On Visualization and Aggregation of Nearest Neighbor Classifiers

Anil K. Ghosh[+], Probal Chaudhuri[+] and C. A. Murthy[*]

[+]Theoretical Statistics and Mathematics Unit, [*]Machine Intelligence Unit, Indian Statistical Institute, 203, B. T. Road, Calcutta-700108, India.

email : res9812@isical.ac.in, probal@isical.ac.in, murthy@isical.ac.in

Abstract

Nearest neighbor classification is one of the simplest and most popular methods for statistical pattern recognition. A major issue in k-nearest neighbor classification is how to find an optimal value of the neighborhood parameter k. In practice, this value is generally estimated by the method of cross-validation. However, the ideal value of k in a classification problem not only depends on the entire data set, but also on the specific observation to be classified. Instead of using any single value of k, this article simultaneously studies the results for a finite sequence of classifiers indexed by k. The results of these classifiers and their corresponding estimated misclassification probabilities are visually displayed using shaded strips. These plots provide an effective visualization of the evidence in favor of different classes when a given data point is to be classified. We also propose a simple weighted averaging technique that aggregates the results of different nearest neighbor classifiers to arrive at the final decision. Based on the analysis of several benchmark data sets, the proposed method is found to be better than using a single value of k.

Index Terms : Bayesian strength function, misclassification rates, multi-scale visualization, neighborhood parameter, posterior probability, prior distribution, weighted averaging.

1 Introduction

In supervised classification problems, we usually have a training sample of the form $\{(\mathbf{x}_n, c_n);\ n = 1, 2, \ldots, N\}$, where the $\mathbf{x}_n$s are the measurement vectors, and the c_ns are the class labels of the training sample observations. Based on this available training sample, one 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 to be classified to the j^{th} population if $\mathbf{x} \in \mathcal{X}_j$. There are some well known parametric [16], [30], and nonparametric [36], [10], [22], methods in the existing literature for finding such partitions. Nearest neighbor technique [11], [7], [9], [8], is one of the most popular nonparametric methods for this purpose.

In order to classify an observation by k-nearest neighbor method (k-NN), we assume that the posterior probability of a specific class to be constant over a small neighborhood around that observation. Generally, a closed ball of radius r_k is taken as this neighborhood, where r_k is the distance between the observation and its k^{th} nearest neighbor. We classify an observation to the class which has the maximum number of representatives in this neighborhood. The parameter k, which determines the size of this neighborhood, can be viewed as a "smoothing parameter" related to the smoothness of the posterior probability estimates, and in future we will refer to it as the *neighborhood parameter*. As k gets larger, posterior probability estimates tend to be smoother in some sense. Therefore, different values of k can be viewed as different scales of smoothing. A discussion on the bias and the variance of the posterior probability estimates for different k is available in [15], [13] and [10]. The performance of the nearest neighbor classification rule depends heavily on the value of this neighborhood parameter k. Existing theoretical results [28], [7], [15], suggest that k should depend on the training sample size N, and it should vary with N in such a way that $k \to \infty$ and $k/N \to 0$ as $N \to \infty$. In practice, the optimal value of k depends on the available training sample observations, and one

generally uses resampling techniques like cross-validation [27], [39], to determine it. However, the optimal value of k is case specific, and it depends on the observation to be classified in addition to depending on the competing population distributions. Therefore, in a classification problem, instead of fixing the value of k, it may be of more use to look at the results for different neighborhood parameters and then combining them to come up with the final decision.

In this article, we have made an attempt to study the results for different neighborhood parameters simultaneously. Broadly speaking, this paper has two major components. In section 2, we propose some discrimination measures to study the strength of the classification results for different k and develop a device for visual presentation of these results using shaded strips. The resulting plots provide a visual comparison between the strength of evidence in favor of different classes for a specific data point in the sample space. They are useful, especially in higher dimensional spaces, to visualize the distribution of data points in the training sample from different populations in neighborhoods of varying sizes of a test case. It eventually helps in making the final decision about classification. Such visual approach in discriminant analysis is available also in [17] and [18], where the authors used a range of values for bandwidth parameters of the kernel density estimates of different competing classes. Earlier authors [5], [19], used similar ideas to visualize significant features in univariate and bivariate function estimation problems.

The other major components of the paper is introduced in Section 3 and it is about the aggregation of different nearest neighbor classifiers. Here, we use a weighted averaging technique to aggregate nearest neighbor classifiers with varying choices of k. The weights of different classifiers are determined using the corresponding estimated (by leave-one-out cross validation) misclassification rates. Well known aggregation techniques like bagging [3], boosting

[37], [14] and arcing [4] also adopt similar approaches for final classification. Recently, Paik and Yang [33] have proposed a similar method for aggregating nearest neighbor classifiers, where the authors used likelihood scores to determine the weights. Alpaydin [1] and Holmes and Adams [24], [25] also developed some aggregation techniques for combining nearest neighbor classifiers. Details of our aggregation methods are given in Section 3 and their performance on some benchmark data sets is reported in Section 4. In this section, we also discuss about the computational complexities of our proposed methods. Section 5 contains a brief summary of the work and related discussion.

2 Visualization of k-NN classification results

Given an observation $\mathbf{x}$, let $\mathbf{x}^{(k,N)}$ be its k^{th} nearest neighbor and $r_k = \rho(\mathbf{x}, \mathbf{x}^{(k,N)})$ be the distance between them. A k-nearest neighbor rule classifies the observation $\mathbf{x}$ to the j^{th} population if $\sum_{n=1}^{N} I\{\rho(\mathbf{x}, \mathbf{x}_n) \leq r_k,\ c_n = j\} \geq \sum_{n=1}^{N} I\{\rho(\mathbf{x}, \mathbf{x}_n) \leq r_k,\ c_n = i\} \quad \forall\ i \neq j$, where $I\{\cdot\}$ denotes the usual 0-1 valued indicator function. Ties can be resolved by gradually shrinking or extending this neighborhood. Simple Euclidean metric is one very popular choice for the distance function ρ. Of course, one may use Mahalanobis distance [29] or any other flexible and adaptive metric [12], [21], as well. In the case of Euclidean distance, for consistency of the nearest neighbor classifier, one allows k to vary with N such that $k \to \infty$ and $k/N \to 0$ as $N \to \infty$ [28], [7], [15]. Under the same conditions, one can establish this consistency also in the case of Mahalanobis distance, when consistent estimates are used for class dispersion matrices.

Fukunaga and Hostetler [15] and McLachlan [30] discussed the importance of finding an optimal value of k for moderately large and small sample sizes. This value is generally estimated from the training sample using cross-validation techniques. However, in a classification problem,

a single value of k often fails to give the true picture for classification of all data points. For instance if $k = 1$ is selected as the optimal value by the method of cross-validation (which may happen in practice), for each observation, it will lead to posterior estimates of 0 or 1, which does not give any idea about the strength of classification and the related uncertainties present there. Analysis using multiple values of k becomes helpful in such situations. In this article, instead of going for the estimation of this optimal value, we shall study the performance of different nearest neighbor classifiers indexed by k simultaneously to build up a more informative discrimination procedure. For a fixed value of k, one can use any suitable distance function to identify the neighbors and to determine which one of the J classes is the most favorable. We now introduce some measures for the strength of this evidence for a specific test case in favor of different competing classes and for different values of the neighborhood parameter.

2.1 Posterior probabilities for different populations

Given a data point $\mathbf{x}$ and a given value of the neighborhood parameter k, the proportion of observations $\sum I\{\rho(\mathbf{x}, \mathbf{x}_k) \leq r_k,\ c_k = j\}/k$ is taken as an estimate $\hat{p}_j = \hat{p}(j \mid \mathbf{x})$ for the posterior probabilities of different classes ($j = 1, 2, \ldots, J$). In this article, to make the notations simpler, we drop the argument $\mathbf{x}$ since the dependence on $\mathbf{x}$ is obvious in all cases. For any fixed value of k, the estimated posterior probabilities determine the favorable class, and they also give an idea about the strength of discrimination.

