June 1995

Pattern Recognition Letters

ELSEVIER

Pattern Recognition Letters 16 (1995) 601–614

A multistage generalization of the rank nearest neighbor classification rule

Subhash C. Bagui [a,*], Nikhil R. Pal [b,1]

[a] *Department of Mathematics and Statistics, The University of West Florida, Pensacola, FL 32514, USA*
[b] *Division of Computer Science, The University of West Florida, Pensacola, FL 32514, USA*

Received 19 June 1994; revised 13 December 1994

Abstract

We consider the problem of classifying an unknown observation from one of s ($\geqslant 2$) univariate classes (or populations) using a multi-stage left and right rank nearest neighbor (RNN) rule. We derive the asymptotic error rate (i.e., total probability of misclassification (TPMC)) of the m-stage univariate RNN (m-URNN) rule, and show that as the number of stages increases, the limiting TPMC of the m-stage univariate rule decreases. Monte Carlo simulations are used to study the behavior of the m-URNN rule and compare it with the conventional k-NN rule. Finally, we incorporate an extension of the m-URNN rule to multivariate observations with empirical results.

Keywords: Bayes error rate; Classification; Rank nearest neighbor; k-Nearest neighbor

1. Introduction

Statistical classifier design can be posed as follows: Let $\omega_1, \omega_2, \ldots, \omega_s$ denote s physically distinguishable classes (or populations), and let ω_i be distributed with probability density function $f(x \mid \omega_i)$ (more briefly, $f_i(x)$), $1 \leqslant i \leqslant s$, with $x \in \mathbb{R}^p$. Let ξ_i be the prior probability of class ω_i, and define $g(x) = \sum_{i=1}^{s} \xi_i f(x \mid \omega_i) = \sum_{i=1}^{s} \xi_i f_i(x)$ as the mixture of s populations. Let z be a random observation drawn according to g. The classification problem with respect to z is to label it as belonging to one of the s classes $\{\omega_i\}$. If $\eta(\omega_i \mid z)$ denotes the posterior probability that, given z, z comes from class ω_i, the Bayes rule states that

$$\eta(\omega_i \mid z) = \xi_i f(z \mid \omega_i)/g(z) = \xi_i f_i(z)/g(z); \tag{1}$$

and the Bayes decision rule (BDR) *is to simply maximize* $\eta(\omega_i \mid z)$:

$$\text{BDR:} \quad \text{decide } z \in \omega_i \Leftrightarrow \eta(\omega_i \mid z) \geqslant \eta(\omega_j \mid z), \quad j = 1, \ldots, s;\ j \neq i. \tag{2}$$

It is well known that BDR minimizes the expected total probability of misclassification (TPMC) (Johnson and Wichern, 1988). And if the risk in deciding $z \in \omega_i$ when $z \in \omega_j$ is equal to *one* for all wrong decisions and

* Corresponding author.
[1] On leave from Indian Statistical Institute, Calcutta, India.

SSDI 0167-8655(95)00009-7

zero for all correct decisions (the (0–1) loss matrix), BDR also minimizes the overall *risk* associated with classification of z. In this case, the optimal Bayes error rate (minimal TPMC) is sometimes called the Bayes risk. We assume the 0–1 loss matrix in this paper.

For $j = 1, 2, \ldots, n_i$, let x_{ij} be labeled as belonging to class ω_i, $1 \leqslant i \leqslant s$; we call $\{x_{ij}\}$ a set of *training* samples. n_i is the number of observations from class ω_i. Given the training data $\{x_{ij}\}$, there are two approaches to the implementation of approximate BDRs. In the *parametric* approach, assumptions about the $\{f(z \mid \omega_i)\}$ and a principle of inference such as maximum likelihood can be used to estimate the right-hand side (RHS) of Eq. (1), and hence the left-hand side (LHS) for use in Eq. (2). *Nonparametric* methods do not require distributional assumptions, and lead to direct estimates of the LHS of (1), and implementation of (2). The leading nonparametric method is the k-nearest neighbor (conventionally known as k-NN) rule, introduced by Fix and Hodges (1951). Their k-NN rule for $s = 2$ populations may be described as follows: Using a distance function $d(x_{ij}, z)$, order the distances $\{d(x_{ij}, z)\}$ for $j = 1, \ldots, n_i$ and $i = 1, 2$. For a fixed integer k, the k-NN rule assigns z to ω_1 if $k_1/n_1 > k_2/n_2$, where k_i is the number of observations from ω_i $(i = 1, 2)$ among the $k = k_1 + k_2$ observations nearest to z. Cover and Hart (1967) considered an k-NN rule which assigns z to the class ω_i $(i = 1, \ldots, s)$, if $k_i = \max_j \{k_j\}$, $k = \sum_{j=1}^{s} k_j$. They showed that for the 1-NN rule, bounds for the limiting error rate R satisfy $R^* \leqslant R \leqslant R^*(2 - (s/(s-1))R^*)$, where R^* is the "*minimum*" Bayes error rate (see (14) below). The Cover and Hart (1967) result requires conditions of the existence of an almost sure continuous density. Devroye (1981a) generalized the results of Cover and Hart (1967) for all distributions; his work strengthens the results of Wagner (1971) and Fritz (1975). Devroye (1981b) obtained the following upper bound on the asymptotic k-NN risk R_k:

