposed criterion gives emphasis not only correct classification but also on the strength of evidences for that classification. This makes $\hat{\alpha}(k)$ in some sense more smooth in nature and thereby reduces the problem of multiple optimizers.

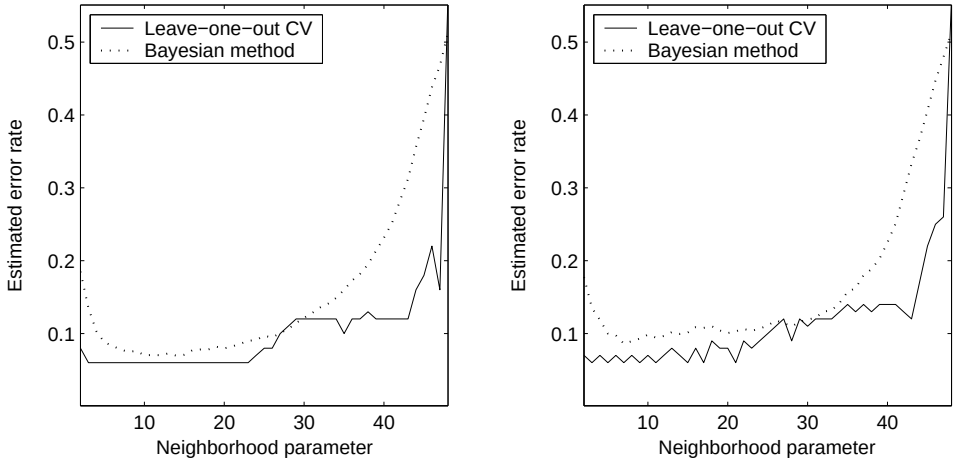

Figure 2.1 : Estimated misclassification rates for different values of k

We conclude this section with an illustrative example on salmon data set taken from Johnson and Wichern (1992). It contains information on growth ring diameters (freshwater and marine water) of 100 salmon fish coming from Alaskan or Canadian water. We divided this data set randomly into two equal halves and tried to find out the best value of k for each of these subsets. Both leave-one-out misclassification rates and the corresponding Bayesian estimates $1 - \hat{\alpha}(k)$ are plotted in Figure 2.1 for these two sets and for different values of k. In both these cases, usual leave-one-out cross-validation algorithm led to multiple minimizers for the estimated error rate. Clearly, it is very difficult to choose a single optimum value of the neighborhood parameter from this set of optimizers. On the other hand, the Bayesian method led to a unique k_0 in both the cases and thereby solves the problem of multiple optimizers that one faces in usual cross-validation methods. This example clearly demonstrates the utility of the Bayesian method in selecting the optimum value of the neighborhood parameter. In terms of misclassification rates, a comparison between these two methods is given in the next section.

3 Case studies

In this section, we use some benchmark data sets to compare the performance of the proposed Bayesian method. Some of these data sets have separate training and test samples. For those data sets, we report the test set misclassification rates for leave-one-out cross-validation (henceforth we will refer to it as CV-class) and the Bayesian method. Misclassification rates for likelihood cross-validation (LCV) method (described earlier in Section 1) are also reported to facilitate the comparison. In the case of other data sets, we formed the training and test samples by randomly partitioning the data. This random partition was carried out 250 times to generate 250 training and test samples. Average test set misclassification rates for different methods over those 250 trials are given in Tables 3.1 and 3.2 along with their corresponding standard errors. We have already demonstrated in Section 2 that the CV-class technique often leads to a situation when we have multiple minimizers for the estimated misclassification rate. Since usual nearest neighbor classifiers assume the posterior probability of a specific class to be constant over the entire neighborhood, it is somewhat reasonable to consider the smallest value of k in such cases. Throughout this section sample proportions for different classes are used as their prior probabilities.