Consider the following example of salmon data taken from [26]. It consists of 100 bivariate observations on growth ring diameter (freshwater and marine water) of salmon fish coming from Alaskan or Canadian water. A scatter plot of this data set is given in Figure 2.1, where dots ('$\cdot$') and crosses ('$\times$') represent the observations coming from Alaskan and Canadian populations,

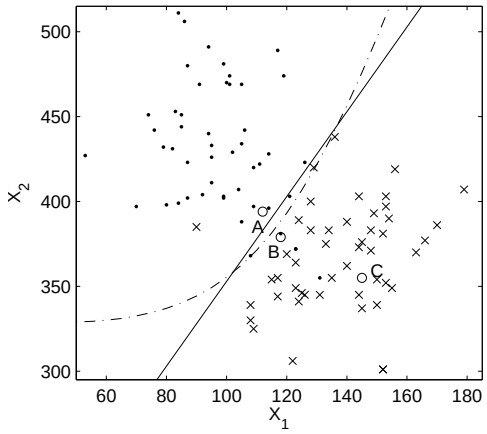

Figure 2.1 : Scatter plot of salmon data.

respectively. We chose three observations at three different parts of the data (marked by '$\circ$' in the figure) for which the class information is known and classified them using the remaining observations. Observation 'A' belongs to the Alaskan population, 'B' and 'C' belong to the Canadian. Estimated class boundaries for linear and quadratic discriminant analysis are also shown in the figure. In this figure one can notice that the evidence visible in the scatter plot in favor of the true class is much stronger in case of 'C' than that in the other two cases. Observations 'A' and 'B' belong to two different populations but they are located near the class boundary. So, one should expect to have three different behaviors of the classification methodology for these three observations. Using simple Euclidean distance function and leave-one-out cross-validation method on this data set, we obtained $k = 7$ as the optimal neighborhood parameter, which failed to exhibit the difference in the strength of classification. It classified the observations 'A' and 'C' correctly but led to the same posterior estimate (6/7) for the true classes. Moreover, the observation 'B' got misclassified by this method.

Using multiple values of k in this case, we obtained a much better result. The results of this multi-scale analysis for these observations are given in Figure 2.2, which shows the grey scale values of posterior probabilities. Here 'white' and 'black' colors represent the posterior

probabilities 1 and 0, respectively. Differences in the classification results and their strength are quite evident from this figure. It clearly suggests that the strength of the evidence in favor of the true class is much higher in case of observation 'C' than those in the other two cases as one would normally expect from Figure 2.1. Moreover, the plots for observations 'A' and 'B' show some interesting features. For small values of k, class-1 (Alaskan salmon) has an edge, but for higher values of k, class-2 (Canadian salmon) seems to be the winner. Both for 'A' and 'B', throughout the figure, we observe very little difference in the posterior probability estimates of the two classes, which gives a clear indication of border line cases.

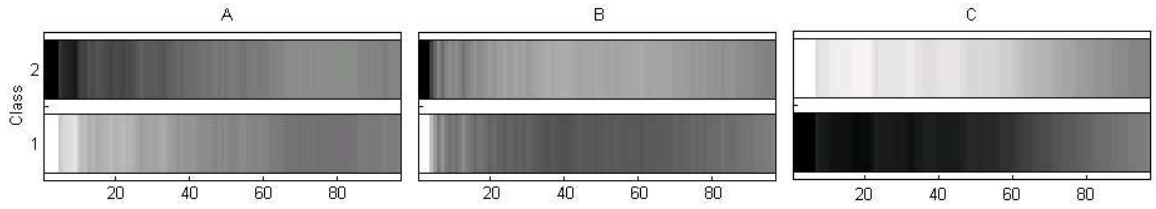

Figure 2.2 : Shaded strips describing posterior probabilities for different values of k.

2.2 A Bayesian measure for strength of evidence for different populations

Posterior probabilities used in nearest neighbor classification are estimated based on the assumption that the probability of a specific class is fixed and non-random in a neighborhood around a specific point $\mathbf{x}$. Instead of that, we now assume a prior distribution for these probabilities. Suppose that $\pi(\mathbf{p})$ is the prior distribution of $\mathbf{p} = (p_1, p_2, \ldots, p_J)$ $[\sum_{j=1}^{J} p_j = 1]$, where p_j is the probability corresponding to the j^{th} class. Now, for some given k, consider the k nearest neighbors of an observation $\mathbf{x}$. If t_{j_k} of these k neighbors come from the j^{th} class, the multinomial

distribution of $\mathbf{t}_k = (t_{1_k}, t_{2_k}, \ldots, t_{J_k})$ $[\sum_{j=1}^{J} t_{j_k} = k]$ for a given $\mathbf{p}$ and k can be expressed as

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

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

$$f(\mathbf{p} \mid k, \mathbf{t}_k) = \pi(\mathbf{p})\ \varphi(\mathbf{t}_k \mid \mathbf{p}, k) \Big/ \int \pi(\mathbf{p})\ \varphi(\mathbf{t}_k \mid \mathbf{p}, k)\ d\mathbf{p}.$$

Using this conditional distribution, we define the Bayesian measure of strength for different populations. Clearly, one will prefer the j^{th} class compared to the i^{th} one if $P\{p_j > p_i \mid k, \mathbf{t}_k\} > P\{p_i > p_j \mid k, \mathbf{t}_k\}$. Following this idea, for a given value of k, the Bayesian strength function for the j^{th} population is defined as

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

Usual estimated posterior probabilities sometimes fail to give a clear idea about the strength of evidence in a discrimination procedure. For instance, in a two-class problem, the posterior estimate for one class turns out to be one in all those situations where the neighbors come from the same population, but certainly the strength is not the same in all these cases. For instance, if 10 out of 10 neighbors are from the same population, that evidence should be considered as stronger than that obtained in a 1-nearest neighbor method. These differences in the strength of discrimination get captured by the strength function S. In a two-class problem, if the training sample observations form two perfectly separated clusters each consisting of observations from one class, the posterior probability estimate for an observation will remain one up to a certain value of k. But the Bayesian strength function obtained using a uniform prior distribution on [0,1] will keep on increasing as long as we get neighbors from the same class.

Figure 2.3 shows the grey-scale value of this Bayesian strength function for the three observations 'A', 'B' and 'C', when $\pi(\mathbf{p})$ is taken to be the uniform distribution on $[0, 1]$. Uniform prior distribution is very easy to handle, both computationally and theoretically. Further, a uniform choice of prior is *unbiased* and *non-informative* as a prior distribution that does not distinguish a priori between different classes. Throughout this article, we use the uniform prior distribution to obtain the Bayesian strength function. However, our empirical study suggests that the results are not very sensitive to the choice of the prior distribution, and one may use many other suitable priors as well.

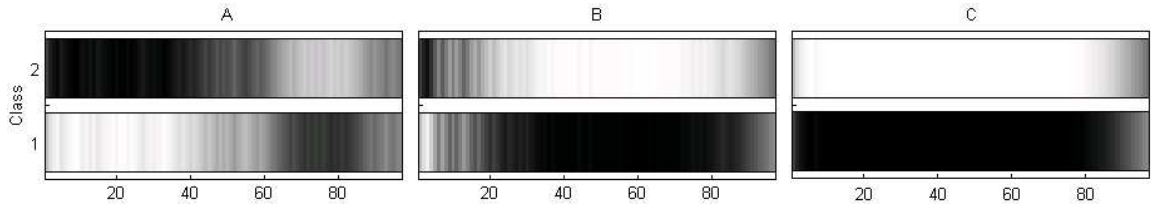

Figure 2.3 : Shaded strips describing Bayesian strength functions for different values of k.