$$R_k \leqslant (1 + a_k)R^*, \qquad a_k = \frac{\alpha\sqrt{k}}{k - 3.25}\left(1 + \frac{\beta}{\sqrt{k-3}}\right), \quad k \text{ odd}, \ k \geqslant 5,$$

and $\alpha = 0.3399$ and $\beta = 0.9749$ are universal constants. This bound is the best possible in a *certain* sense. For other aspects of the k-NN rule see (Wagner, 1971; Fritz, 1975; Devijver and Kittler, 1982).

For the special case of univariate populations ($p = 1$), Anderson (1966) proposed a nonparametric classification rule for two populations by ranking the training samples (not their distances from z) that was further investigated by Das Gupta and Lin (1980). These ranking rules are related to statistically equivalent blocks; see (Anderson, 1966; Gessaman, 1970). Bagui (1989, 1993) extended Das Gupta and Lin's work to $s > 2$ populations. The 1-stage ($m = 1$) Univariate Rank Nearest Neighbor (URNN) rule for s populations may be described as follows.

1-URNN Algorithm (Bagui, 1989, 1993)

Pool the observations $\{x_{ij}\text{'s}\}$ $(i = 1, \ldots, s;\ j = 1, \ldots, n_i)$ and z, and rank order them; then, (i) if both the immediate left-hand (LH) and right-hand (RH) neighbors of z belong to the same population, classify z to that population; (ii) if z is either the smallest or the largest observation, classify z into the population of its immediate rank nearest neighbor; (iii) if both the immediate LH and RH rank nearest neighbors of z belong to different populations, classify z into either population arbitrarily.

The asymptotic error rate of the 1-stage URNN (1-URNN) rule for s populations turns out to be exactly the same as the asymptotic error rate of the 1-NN rule of Cover and Hart (1967) for s populations (Bagui, 1989, 1993). In this article we introduce a multi-stage (m-stage) generalization of the 1-stage URNN rule with s populations and investigate its theoretical properties. The m-stage URNN (m-URNN) rule for s populations may be described as follows:

***m*-URNN Algorithm**

1. Sort training data $\{x_{ij}\}$ in ascending order to $\{\hat{x}_t,\ t = 1, 2, \ldots, N;\ N = \sum_{i=1}^{s} n_i\}$.
2. Fix $m \in \mathbb{N}^+$ a positive integer given $z \in \mathbb{R}$.

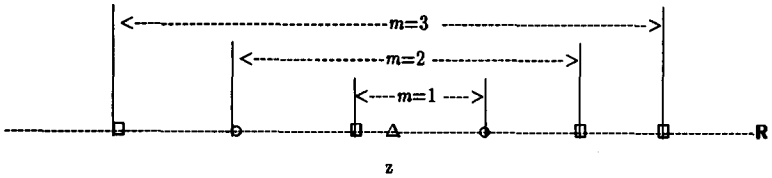

Fig. 1. The m-stage URNN rule.

3. If $z \notin [\hat{x}_1, \hat{x}_N]$, classify z with its rank nearest neighbor, exit.
4. If $z = \hat{x}_t$ for some t, classify z with the label of $\hat{x}_t$, exit.
5. $j \leftarrow 1$.
6. While $j \leqslant m$
   a. Find labels L_L and L_R of the jth LH and RH neighbors of z.
   b. If $L_L = L_R = L$, classify z in L; exit.
   c. If $j = m$, classify z arbitrarily to its LH or RH label; exit.
   d. $j \leftarrow j + 1$
   Wend (end while)

In step 6a, the jth LH and RH neighbors of z are the points in $\{\hat{x}_t\}$ which are, respectively, j positions to the left and right of z in the ordered data. As described, m bounds the number of (pairs) of LH and RH neighbors examined during the attempt to classify z. Accordingly, it is proper to call this algorithm the m-URNN rule.