3.1 Classification between two competing classes

We begin with some examples on two-class classification problems. Salmon data is one such example which was described earlier in Section 2. Here, along with that salmon data, we use four other data sets (synthetic data, biomedical data, Australian credit data and Pima Indian diabetes data) for illustration. For salmon data and synthetic data, where the measurement variables are of same unit and scale, we report the results based on both Euclidean and Mahalanobis distances. For all other data sets, only Mahalanobis distances were used for classification. Unlike the synthetic data, the other four data sets do not have separate training and test samples. We formed those sets by randomly partitioning the data. For salmon data and biomedical data, we used 50 and 100 observations, respectively, to form the training sample, while in each of other two cases 300 observations were used for this purpose. In each case, the rest of the observations were used to form the corresponding test set. In the case of biomedical data, we removed 15 out 209 observations, which have missing values, and carried out our analysis with remaining 194 observations. All the data sets that we consider in this section and in Section 3.2 and their descriptions are available at either at UCI Machine Learning Repository (http://www.ics.uci.edu) or at CMU Data Archive (http://lib.stat.cmu.edu).

Table 3.1 : Average misclassification rates for two-class problems and their corresponding standard errors.

	Synthetic		Salmon		Biomedical	Australian	Pima Indian
	Euclid.	Mahal.	Euclid.	Mahal.	Mahal.	Mahal.	Mahal.
LCV	9.20	10.00	9.43 (0.20)	8.30 (0.19)	18.48 (0.22)	14.28 (0.08)	26.11 (0.10)
CV-class	8.70	11.70	9.95 (0.22)	8.82 (0.20)	17.66 (0.20)	14.06 (0.09)	25.96 (0.09)
Bayesian	8.50	9.80	9.04 (0.18)	8.29 (0.19)	17.63 (0.20)	13.89 (0.09)	25.80 (0.09)

In all these data sets, the Bayesian method led to better average misclassification rates than LCV and CV-class techniques (see Table 3.1). One should note that when the true misclassification

rates for different k are quite close, due to stepwise nature of estimated error rate, usual cross-validation method often fails to figure out those small differences and leads to multiple minimizers of estimated error rate. Even in some cases, especially when the training sample is small, it may lead to the same estimated error rate for different values of k, when the difference between their true error rates is not insignificant. As a result, in such cases, this usual cross-validation method may choose some undesired value of k, which results in high misclassification rate in the test sample. Paik and Yang (2004) suggested to use cross-validation methods only when it easy to choose a single optimum k from the cross-validated error rates of different nearest neighbor classifiers. Here also we observed a similar phenomenon. On biomedical, Pima Indian and Australian data, in most of the cases, CV-class method could lead to a single optimum value of k. Out of 250 simulations, multiple minimizers were obtained in 13%, 5% and 28% cases, respectively, and the average cardinality of the optimizing sets was found to be 1.20, 1.05 and 1.71 in the respective cases. In these data sets, we could not achieve significant gain by using the Bayesian technique. As compared to their standard errors, there was no significant difference between the error rates of the CV-class and the Bayesian method. However, the error rates of the Bayesian method were significantly better than those of LCV in all these three data sets. On biomedical data, CV-class also performed significantly better than LCV. In the case of salmon data, in more that 65% of the cases (67.6% in case of Euclidean distance and 69.2% in case of Mahalanobis distance) CV-class failed to select a single optimum value of k, and the average cardinality of the optimizing set was also very high (6.75 and 5.79 for Euclidean and Mahalanobis distance, respectively). Clearly, selection of optimum k by cross-validation is more difficult in that case. On this data set, both LCV and the Bayesian method led to significantly better performance than CV-class technique.

Though we have used uniform distribution as the common prior distribution for all k to estimate k_0 and the corresponding misclassification rate, the final result is not sensitive to the choice of the prior distribution as long as it is symmetric in its arguments. Table 3.2 below shows the average test set misclassification rates of the Bayesian method for different choices of symmetric priors along with their corresponding standard error. The reported results clearly show that the differences between the error rates for any two different priors are statistically insignificant in all the data sets that we have analyzed here.