Once again, Figure 2.3 indicates a stronger evidence in favor of the true class for observation 'C', but for the other two cases, we observed white as well as black shades depending on the values of k. The fraction of white or black areas in the shaded strips gives us some rough idea about final classification of these observations. One should also note that in all these cases, this strength function leads to sharper images by making the plots more white or more black, and thereby makes it easier to identify the winner. From the plots of posterior probabilities in Figure 2.2, it is quite difficult to judge which class is the winner for observations 'A' and 'B', but the plots of Bayesian strength functions in Figure 2.3 makes it quite clear. In that sense, Bayesian strength function is useful alternative to the posterior estimate as a visualization tool. Some interesting theoretical properties of this strength function is given in the following theorem.

Theorem 2.1 : *For some fixed k consider the k-nearest neighbor classifier, and let $\hat{p}_j$ be*

the estimated posterior probability for the j^{th} $(j = 1, 2, \ldots, J)$ *population as described in Section 2.1. Also, assume that* $\pi(\mathbf{p})$ *is symmetric in its arguments. Then,* $S(j \mid k) > S(i \mid k)$ *if and only if* $\hat{p}_j > \hat{p}_i$*. Further, in a two-class problem,* $S(j \mid k)$ *increases with* $\hat{p}_j$*.*

From the above theorem it is quite evident that the posterior probability estimates and the Bayesian strength functions are equivalent as far as k nearest neighbor classification of a given test case is concerned for any fixed k. In other words, the posterior ordering of different classes is strictly preserved in the Bayesian strength functions for different classes. Moreover, in the case of binary classification, given the value of k, our strength function is a strictly increasing function of the posterior estimate. Therefore, Bayesian strength function does not lead to any loss of information but it sharpens the plots. Sharpening occurs due to the enhancement in the difference of the discrimination measures, and it increases the visual separability without disturbing the order of the posterior estimates. In that sense, this Bayesian strength function is a new discrimination measure, which provides an effective alternative to posterior probability, especially for the visualization purpose. The sharpening property of the strength function gets well explained by the following theorem.

Theorem 2.2 : *Suppose that in a J-class problem,* $\pi_1, \pi_2, \ldots, \pi_J$ *are the priors and* $f_1, f_2, \ldots, f_J$ *are the continuous probability density functions for J populations. For a given* $\mathbf{x}$*, define* $P_j = \pi_j f_j(\mathbf{x}) / \sum_{l=1}^{J} \pi_l f_l(\mathbf{x})$ *as the conditional probability of the* j^{th} *population* $(j = 1, 2, \ldots, J)$*. Now, assume that* (i) $P_i > P_j$ *for all* $j \neq i$*, and* (ii) $k \to \infty$ *and* $k/N \to 0$ *as* $N \to \infty$*. If* $\pi(\mathbf{p})$ *is symmetric in its arguments,* $S(i \mid k) \xrightarrow{P} 1$ *and* $S(j \mid k) \xrightarrow{P} 0$ *for all* $j \neq i$ *as* $N \to \infty$*.*

From the existing results [28], [7], [15], we know that under the conditions of Theorem 2.2, the estimates of the posterior probabilities converge to the true posteriors, which are values

between 0 and 1. But in such cases, our Bayesian strength function converges to either 0 or 1 and makes the evidence sharper in favor of the class having the largest true posterior, and this is visible in the images in the corresponding plots in Figure 2.3.

In two-dimensional problems, we can always get an idea about the location of a data point from the scatter plot itself. But in higher dimensions, it is difficult to visualize which points are near the population boundaries and which are away from them. The visual display of the discrimination measures is quite useful in such situations. It gives a visual idea about the distribution of data points from different classes in a neighborhood of a test case and thereby helps to differentiate between the border line and clear cut cases. It is appropriate to note here that as we look at those shaded strips of posterior probabilities and Bayesian strength measures and try to decide about the class to which the test is likely to belong, it leads to some kind of a visual aggregation of the results. In a sense, this visual aggregation can be viewed as a useful supplement to mathematical aggregation procedure described in the following section.

3 Aggregation of nearest neighbor classifiers

Discrimination measures like posterior probability and Bayesian strength function show evidence in favor of different classes for various choices of k. In many of the cases, from the plots of these discrimination measures the final result becomes quite transparent. However, this may not be the case always. For instance, in the case of observations 'A' and 'B' in salmon data, we observe strong evidence for class-1 when k is small, but for relatively larger values of k, the plot gives an indication in favor of the other class. As we have mentioned earlier, fractions of white and black areas in the shaded strips give a tentative idea about the class to which an observation belongs. However, to reach the final decision, it is also important to know the reliability of classification

results for different values of k. From the corresponding estimated misclassification rate, we get an idea about that.

Here, we have used standard leave-one-out cross-validation method to estimate these misclassification rates and plotted the estimated probabilities of correct classification (re-scaled to have a minimum value 0 and maximum value 1) for different choices of k (see Figure 3.1). This plot shows the performance of nearest neighbor classifiers with different neighborhood parameters, where white and black colors indicate the lowest and the highest misclassification rates, respectively.

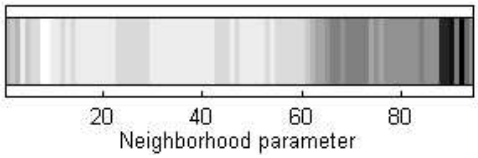

Figure 3.1 : Shaded strips describing the probability of correct classification.

3.1 Details of the aggregation procedure

A natural way to aggregate the results of different classifiers is to take some kind of average of the discrimination measures. Instead of taking equal weights for all classifiers, it is more reasonable to give priority to those values of k, which lead to lower misclassification rates. The weight function should take the highest value for the classifier(s) which leads(lead) to the lowest estimate of misclassification error, and it should decrease gradually as the error rate increases. Well-known aggregation methods like bagging [3], boosting [37], [14], and arcing [4] also adopt similar strategy and assign different weights to different classifiers based on their corresponding misclassification rates. An empirical study on the performance of these different ensemble methods is available in [32]. Breiman [3] pointed out that nearest neighbor classifiers are more stable than neural nets [36] or classification trees [2], and there is not much gain in

combining these classifiers using bagging or boosting. Shalak [38] suggested to combine the classifiers only when they have reasonable amount of diversity among themselves. However, it should also be noted that in terms of misclassification rates, one would normally expect to gain by combining classifiers, and diversity among classification rules can be viewed as a measure of extent to which the error rates can be improved. Over the last few years, there has been a revival of interest in aggregating nearest neighbor classifiers. Alpaydin [1] used the condensed nearest neighbor approach [20] and combined a number of nearest neighbor rules developed on different representative sets from the training sample. Ho, Hull and Srihari [23] tried to build up a multiple classifier system based on class ranks. Recently, Holmes and Adams [24], [25], developed a probabilistic framework for nearest neighbor classification, where they combined the nearest neighbor rules by using a likelihood based method and a Bayesian technique. Paik and Yang [33] used a likelihood based weighting scheme for aggregation of nearest neighbor classifiers.