The m-stage URNN is depicted schematically in Fig. 1. In this figure, $\Delta = z$ is the point to be labeled, and we show the six most immediate neighbors of z from the pooled training data, with, say ($\square$) denoting samples from class ω_i, and ($\bigcirc$) denoting samples from class ω_j. In Fig. 1, z is not labeled at stage 1 or stage 2, but at $m = 3$ both URNNs of z are ($\square$) $= \omega_i$, so the 3-stage URNN rule labels z as belonging to class ω_i. Applying the conventional k-NN rule to z using any metric on $\mathbb{R}$ would result in the label assignments shown in Table 1.

In Section 2 we derive the limiting TPMC, $R^{(m)}$ of the m-URNN rule in the s-population case and show that the limiting TPMC decreases as the number of stages employed increases, i.e., $R^{(m)} \leqslant R^{(m-1)}$. In Section 3, Monte Carlo results are reported to compare the performance of the m-stage URNN with the k-NN and $2k$-NN ($k = m$) rules using 3-population univariate mixtures. In Section 4 we discuss the possible extension of our m-URNN to the m-stage Mulivariate Rank Nearest Neighbor (m-MRNN) rule and compare it with the conventional k-NN rule on a real-world application. Finally, Section 5 contains some concluding remarks.

2. Limiting TPMC of the m-stage rule

For brevity, we denote the cumulative distribution of the ith class by F_i and its density by f_i ($i = 1, 2, \ldots, s$). Let $n = \min(n_1, n_2, \ldots, n_s)$. Let the probability of misclassification (PMC) at the mth stage (i.e., the probability of classifying Z as being from ω_l when in fact it is from ω_j ($l \neq j$)) be:

$$\alpha_{lj}^{(m)}(n_1, \ldots, n_s) = \mathrm{P}\big(\text{decide } Z \in \omega_l \mid Z \in \omega_j,\ l \neq j\big).$$

Table 1
k-NN rule labels for z in Fig. 1

k	1	2	3	4	5	6
label of z	$\square$	tie	$\bigcirc$	tie	$\square$	$\square$

Then the total probability of misclassification (TPMC) at stage m is:

$$R^{(m)}(n_1,\ldots,n_s) = \sum_{l=1}^{s}\sum_{\substack{j=1\\ j\neq l}}^{s} \xi_j \alpha_{lj}^{(m)}(n_1,\ldots,n_s). \tag{3}$$

Let the left and right RNNs of z at the mth stage be $U^{(m)}$ and $V^{(m)}$ respectively. We state two lemmas which are required to prove our main results. The proofs of these two lemmas are omitted as they essentially appear in (Das Gupta and Lin, 1980). The following lemma deals with the existence of m-stage left and right RNNs of z.

Lemma 2.1. *If $m/n \to 0$ as $n \to \infty$, the probability that there are at least m observations to the left of Z and at least m observations to the right of Z in the pooled sample converges to 1 as $n \to \infty$.*

The next lemma shows almost sure (a.s.) convergence of $U^{(m)}$ and $V^{(m)}$ to Z as $n \to \infty$.

Lemma 2.2. *If $m/n \to 0$ and Z has density f_l, then*

$$U^{(m)},\ V^{(m)} \xrightarrow{\text{a.s.}} Z \quad \text{as } n \to \infty.$$

Next, we define $\phi_l^{(m)}(Z;\ X_{ij},\ i=1,\ldots,s;\ j=1,\ldots,n_i)$ as

$$\phi_l^{(m)} = \begin{cases} 1, & \text{if both } U^{(m)} \text{ and } V^{(m)} \text{ are from the } l\text{th population, or if } z \text{ becomes an extreme} \\ & \text{observation at the } m\text{th stage and its RNN is from the } l\text{th population;} \\ \frac{1}{2}, & \text{if either one of } U^{(m)} \text{ or } V^{(m)} \text{ is from the } l\text{th population;} \\ 0, & \text{otherwise.} \end{cases} \tag{4}$$

Let A_i be the event that both $U^{(i)}$ and $V^{(i)}$ exist at the ith stage. Since the steps of proofs of 1-URNN limiting risk are needed to derive the limiting m-URNN error rate, we first obtain the limiting TPMC of the 1-URNN risk. The conditional probability that Z is in π_l, given $Z=z$, using the 1-URNN rule is given by

$$\Pi_l^{(1)}(z;\ n_1,\ldots,n_s) = \mathrm{E}\big(\phi_l^{(1)} \mid Z=z\big) = \mathrm{E}\big(\phi_l^{(1)} I_{A_1} \mid Z=z\big) + \mathrm{E}\big(\phi_l^{(1)} I_{A_1^c} \mid Z=z\big) \tag{5}$$

where in (5) $I_{[\cdot]}$ stands for the indicator function.