Table 3.2 : Average misclassification rates of the Bayesian method and their standard errors for different choices of prior.

Prior	Synthetic		Salmon		Biomedical	Australian	Pima Indian
distribution	Euclid.	Mahal.	Euclid.	Mahal.	Mahal.	Mahal.	Mahal.
Uniform	8.50	9.80	9.04 (0.18)	8.29 (0.19)	17.63 (0.20)	13.89 (0.09)	25.80 (0.09)
Beta(0.5,0.5)	8.50	9.80	9.20 (0.19)	8.36 (0.18)	17.52 (0.20)	13.91 (0.09)	25.77 (0.09)
Beta(2,2)	8.50	9.70	8.98 (0.18)	8.17 (0.19)	17.65 (0.20)	13.89 (0.09)	25.83 (0.09)
Beta(3,3)	8.50	9.70	8.90 (0.18)	8.04 (0.19)	17.62 (0.20)	13.90 (0.09)	25.83 (0.09)
Triangular	8.50	9.70	8.93 (0.18)	8.06 (0.19)	17.70 (0.20)	13.86 (0.08)	25.84 (0.09)
Normal*	8.50	9.80	9.00 (0.18)	8.32 (0.19)	17.64 (0.20)	13.89 (0.08)	25.81 (0.09)
Cauchy*	8.50	9.80	9.00 (0.18)	8.30 (0.19)	17.63 (0.20)	13.89 (0.09)	25.80 (0.09)
Double Exp.*	8.50	9.80	9.01 (0.18)	8.30 (0.19)	17.64 (0.20)	13.88 (0.09)	25.80 (0.09)

* Location parameter 0.5, scale parameter 1.0, truncated at [0,1].

3.2 Classification with more than two populations

For further illustration of our proposed method, we consider six other data sets each having observations from more than two populations. Out of these six data sets, vowel data has separate training and test samples. For this data set, we reported the test set error rates for different methods. In all other cases, training and test samples were formed by randomly partitioning the data as before. The sizes of the training and the test samples in each partition are reported in Table 3.3. Average test set misclassification rates (based on 250 partitions) for different methods and their standard are also reported in that table. For vowel data and Iris data, where the measurement variables are of same unit and scale, both Euclidean and Mahalanobis distances were used for classification. For all other data sets, we used only Mahalanobis distances.

For computing the Bayesian strength of a population, one can approximate the integrals appearing in its expression (see Section 2) using any numerical integration method based on an appropriate averaging of the integrand over a suitable grid in the domain of integration. Given the value of k and the vector $\mathbf{t}_k$, the required number of computations for this approximation is proportional to γ^{J-1}, where γ is the number of grid points chosen on each axis and J is number of competing classes. Clearly, this computational cost grows up rapidly with J, and in the presence of several competing populations, it becomes computationally difficult to use this numerical integration method. In such cases one can resort to some other approximation. If $\xi_k(\mathbf{p})$ is the pdf of a uniform distribution in [0,1], it is easy to see that given the value of k and $\mathbf{t}_k$, $\mathbf{p}$ follows a Dirichlet distribution. Therefore, instead of using any Markov Chain Monte Carlo type algorithm one can easily generate observations from appropriate Dirichlet distribution for approximating the Bayesian strengths of different populations. For our data analytic purpose, we adopted the numerical integration method when $J \leq 3$. In all other cases, strengths functions were approximated using 10,000 observations generated from appropriate Dirichlet distribution.

Another useful way of reducing this computational cost is to adopt a pairwise classification method, where we split a J-class problem into $\binom{J}{2}$ binary classifications taking a pair of classes at a time. Results of these pairwise classifications can be combined by the method of majority voting (see e.g., Friedman, 1996) or by the method of pairwise coupling (see e.g., Hastie and Tibshirani, 1998). Apart from reducing the computational cost, this pairwise approach allows the flexibility of using different k for different binary classifications. In practice, if not better, this pairwise method produces competitive performance in most of the cases. Misclassification rates for this pairwise classification method are also reported in Table 3.3 below, when voting was used to combine the results of pairwise classifications. Error rates for pairwise versions of LCV and CV-class are reported as well.