The aggregation procedure that we adopt in this article is much simpler than most of these recently developed techniques. It uses an weighted average of discrimination measures to reach the final decision, and consequently, it is computationally more straight forward in the sense that it does not involve any iterative computations like some of the other techniques [24], [25], for aggregating nearest neighbor classifiers. The aggregated decision rule is given by

$$\mathsf{d}(\mathbf{x}) = \arg\max_j \sum_k \omega(k)\mathcal{D}_j(k,\mathbf{x}),$$

where $\mathcal{D}_j(k,\mathbf{x})$ is the value of discrimination measure (i.e., the posterior probability or the Bayesian strength function) for the j^{th} population (at the point $\mathbf{x}$), and $\omega(k)$ is the weight function associated with the k-nearest neighbor classifier. As we have mentioned before, weights should be chosen in such a way that it is higher for those values of k, which lead to lower

misclassification rate $\Delta(k)$, which is estimated by leave one out cross-validation (the estimate is denoted as $\widehat{\Delta}(k)$). In this article, we have used the weight function

$$\omega(k) = \begin{cases} e^{-\frac{1}{2}\left\{\frac{\widehat{\Delta}(k)-\Delta_\circ}{\sqrt{\Delta_\circ(1-\Delta_\circ)/N}}\right\}^2} & \text{if } \frac{\widehat{\Delta}(k)-\Delta_\circ}{\sqrt{\Delta_\circ(1-\Delta_\circ)/N}} \leq \tau \quad \text{and} \\ 0 & \text{otherwise,} \end{cases}$$

where N is the training sample size, and $\Delta_\circ = \min_k \widehat{\Delta}(k)$ (see also [17], [18]). Notice that $\Delta_\circ$ and $\Delta_\circ(1-\Delta_\circ)/N$ can be viewed as estimates for the mean and the variance of the empirical misclassification rates of the best nearest neighbor classifier, when it is used to classify N independent observations. The constant τ determines the maximum amount of deviation from $\Delta_\circ$ in a standardized scale beyond which the weighting scheme ignores the classifiers by putting zero weight on them. Clearly, $\tau = 0$ corresponds to the situation of putting all the weights only on those classifiers $\mathcal{C}_k$ for which $\widehat{\Delta}(k) = \Delta_\circ$. Because of the choice of a Gaussian weight function above, one does not have to consider a value of τ larger than 3 in practice.

Our choice of weight function is subjective, and instead of that one may use many other suitable functions. However, our empirical experience suggests that the final result is not very sensitive to the choice of the weight function as long as any reasonable weight function is used that decreases with the cross-validated error at an exponential rate or at a polynomial rate with a very high degree.

As discussed in [33], it is better to combine the nearest neighbor classifiers, when there are multiple values of k leading to misclassification rates close to the $\Delta_\circ$. Otherwise, it is better to adopt the classification method based on the cross validated choice of k. Note that the weight function $\omega(k)$ automatically takes care of it.

Let us consider the example on salmon data once again. For $\tau = 3$, weighted averages of posterior probabilities in favor of Alaskan population for the three observations 'A', 'B' and

'C' in Figure 2.1 were found to be 0.6261, 0.4146 and 0.1445 respectively, while we obtained 0.7777, 0.2808 and 0.0134 as weighted averages of Bayesian strength functions for the respective cases. One should notice that aggregation of both the discrimination measures could correctly classify all the three observations. However, we have already pointed out in Section 2.1 that if leave-one-out cross-validated choice of k is used, the resulting classifier failed to classify the observation 'B' correctly.

4 Case studies

In this section, we use some benchmark data sets to illustrate the usefulness of the proposed methods. Some of these data sets were used in [1] and [24], [25] for combining the nearest neighbor classification rules. We have quoted some results directly from those articles and compared the performance of our aggregation methods with them. Error rates for the best nearest neighbor rule (selected on the basis of leave-one-out cross-validation) are also reported to facilitate the comparison. Cross-validation methods estimate the misclassification rates by naive empirical proportions, and as a result, it is often possible to have more than one value of k that will lead to the smallest value of $\widehat{\Delta}(k)$. Since nearest neighbor classifiers assume the posterior probability of a specific class to be constant over the entire neighborhood, it is reasonable to consider the lowest value of k in such cases. Throughout this section, we have used $\tau = 3$ for aggregating nearest neighbor classifiers. In all the cases, we use the Euclidean metric after standardizing the data sets using an estimate of the pooled dispersion matrix. This essentially leads to nearest neighbor classification using Mahalanobis distances [29]. For those data sets, which have separate training and test sets, we estimated the pooled dispersion matrix using the training sample, while in all other cases, the full data set was used for estimation.

We have already described the salmon data in Section 2 for the purpose of illustration. Among the other data sets analyzed in this section, description of vowel data-1 is given in [6]. Adult data can be obtained form Delve data archive (http://www.cs.toronto.edu/∼delve). Iris data, biomedical data, chemical and overt diabetes data (referred to as 'diabetes data' in this article), synthetic data and glass data are available at CMU data archive (http://lib.stat.cmu.edu). The rest of the data sets (i.e., Pima Indian diabetes data, Australian credit data, wine data and vowel data-2) and their descriptions can be obtained from UCI machine learning repository (http://www.ics.uci.edu).

4.1 Comparison with the cross-validated selection of neighborhood parameter

We begin with the comparison between our aggregation techniques and the usual nearest neighbor method, where a single value of k (chosen by leave-one-out cross-validation) is used for classification. Both, weighted average of posterior and that of Bayesian strength functions are used as aggregation methods. In this section, we use only five data sets for comparison. Results on other data sets will be reported in subsequent sections, where we will compare the performance of the proposed methods with some other aggregation techniques available in the literature. Out of these five data sets, vowel data-1 has separate training and test samples. For this data set, we report the test set misclassification errors for different classifiers. In all other cases, we formed the training and the test samples by randomly partitioning the data. This random partitioning was carried out 300 times to generate 300 training and test sets. Average test set misclassification rates over these 300 partitions are reported (see Table 4.1) for different methods along with their corresponding standard errors. Only in the case of adult data, results

are based on 100 random partitions. Sizes of the training and the test samples in each partition are also given in the table.

Table 4.1 : Misclassification rates (in %) for usual and combined nearest neighbor classifiers

Data sets	Sample size		k-NN	Weighted	Weighted
	Training	Test	(cross-valid.)	posterior	strength
Salmon	30	70	9.38 (0.18)	8.30 (0.14)	8.30 (0.15)
Vowel-1	338	333	17.75 (2.09)	18.93 (2.15)	19.25 (2.16)
Diabetes	50	95	11.50 (0.16)	10.47 (0.15)	10.71 (0.15)
Biomedical	100	94	17.61 (0.18)	17.10 (0.16)	17.56 (0.17)
Adult$^+$	32561	16281	20.21 (0.04)	20.14 (0.04)	20.19 (0.05)

Figures in braces are standard errors (in %)

$^+$ We did not consider the eight categorical variables and two numerical variables, which have most of the values zero, and carried out our analysis using only the remaining four variables.

Apart from vowel data-1, in all other cases, the aggregation methods performed better than the usual nearest neighbor method based on cross-validated choice of k. In the case of salmon data and diabetes data, both weighted posterior and weighted Bayesian strength led to significantly lower misclassification rates compared to that of the usual nearest neighbor classification. There was no significant difference between the error rates of these two aggregation techniques. However, in the case of biomedical data, weighted averaging of posteriors could led to significantly lower misclassification rates compared to the other two classification methods. Only for vowel data-1, the performance of the usual nearest neighbor classifier based on leave-one-out cross-validated estimate of neighborhood parameter was marginally better than the performance of the proposed aggregation methods. But in view of high standard errors, these differences are statistically insignificant.

4.2 Comparison with probabilistic nearest neighbor and likelihood based aggregation procedure

In this section, we have used some benchmark data sets to compare the performance of the proposed aggregation procedure with that of usual k-nearest neighbor methods (with k chosen by leave-one-out cross-validation) and other aggregation methods suggested in [24], [25]. In [24], Holmes and Adams developed a probabilistic framework for nearest neighbor classification, where they proposed an aggregation procedure based on Bayesian techniques using Markov Chain Monte Carlo ($MCMC$) methods. In another article [25], they developed an aggregation method based on a likelihood function, where iteratively re-weighted least squares technique was used to estimate the posterior probabilities. In both these articles, the authors used some benchmark data sets to evaluate the performance of their aggregation methods. We have taken four of those data sets for comparison. Out of these four data sets, synthetic data and vowel data-2 have separate training and test sets. In these cases, we have reported the test set misclassification rates for different classifiers. For the other two data sets, (Pima Indian data and Australian credit data), the reported results are the cross-validated error rates. We partitioned these data sets into 12 and 10 folds for the Pima Indian and the Australian credit data, respectively, as it has been done in [24] and [25]. We have repeated this partitioning 25 and 30 times for Pima Indian and Australian credit data, respectively, and the average cross-validated error rates over those 25 and 30 trials are reported in Table 4.2 along with their corresponding standard errors. Note that this way of repeated partitioning leads to 300 training and test set combinations both for the Pima Indian and the Australian credit data. The result of probabilistic nearest neighbor method on vowel data-2 was not reported in [24], which is why we have a blank space in the table.

Table 4.2 : Misclassification rates (in %) for different nearest neighbor classifiers

Data sets	k-NN (cross valid.)	Likelihood (Holmes-Adams)	Prob. NN (Holmes-Adams)	Weighted posterior	Weighted strength
Synthetic	11.70 (1.02)	8.2	8.4	9.80 (0.94)	9.90 (0.94)
Vowel-2	46.75 (2.32)	49.3	—	46.75 (2.32)	46.75 (2.32)
Pima Indian	25.27 (0.25)	23.9	24.7	24.48 (0.24)	24.64 (0.24)
Australian credit	13.20 (0.23)	13.3	14.7	13.16 (0.24)	12.97 (0.23)

Figures in braces are standard errors (in %)

In all these data sets, the performance of our proposed aggregation methods was fairly competitive with the other nearest neighbor classifiers. For the synthetic data and the Pima Indian data, our aggregation methods could achieve lower misclassification rates than that of the usual k-nearest neighbor classifier, while in the other two cases they have similar error rates. When unstandardized version of the synthetic data was used for classification, the k-nearest neighbor method with cross-validated choice of k led to an error rate of 9.7%, but both the aggregation methods, weighted posterior and weighted Bayesian strength, could reduce it to 8.30%. These two proposed aggregation techniques based on weighted averaging had nearly the same misclassification rates in all these examples. The standard errors indicate that the error rates of our aggregation methods are not significantly different from the error rate of the usual nearest neighbor classifier and that of the other aggregation methods proposed by Holmes and Adams [24], [25]. However, our aggregation methods are computationally more straight forward than the iterative computations required in the likelihood based and the probabilistic nearest neighbor algorithms.

4.3 Comparison with weighted CNN methods

Next, we compare our method with the performance of aggregated condensed nearest neighbor (CNN) classifier reported in [1]. Along with the vowel data-2, the wine data, the Iris data and the glass data are also used for this comparison. Vowel data-2 has separate training and test sets. For the other data sets, we used the random partitioning method to generate the training and test sets of the same sizes as used in [1].

For each of these data sets, Alpaydin [1] used CNN method on 10 representative samples taking from the training set and combined them by some weighted averaging procedure. They also proposed another classification method "NN-union", which classifies an observation using the union of the representative sets as the training sample. But, the author did not perform these experiments over different training and test sets. However, we divided the data sets (except for the vowel data-2, which has a given test set) randomly to form 300 different training and test sets and carried out our analysis over those 300 random partitions. Average test set misclassification rates over those 300 trials and their corresponding standard errors are reported in Table 4.3.

Table 4.3 : Misclassification rates (in %) for nearest neighbor and CNN classifiers

Data	Sample size		k-NN	Condensed NN	Condensed NN	NN on	Weighted	Weighted
sets	Train	Test	(cross valid.)	simple	weighted	union	posterior	strength
Vowel-2	528	462	46.75 (2.32)	43.44	44.03	42.86	46.75 (2.32)	46.75 (2.32)
Iris	15	135	4.12 (0.11)	7.33	6.00	7.78	3.71 (0.10)	3.68 (0.10)
Wine	100	78	1.05 (0.07)	6.15	5.00	6.03	0.45 (0.05)	0.41 (0.04)
Glass	100	114	34.59 (0.21)	30.00	28.33	28.60	34.93 (0.20)	35.51(0.23)

Figures in braces are standard errors (in %)

Once again, our proposed aggregation methods showed a competitive performance in all the data sets. In Iris data and wine data, both weighted posterior and weighted Bayesian

strength performed significantly better than the other nearest neighbor classifiers. For vowel data-2, there was no significant difference between the error rates of different classifiers. Alpaydin [1] reported the best error rate when unstandardized version of this data set was used for classification. On that unstandardized data, both of the usual nearest neighbor classifier and our weighted averaging methods lead to an error rate of 43.72%. However, these aggregation methods had a slightly higher error rates for glass data. But in view of high standard errors of the misclassification rates, their differences with the error rate of the usual nearest neighbor classifier based on cross-validated estimate of k were not statistically significant. It should be noted that in the case of glass data, there are only 9, 13 and 17 observations in three of the competing classes, and this makes it construction a good classifier for this data set very difficult.

In case of wine data, since the populations are quite well separated, we observed low cross-validated error rates over a wide range value of k. Consequently, the minimum error is obtained for a large number of choices of k. This makes it difficult to choose a single optimum value of k based on cross-validation method. Figure 4.1 shows the cross validated error rates and test set error rates for two different partitions of wine data, from which it is quite evident that there are multiple values of k, which lead to the lowest cross validation error. This feature of estimated error rates was obtained for almost all the 300 partitions. Paik and Yang [33] pointed out that aggregation is always better than cross-validation in such cases. We also observed the same thing. Both weighted average of posteriors and weighted average of Bayesian strength could lead to significantly lower misclassification rates than that of the usual nearest neighbor method, where k is selected by cross validation technique.

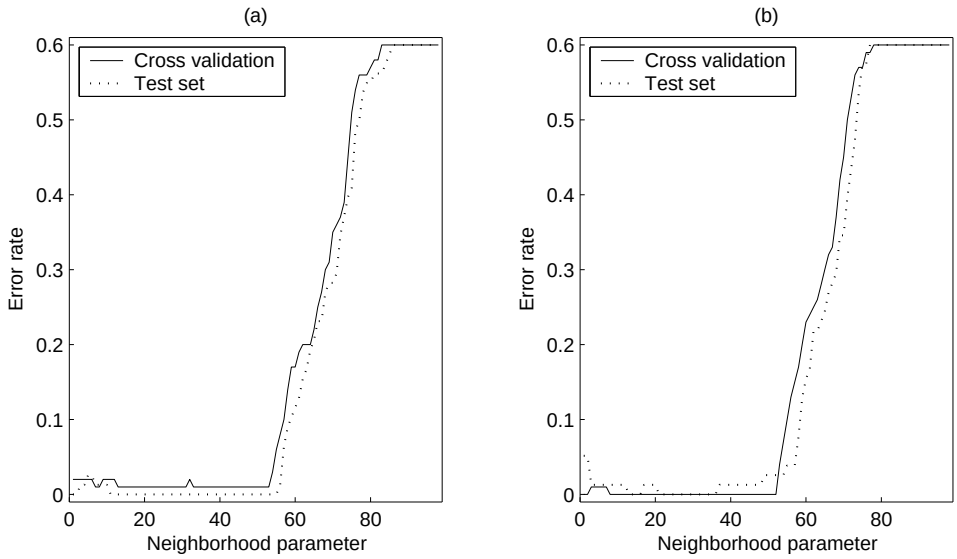

Figure 4.1 : Misclassification rates for two different partitions Wine data .