In most of these data sets, the Bayesian method performed better than its competitors though the differences in the error rates were not always statistically significant. On vowel data and vehicle data, LCV, CV-class and Bayesian, all three three methods achieved competitive misclassification rates. In almost all other cases, the Bayesian method had an edge over LCV and CV-class techniques. In the case of vehicle and chemical and overt diabetes data (which is referred to as diabetes data in Table 3.3), pairwise classification method could led to significant improvement in misclassification rates. Their performance in all other cases was also fairly competitive.

Table 3.3 : Average misclassification rates and standard errors
for classification problems with more than two classes.

	Sample size		Combined			Pairwise		
	Train	Test	LCV	CV-class	Bayesian	LCV	CV-class	Bayesian
Vowel (Euclid.)	528	462	42.64	43.72	41.13	45.02	43.29	42.21
Vowel (Mahal.)			46.75	46.75	47.62	43.07	43.94	44.59
Iris (Euclid.)	75	75	3.91 (0.13)	4.25 (0.12)	3.90 (0.14)	3.87 (0.12)	4.26 (0.13)	3.89 (0.13)
Iris (Mahal.)			2.89 (0.11)	3.07 (0.12)	2.53 (0.10)	3.10 (0.10)	3.17 (0.11)	2.98 (0.11)
Diabetes (Mahal.)	100	45	14.80 (0.47)	9.92 (0.24)	9.96 (0.25)	8.80(0.24)	9.18 (0.254)	8.67 (0.23)
Wine (Mahal.)	100	78	2.29 (0.10)	2.26 (0.10)	2.07 (0.09)	2.75 (0.12)	2.73 (0.10)	2.40 (0.11)
Crab (Mahal.)	100	100	6.55 (0.13)	6.67 (0.14)	6.30 (0.13)	7.40 (0.13)	7.18 (0.15)	6.86 (0.14)
Vehicle (Mahal.)	400	446	21.44 (0.10)	21.40 (0.11)	21.51 (0.10)	19.26 (0.09)	19.10 (0.11)	19.29 (0.10)

In view of the above data analysis, the proposed Bayesian method seems to be better than usual cross-validation or likelihood cross-validation technique for selecting the optimal neighborhood parameter. It provides an alternative estimate for misclassification rates, which is in some sense more smooth in nature, and thereby resolves the problem of multiple minimizers that one faces in usual cross-validation approach. In almost all data sets considered in this article, the Bayesian method led to better performance that LCV and CV-class techniques. In many of these cases, the differences between their misclassification rates found to be statistically significant as well.

3.3 Computational complexity

In this section, we compare the computational complexity of our proposed method with that of usual leave-one-out cross-validation technique as the number of training data points becomes large. Note that the dimension d is involved only in distance computation, and that is required for both these methods. Therefore, it is not at all important for our comparison, and troughout this section we do not consider the dimension in our calculation. Instead, we start from the stage when all pairwise distances are given to us.

Classification of a specific training data point based on remaining $N-1$ training sample observations requires sorting of $N-1$ distances, if we want to classify it using all possible values of k. This sorting of $N-1$ elements needs $O(NlogN)$ calculations. Given the sorted array of distances, one can find out the values of $t_{1k}, t_{2k}, \ldots, t_{Jk}$ for $k = 1, 2, \ldots, N-1$ in $O(N)$ calculations, and this gives the classification results for that observation for all possible values of k. Now, for our Bayesian method, we need to compute the Bayesian strength functions for different classes as well, and it needs some additional calculations. If we use the numerical integration method, given the value of k and $\mathbf{t}_k$, in a J-class problem, the number of calculations for strength computation is proportional to γ^{J-1}, where γ is the number of grid points chosen on each axis. For the simulation method that we use for $J > 3$ (see Section 3.2), this number is proportional to the number of observations generated from the Dirichlet distribution.