5 Computational aspects and related issues

This section deals with the computational complexities of our proposed aggregation methods. Since the dimension d is involved only in distance computation, we do not consider it in our calculation and 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 classification takes $O(NlogN)$ calculations. Clearly, this operation has to be repeated N times (taking one data point at a time) to find the leave-one-out error rates for different k, and that makes the computational cost $O(N^2logN)$. From the description given in Section 3.1, it is quite clear that our proposed aggregation methods require the same order of computations $O(N^2logN)$ to define the weight function. After finding the weights, a future observation is classified either using weighted posteriors or weighted Bayesian strength function. For classification of this new observation, both these methods require the same order of calculations $O(NlogN)$, which essentially arises due to sorting of N distances. Though both

these methods perform almost similar calculations, the number of computations is higher in the latter case as it requires the Bayesian strength functions to be computed instead of posterior probabilities. When $J \leq 3$, it is computationally feasible to use any numerical integration method based on an appropriate averaging of the integrand over a suitable grid in the domain of integration to approximate the integral appearing in the expression of the Bayesian strength function (see Section 2.2). Given the sorted array of distances, for a fixed value of k, while the number of computations for posterior probabilities is proportional to J, it is proportional to m^{J-1} for Bayesian strength computation, where m is the number of grid points chosen on each axis. When all possible values of k are considered, cost for Bayesian strength computation (for different k) becomes proportional to Nm^{J-1}. Though these calculations do not affect the order of computations for classification of a future test case, in many practical situations, when N is not very large, strength computation becomes computationally more expensive than sorting the distances. Since the computational cost for Bayesian strength function increases exponentially with the number of classes, for $J \geq 4$, we have adopted a different procedure for approximating Bayesian strengths of different populations. A large number (M_0) of samples are generated from appropriate Dirichlet distributions to approximate the strength functions for different populations. This method reduces the computational cost for strength function by making it proportional to NM_0. Throughout this article, we have taken $M_0 = 10000$ for our data analytic purpose. One should also note that given the value of k, the usual k-nearest neighbor algorithm requires $O(N)$ calculations to classify a future observation but in order to select that value of k it has to estimate the error rates for different k, which needs $O(N^2 logN)$ calculations if leave-one-out or any V-fold cross-validation method is used.

From the above discussion, it is quite clear that the computational cost for our aggregation techniques is the same as that of the usual cross validation method. We can reduce the

computational cost if we restrict our aggregation methods to values of k smaller than or equal to $\sqrt{N}$ instead of combining all nearest neighbor classifiers for $k = 1, 2, \ldots, N-1$. In that case, the weighted averaging methods require $O(N^2)$ calculations to define the weight function and $O(N)$ computations after that to classify a future observation. Clearly, this partial aggregation of nearest neighbor classifiers makes the aggregation procedure much faster, and the resulting classifiers in practice lead to a fairly satisfactory performance as well. This choice of $\sqrt{N}$ is partially motivated by the result on consistency of k-nearest neighbor classifier, which requires the conditions (i) $k \to \infty$ and (ii) $k/N \to 0$ as $N \to \infty$ to be satisfied. Some other authors [34], [31] have also used the same range of values for k in classification.

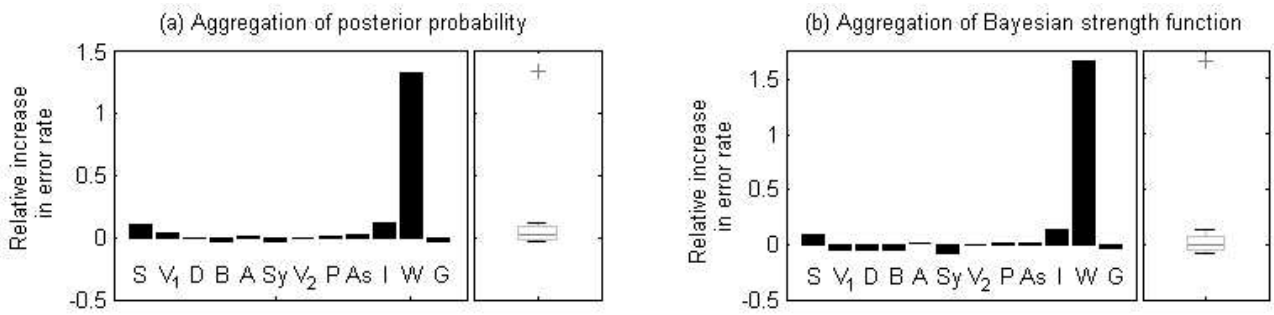

Figure 5.1 : Relative increase in misclassification rates due to truncated aggregation

A= Adult data, As= Australian credit data, B= Biomedical data, D= Diabetes data, G= Glass data,

I= Iris data, P= Pima Indian data, S= Salmon data, Sy= Synthetic data,

V_1= Vowel data-1, V_2= Vowel data-2, W= Wine data.

The bar diagrams and the box-plots in Figure 5.1 show the relative increase in misclassification rates (R) due to this partial aggregation. This relative change is defined as $R = (\Delta_2 - \Delta_1)/\Delta_1$, where Δ_1 and Δ_2 denote the error rate for the original aggregation and partial aggregation methods, respectively. Apart from wine data, in all other cases, error rates of the partial aggregation method could match the performance of the original aggregated clas-

sifier. From this figure, it is quite evident that the restriction ($k \leq \sqrt{N}$) did not affect the misclassification rates much, but it makes an enormous savings in computation in most of the cases.

6 Conclusions and discussion

This article describes two new methods for visual representation of classification results based on nearest neighbor classification algorithms. The first one is based on estimated posterior probabilities while the other one uses the Bayesian strength functions. Instead of using a single value of the neighborhood parameter, here we study the results for a finite sequence of nearest neighbor classifiers in order to get more information for classification and its strength. Visual displays lead to a way of comparison between the strengths of different competing populations for a range of values of the neighborhood parameter.

The second part of the paper describes an aggregation method, which is simpler than most of the similar aggregating procedures available in the literature. When compared with the usual nearest neighbor classification, where k is chosen by cross-validation techniques, these aggregation methods produced significantly better performance on some of the data sets, while their performance on the other data sets was also quite competitive. In view of the above data analysis, it is appropriate to conclude that it would usually be better to aggregate the results of nearest neighbor classifiers for different choices of neighborhood parameters than using a single optimum value of k estimated by cross-validation.

Both the aggregation methods, namely weighted posterior and weighted Bayesian strength, led to almost similar performance on all the data sets. The first one requires relatively less number of computations while the latter one makes the plots sharper preserving the ordering of the

classes according to their posterior probabilities (see Theorem 2.1 and 2.2). The choice of the method depends on the specific purpose of the user. When visualization is of prime interest, one will naturally look for the Bayasian strength function, whereas for aggregation the user will prefer the weighted averaging of posterior probabilities to arrive at the final decision.

Acknowledgement

We are thankful to the associate editor and the referees for their careful reading of the earlier version of the paper and providing us with several helpful comments.

Appendix

Proof of Theorem 2.1 : Without loss of generality, let us assume $i < j$. Since $\pi(\mathbf{p})$ is symmetric in its arguments, it is easy to see that

$$\int\limits_{p_i=max\ \{p_1,p_2,...,p_J\}} \left(\prod_{m=1}^{J} p_m^{t_m}\right)\ \pi(\mathbf{p})\ d\mathbf{p} = \int\limits_{p_j=max\ \{p_1,p_2,...,p_J\}} \left(\prod_{\substack{m=1\\ m\neq i,j}}^{J} p_m^{t_m}\right) p_i^{t_j}\ p_j^{t_i}\ \pi(\mathbf{p})\ d\mathbf{p}.$$

Now, note that $S(j \mid k)$ and $S(i \mid k)$ have the same denominator, which is positive, and the numerator of $S(j \mid k) - S(i \mid k) = \int\limits_{p_j=max\ \{p_1,p_2,...,p_J\}} \left(\prod_{\substack{m=1\\ m\neq i,j}}^{J} p_m^{t_m}\right)\ p_i^{t_i}\ p_j^{t_i}\ (p_j^{t_j-t_i} - p_i^{t_j-t_i})\ \pi(\mathbf{p})\ d\mathbf{p}.$

Since $p_j > p_i$ in this domain of integration, from the above expression, it is quite transparent that $S(j \mid k)$ is greater (smaller) than $S(i \mid k)$ if and only if t_j is grater (smaller) than t_i.

Further, in a two-class problem, suppose for some fixed k, in the neighborhood of $\mathbf{x}$, we have t_{1_k} and t_{2_k} observations from the two classes. Now let us define $t'_{1_k} = t_{1_k} + \alpha$ and $t'_{2_k} = t_{2_k} - \alpha$ for some positive integer α.

Now, it is fairly easy to check that

$$\gamma_1^{'} = \int_{0.5}^{1} p^{t_{1k}^{'}}(1-p)^{t_{2k}^{'}}\,dp = \int_{0.5}^{1} p^{t_{1k}}(1-p)^{t_{2k}}\left(\frac{p}{1-p}\right)^{\alpha} dp > \int_{0.5}^{1} p^{t_{1k}}(1-p)^{t_{2k}}\,dp = \gamma_1 \quad \text{and}$$

$$\gamma_2^{'} = \int_{0}^{0.5} p^{t_{1k}^{'}}(1-p)^{t_{2k}^{'}}\,dp = \int_{0}^{0.5} p^{t_{1k}}(1-p)^{t_{2k}}\left(\frac{p}{1-p}\right)^{\alpha} dp < \int_{0}^{0.5} p^{t_{1k}}(1-p)^{t_{2k}}\,dp = \gamma_2$$

This implies that $S^{'}(1 \mid k) = \frac{\gamma_1^{'}}{\gamma_1^{'}+\gamma_2^{'}} > \frac{\gamma_1}{\gamma_1+\gamma_2} = S(1 \mid k)$ and $S^{'}(2 \mid k) = \frac{\gamma_2^{'}}{\gamma_1^{'}+\gamma_2^{'}} < \frac{\gamma_2}{\gamma_1+\gamma_2} = S(2 \mid k)$. □

Lemma 2.1 : Suppose that $q_k(\mathbf{p}) = \prod_{m=1}^{J} p_m^{\theta_{mk}}$ ($\sum_m \theta_{mk} = 1$ for all $k = 1, 2, \ldots$) is a sequence of functions defined on $\{(p_1, p_2, \ldots, p_J) : 0 < p_1, p_2, \ldots, p_J < 1 \text{ and } \sum_m p_m = 1\}$, and $g(\mathbf{p})$ is another positive function defined on the same domain. Also, assume that as $k \to \infty$, $\boldsymbol{\theta}_k = (\theta_{1k}, \theta_{2k}, \ldots, \theta_{Jk})$ converges to $\mathbf{P} = (P_1, P_2, \ldots, P_J)$, where $\sum_m P_m = 1$ and $P_i > P_j$ for all $j \neq i$. Define a new sequence of functions ζ_k on the set $C = \{(p_1, p_2, \ldots, p_j) : \ p_i \geq p_j \ \forall j \neq i\}$, which is given by

$$\zeta_k(\mathbf{p}) = q_k^k(\mathbf{p})\ g(\mathbf{p}) \Big/ \int_C q_k^k(\mathbf{p})\ g(\mathbf{p})\ d\mathbf{p}.$$

Then, for every $\epsilon > 0$ and any $j \neq i$, there exists a $\delta > 0$ and a $k_0 \geq 1$ such that for all $k \geq k_0$, we have

$$\int_{C \cap \{1-\delta \leq p_j/p_i \leq 1\}} \zeta_k(\mathbf{p})\ d\mathbf{p} < \epsilon.$$

Proof of Lemma 2.1 : Consider the function $q(\mathbf{p}) = \prod_{m=1}^{J} p_m^{P_m}$ defined on the same domain as that of $q_k(\mathbf{p})$. Fixing the values of all p_m's such that $m \neq i, j$, it is easy to see that q is maximized when $p_j/p_i = P_j/P_i = 1 - \eta$ ($0 < \eta < 1$). One can also notice that q decreases with p_j/p_i when $p_j/p_i > 1 - \eta$ and increases when $p_j/p_i < 1 - \eta$. Now choose some $\delta < \eta/2$ and define the set $A_\delta = \{(p_1, p_2, \ldots, p_J) : 1 - \delta \leq p_j/p_i \leq 1\}$. On this set q is maximized at $\mathbf{p}^\delta = (p_1^\delta, p_2^\delta, \ldots, p_J^\delta)$, where $p_m^\delta = P_m$ for all $m \neq i, j$, $p_i^\delta = (P_i + P_j)/(2 - \delta)$ and $p_j^\delta = (1 - \delta)(P_i + P_j)/(2 - \delta)$, and

this maximum value is given by

$$\lambda(\mathbf{P}) = \Big(\prod_{\substack{m=1 \\ m \neq i,j}}^{J} P_m^{P_m} \Big) \; (P_i + P_j)^{P_i+P_j} (1-\delta)^{P_i} / (2-\delta)^{P_i+P_j} = M_1, \text{ say.}$$

Now, as $k \to \infty$, $\theta_{mk} \to P_m$ for all $m = 1, 2, \ldots, J$. Therefore, $\theta_{jk}/\theta_{ik} \to P_j/P_i$ (since $P_i > 0$), and because of the continuity of the function λ, $|\lambda(\boldsymbol{\theta}_k) - \lambda(\mathbf{P})| \to 0$. Hence, for every $\epsilon > 0$, one can always find some $k_1 \geq 1$ such that $\theta_{jk}/\theta_{ik} < 1 - \delta$ and $|\lambda(\boldsymbol{\theta}_k) - \lambda(\mathbf{P})| < \epsilon$ for all $k \geq k_1$. Next, note that for such values of $\boldsymbol{\theta}_k$, q_k is decreasing in p_j/p_i when $p_j/p_i \geq 1 - \delta$, and on the set A_δ, it has an upper bound $\lambda(\boldsymbol{\theta}_k)$. Therefore,

$$\sup_{\mathbf{p} \in A_\delta \cap C} q_k(\mathbf{p}) < M_1 + \epsilon, \; \forall k \geq k_1.$$

On the other hand, the function q is maximized at $\mathbf{p} = \mathbf{P}$, which is an interior point of C. Let us denote this value $\prod_{m=1}^{J} P_m^{P_m}$ by M_2. Clearly, $M_2 > M_1$, and because of the continuity of $\prod_{j=1}^{J} x_j^{r_j}$ in x_j's and r_j's, for every $\epsilon > 0$, it is possible to find ϵ_0 such that $\|\mathbf{p} - \mathbf{P}\| + \|\boldsymbol{\theta}_k - \mathbf{P}\| < \epsilon_0 \Rightarrow |\prod_{m=1}^{J} p_m^{\theta_{mk}} - \prod_{m=1}^{J} P_m^{P_m}| < \epsilon$. Here $\| \; \|$ denotes the usual Euclidean norm on R^J. Since $\boldsymbol{\theta}_k$ converges to $\mathbf{P}$, it is always possible to choose a ball $B \subset C$ of radius $\epsilon_1 < \epsilon_0$ around $\mathbf{P}$ such that for some $k_2 \geq 1$ and all $k > k_2$, we have

$$\inf_{\mathbf{p} \in B \cap C} q_k(\mathbf{p}) > M_2 - \epsilon.$$

Note that the above results hold for every epsilon $\epsilon > 0$. Choose an ϵ such that $(M_1 + \epsilon)/(M_2 - \epsilon) < t$ for some $t < 1$. Define

$$\alpha = \int_{A_\delta \cap C} g(\mathbf{p}) \; d\mathbf{p} \;\; \text{and} \;\; \beta = \int_{B \cap C} g(\mathbf{p}) \; d\mathbf{p}.$$