Clearly, all these operations have to be repeated N times (taking one data point at a time) to find the leave-one-out error rates and the average Bayesian strengths for different k. Based on these estimated error rates or estimated average Bayesian strengths, one selects the optimum value of k for a specific data set. Though the number of computations is higher in the latter case, from the above discussion it is quite clear that the asymptotic order of computations is the same $O(N^2 logN)$

for both these methods. However, for faster implementation of the Bayesian technique, sometimes it may be useful to run the usual cross-validation technique for choosing a desired range of values for k and then the Bayesian method to find k_0 in that range. It should be noted that after finding the optimum value of k, one needs only $O(N)$ calculations (see the algorithm for finding order statistics in Aho, Hopcroft and Ullman, 1974) to classify a future observation.

Recent advances in data mining demands a classification method to be fast. Therefore, in the presence of large data set, instead of using leave-one-out method, one may like to use V-fold cross-validation technique with some small value of V (say V=5 or 10). Though this V-fold cross validation technique needs the same order of calculation as that required by leave-one-out method, the former requires lesser computations both for estimating the error rate and the average Bayesian strength. Hence it is computationally preferable when we have large training sets. The pairwise classification method discussed in Section 3.2 also reduces the number of calculations without changing its order.

Acknowledgement

I am thankful to Prof. Probal Chaudhuri of Indian Statistical Institute, Kolkata for useful academic discussions and suggestions. I would also like to thank an associate editor and two referees for their careful reading of an earlier version of the paper and for providing me with several helpful comments.

Appendix

Proof of Theorem 2.1 : The first part of the theorem follows from Theorem 2.1 of Ghosh, Chaudhuri and Murthy (2005). For the proof of the other part, first note that if $\xi_k(\mathbf{p})$ is uniform, given the value of k and $\mathbf{t}_k$, the conditional distribution of $\mathbf{p}$ is Dirichlet with parameters $k, t_{1k}+1, t_{2k}+1, \ldots, t_{Jk}+1 \ \ (\sum_{j=1}^{J} t_{jk} = k)$. Now consider the random variables $X_1, X_2, \ldots, X_J$, which are independent and $X_j \sim Gamma(t_{jk}+1)$ for $j = 1, 2, \ldots, J$. If we define $S = \sum_{j=1}^{J} X_j$, it is quite well-known that $(X_1/S, X_2/S, \ldots, X_J/S)$ jointly follows a Dirichlet distribution with parameters $k, t_{1k}+1, t_{2k}+1, \ldots, t_{Jk}+1 \ \ (\sum_{j=1}^{J} t_{jk} = k)$. Therefore, $P\{p_j > p_i \ \forall i \neq j \mid k, \mathbf{t}_k\} = P\{X_j > X_i \ \forall i \neq j\}$. Now, it is easy to verify the result on monotonicity from the fact that X_j is a sum of $t_{jk}+1$ independent and identically distributed exponential variables with unit mean.

Proof of Theorem 2.2 : Suppose that for some given $\mathbf{x}$, $\pi_j f_j(\mathbf{x}) > \pi_i f_i(\mathbf{x})$ for all $i \neq j$. Now, under the given condition, $S(j \mid k_n) \xrightarrow{P} 1$ and $S(i \mid k_n) \xrightarrow{P} 0$ for all $i \neq j$ as $n \to \infty$ (see Theorem 2.2 of Ghosh, Chaudhuri and Murthy, 2005). Therefore, as $n \to \infty$, $S(j \mid k_n) \xrightarrow{P} 1$ if and only if $\pi_j f_j(\mathbf{x}) > \pi_i f_i(\mathbf{x})$ for all $i \neq j$, otherwise it converges to 0 in probability. Now, the proof of the theorem follows from the definition of $\alpha(k)$ using Dominated Convergence Theorem.