It is now quite easy to see that for all $k > k^* = max\{k_1, k_2\}$,

$$\int_{A_\delta \cap C} \zeta_k(\mathbf{p}) \; d\mathbf{p} < \int_{A_\delta \cap C} q_k^k(\mathbf{p}) \; g(\mathbf{p}) \Big/ \int_{C} q_k^k(\mathbf{p}) \; g(\mathbf{p}) \; d\mathbf{p} \;\; < \;\; \alpha t^k / \beta.$$

To arrive at the final result choose $k_0 > k^*$ such that $\alpha t^k/\beta < \epsilon$. □

Proof of Theorem 2.2 : Take $\theta_{mk} = t_{m_k}/k$ for $m = 1, 2, \ldots, J$, $g(\mathbf{p}) = \pi(\mathbf{p})$ and consider the sequence of functions q_k and ζ_k as described in Lemma 2.1. From the existing results [28], [7], we know that if $k \to \infty$ and $k/N \to 0$ as $N \to \infty$, under the assumption on continuity of f_js, $\boldsymbol{\theta}_k$ converges in probability to $\mathbf{P}$, the vector of true conditional probabilities (at $\mathbf{x}$) of different classes. Now, following the idea of exchangeability of p_i and p_j as used in the proof of Theorem 2.1, it is easy to see that

$$S(j \mid k)/S(i \mid k) = \int_C (p_j/p_i)^{k(\theta_{ik}-\theta_{jk})} \zeta_k(\mathbf{p})\ d\mathbf{p},$$

where ζ_k and C have the same meaning as in Lemma 2.1. For $0 < \delta < 1$, define $A_\delta = \{(p_1, p_2, \ldots, p_J) : 1 - \delta \leq p_j/p_i \leq 1\}$. Since $p_j/p_i \leq 1$ on C and $k(\theta_{ik} - \theta_{jk}) \xrightarrow{P} \infty$ as $k \to \infty$, for every $\epsilon, \lambda > 0$, it is possible to find $\delta > 0$ and $k_0 \geq 1$ (see Lemma 2.1) such that

$$P\Big\{ \int_{A_\delta \cap C} (p_j/p_i)^{k(\theta_{ik}-\theta_{jk})}\ \zeta_k(\mathbf{p})\ d\mathbf{p} < \int_{A_\delta \cap C} \zeta_k(\mathbf{p})\ d\mathbf{p} < \epsilon/2 \Big\} > 1 - \lambda/2 \ \ \forall\ k \geq k_0.$$

Again, note that on the set $A_\delta^c \cap C$, $(p_j/p_i)^{k(\theta_{ik}-\theta_{jk})}$ uniformly converges to 0 in probability. Therefore, for those same ϵ and λ, one can find some $k_1 \geq 1$ such that

$$P\left\{ \sup_{A_\delta^c \cap C} (p_j/p_i)^{k(\theta_{ik}-\theta_{jk})} < \epsilon/2 \right\} > 1 - \lambda/2 \ \ \forall\ k \geq k_1.$$

Hence, $P\{S(j \mid k)/S(i \mid k) < \epsilon\} > 1 - \lambda$ for all $k > max\{k_0, k_1\}$. This implies that $S(j \mid k) \xrightarrow{P} 0$ for all $j \neq i$ (since $S(i \mid k) < 1$). Now, the result $S(i \mid k) \xrightarrow{P} 1$ follows from the facts that $\sum_{j=1}^{J} S(j \mid k) = 1$ and J is finite. □

References

[1] Alpaydin, E. (1997) Voting over multiple condensed nearest neighbor. *Art. Intell. Review*, **11**, 115-132.

[2] Breiman, L., Friedman, J. H., Olshen, R. A. and Stone, C. J. (1984) *Classification and Regression Trees*. Wadsworth & Brooks, Monterrey, California.

[3] Breiman, L. (1996) Bagging predictors. *Machine Learning*, **24**, 123-140.

[4] Breiman, L. (1998) Arcing classifiers (with discussion). *Ann. Statist.*, **26**, 801-849.

[5] Chaudhuri, P. and Marron, J. S. (1999) SiZer for exploration of structures in curves. *J. Amer. Statist. Assoc.*, **94**, 807-823.

[6] Cooley, C.A. and S.N. MacEachern (1998) Classification via kernel product estimators. *Biometrika*, **85**, 823-833.

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

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

[9] Devijver, P. A. and Kittler, J. (1982) *Pattern Recognition: A Statistical Approach*. Prentice Hall, London.

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

[11] 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.*

[12] Friedman, J. H. (1996) Flexible metric nearest neighbor classification. *Tech. Rep., Dept. of Stat., Stanford University.*

[13] Friedman, J. H. (1997) On bias, variance, 0-1 loss, and the curse of dimensionality. *Data Mining and Knowledge Discovery*, **1**, 55-77.

[14] Friedman, J. H., Hastie, T. and Tibshirani, R. (2000) Additive logistic regression : a statistical view of boosting (with discussion). *Ann. Statist.*, **28**, 337-407.

[15] Fukunaga, K. and Hostetler, L. D. (1973) Optimization of k-nearest neighbor density estimates. *IEEE Trans. Info. Theory*, **19**, 320-326.

[16] Fukunaga, K. (1990) *Introduction to Statistical Pattern Recognition.* Academic Press, New York.

[17] Ghosh, A. K., Chaudhuri, P. and Sengupta, D. (2003) Multi-scale kernel discriminant analysis. *Proceedings of Fifth International Conference on Advances in Pattern Recognition*, Ed. D. P. Mukherjee and S. Pal, Allied Publishers, Calcutta, pp. 89-93.

[18] Ghosh, A. K., Chaudhuri, P. and Sengupta, D. (2004) Classification using kernel density estimates : multi-scale analysis and visualization. *Technometrics (Under revision).*

[19] Godtliebsen, F., Marron, J. S. and Chaudhuri, P. (2002) Significance in scale space for bivariate density estimation. *Journal of Computational and Graphical Statistics*, **11**, 1-22.

[20] Hart, P. E. (1968) The condensed nearest neighbor rule. *IEEE Trans. Info. Theory*, **14**, 515-516.

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

[22] Hastie, T., Tibshirani, R. and Friedman, J. H. (2001) *The elements of statistical learning : data mining, inference and prediction.* Springer Verlag.

[23] Ho, T. K., Hull, J. J. and Srihari, S. N. (1994) Decision combination in multiple classifier systems. *IEEE Trans. Pattern Analysis and Machine Intell.*, **16**, 66-75.

[24] 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.

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

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

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

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

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

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

[31] Mitra, P., Murthy, C. A. and Pal, S. K. (2002) Density based multiscale data condensation. *IEEE Trans. on Pattern Analysis and Machine Intelligence* , **24**, 734-747.

[32] Opitz, D. and Maclin, R. (1999) Popular ensemble methods : an empirical study. *J. Art. Intell. Research*, **11**, 169-198.

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

[34] Pal, S. K., Bandopadhyay, S. and Murthy, C. A. (1998) Genetic algorithms for generation of class boundaries. *IEEE Trans. Syst. Man. and Cybern.*, **28**, 816-828.

[35] Peterson, G. E. and Barney, H. L. (1952) Control methods used in a study of vowels. *J. Acoust. Soc. Amer.*, **24**, 175-185.

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

[37] Schapire, R. E., Fruend, Y., Bartlett, P. and Lee, W. (1998) Boosting the margin : a new explanation for the effectiveness of voting methods. *Ann. Statist.*, **26**, 1651-1686.

[38] Shalak, D. B. (1996) Prototype selections for composite nearest neighbor classifiers. *Ph.D. Thesis, Dept. of Computer Science, University of Massachusetts.*

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