References

[1] Aho, A. V., Hopcroft, J. E. and Ullman, J. D. (1974) *Design and analysis of computer algorithms.* Addition-Wesley, London.

[2] Anderson, T. W. (1984) *An Introduction to Multivariate Statistical Analysis.* Wiley, New York.

[3] Cover, T. M. and Hart, P. E. (1968) Nearest neighbor pattern classification. *IEEE Trans. Info. Theory*, **13**, 21-27.

[4] Dasarathy, B. V. ed. (1991) *Nearest Neighbor (NN) Norms : NN Pattern Classification Techniques.* IEEE Computer Society, Washington.

[5] Duda, R., Hart, P. and Stork, D. G. (2000) *Pattern Classification.* Wiley, New York.

[6] Fix, E. and Hodges, J. L. (1951) Discriminatory analysis - nonparametric discrimination : consistency properties. *Project 21-49-004, Report 4, pp. 261-279. US Air Force School of Aviation Medicine, Randolph Field.*

[7] Friedman, J. H. (1994) Flexible metric nearest neighbor classification. *Tech. Report, Dept. of Stat., Stanford University.*

[8] Friedman, J. H. (1996) Another approach to ploychotomous classification. *Tech. Report, Dept. of Stat., Stanford University.*

[9] Ghosh, A. K. and Chaudhuri, P. (2004) Optimal smoothing in kernal discriminant analysis. *Statistica Sinica*, **14**, 457-483.

[10] Ghosh, A. K., Chaudhuri, P. and Murthy, C. A. (2005) On visualization and aggregation of nearest neighbor classifiers. *IEEE Trans. Pattern Analysis and Machine Intell.* (To appear).

[11] Hastie, T. and Tibshirani, R. (1996) Discriminant adaptive nearest neighbor classification. *IEEE Trans. Pattern Analysis and Machine Intell.*, **18**, 607-16.

[12] Hastie, T. and Tibshirani, R. (1998) Classification by pairwise coupling. *Ann. Statist.*, **26**, 451-471.

[13] Hastie, T., Tibshirani, R. and Friedman, J. H. (2001) *The Elements of Statistical Learning : Data Mining, Inference and Prediction.* Springer Verlag, New York.

[14] Holmes, C. C. and Adams, N. M. (2002) A probabilistic nearest neighbor method for statistical pattern recognition. *J. Royal Statist. Soc., B*, **64**, 295-306.

[15] Holmes, C. C. and Adams, N. M. (2003) Likelihood inference in nearest-neighbor classification methods. *Biometrika*, **90**, 99-112.

[16] Johnson, R. A. and Wichern, D. W. (1992) *Applied Multivariate Statistical Analysis.* Prentice Hall, New Jersey.

[17] Lachenbruch, P. A. and Mickey, M. R. (1968) Estimation of error rates in discriminant analysis. *Technometrics*, **10**, 1-11.

[18] Loftsgaarden, D. O. and Quesenberry, C. P. (1965) A nonparametric estimate of multivariate density function. *Ann. Math. Statist.*, **36**, 1049-1051.

[19] Mahalanobis, P. C. (1936) On the generalized distance in statistics. *Proceedings of the National Academy of Sciences, India*, **12**, 49-55.

[20] McLachlan. G. J. (1992) *Discriminant Analysis and Statistical Pattern Recognition.* Wiley, New York.

[21] Paik, M. and Yang, Y. (2004) Combining nearest neighbor classifiers versus cross-validation selection *Statistical Applications in Genetics and Molecular Biology*, **3**, Article 12.

[22] Ripley, B. D. (1996) *Pattern Recognition and Neural Networks.* Cambridge University Press, Cambridge.

[23] Silverman, B. W. (1986) *Density Estimation for Statistics and Data Analysis.* Chapman and Hall, London.

[24] Stone, M. (1977) Cross validation : a review. *Mathematische Operationsforschung und Statistik, Series Statistics*, **9**, 127-139.