In subsequent discussions, when there is no fear of confusion, we use z instead of $Z=z$ when conditioning by $Z=z$. Lemma 2.1 implies that, asymptotically, the nonexistence of the left and right RNNs of z is impossible.

Thus, the second expectation in (5) satisfies

$$\mathrm{E}\big(\phi_l^{(1)} I_{A_1^c} \mid z\big) \leqslant \mathrm{P}\big(A_1^c \mid z\big) \to 0, \quad n \to \infty. \tag{6}$$

Now we express the first expectation in (5) as follows:

$$\begin{aligned} \mathrm{E}\big(\phi_l^{(1)} I_{A_1} \mid z\big) &= \mathrm{P}\big(\phi_l^{(1)} = 1 \cap A_1 \mid z\big) + \tfrac{1}{2}\mathrm{P}\big(\phi_l^{(1)} = \tfrac{1}{2} \cap A_1 \mid z\big) \\ &= \mathrm{E}\big(\mathrm{P}^{(ll)}(U, V, z)\big) + \tfrac{1}{2}\mathrm{E}\big(\mathrm{P}^{(l0)}(U, V, z)\big), \end{aligned} \tag{7}$$

where

$$\mathrm{P}^{(ll)}(u, v, z) = \mathrm{P}\big(\phi_l^{(1)} = 1 \mid u, v, z\big) \quad \text{and} \quad \mathrm{P}^{(l0)}(u, v, z) = \mathrm{P}\big(\phi_l^{(1)} = \tfrac{1}{2} \mid u, v, z\big).$$

$\mathrm{P}^{(ll)}(u, v, z)$ is the conditional probability that both RNNs of z at the 1st stage are from the lth population, and $\mathrm{P}^{(l0)}(u, v, z)$ is the conditional probability that one of the RNNs of z at the 1st stage is from the lth population.

Note that

$$P^{(ll)}(u, v, z) = \frac{C_l(n_1, \ldots, n_s)}{B(n_1, \ldots, n_s)}, \qquad P^{(l0)}(u, v, z) = \frac{C_{l0}(n_1, \ldots, n_s)}{B(n_1, \ldots, n_s)}, \tag{8,9}$$

where for all $l = 1, \ldots, s$,

$$C_l(n_1, \ldots, n_s) = n_l(n_l - 1)\left[1 - (F_l(v) - F_l(u))\right]^{n_l - 2} \prod_{\substack{i=1 \\ i \neq l}}^{s} \left[1 - (F_i(v) - F_i(u))\right]^{n_i} f_l(u) f_l(v), \tag{10}$$

$$C_{l0}(n_1, \ldots, n_s) = \sum_{\substack{j=1 \\ j \neq l}}^{s} n_l n_j \left[1 - (F_l(v) - F_l(u))\right]^{n_l - 1} \left[1 - (F_j(v) - F_j(u))\right]^{n_j - 1}$$

$$\times \prod_{\substack{i=1 \\ i \neq j, i \neq l}}^{s} \left[1 - (F_i(v) - F_i(u))\right]^{n_i} \left[f_l(u) f_j(v) + f_j(u) f_l(v)\right], \tag{11}$$

and

$$B(n_1, \ldots, n_s) = \sum_{l=1}^{s} C_l(n_1, \ldots, n_s) + \sum_{l=1}^{s} C_{l0}(n_1, \ldots, n_s). \tag{12}$$

Let $p_i = \lim_{n \to \infty} n_i / \sum_{i=1}^{s} n_i$ and clearly we see $p_i \overset{\text{a.s.}}{=} \xi_i$ as $n \to \infty$, and assume $0 < p_i < 1$ $(i = 1, \ldots, s)$. Now we state a theorem from (Bagui, 1993) without proof:

Theorem 2.1. *Suppose z is a continuity point of f_i $(i = 1, \ldots, s)$. Then, for almost all z (under f_i, $i = 1, \ldots, s$), the asymptotic conditional probability of classifying z to ω_l, using the 1-URNN rule is given by*

$$\Pi_l^{(1)}(z) = \lim_{n \to \infty} \Pi_l^{(1)}(z; n_1, \ldots, n_s) = \nu_l(z) + \tfrac{1}{2}\nu_{l0}(z),$$

where

$$\nu_l(z) = p_l^2 f_l^2(z) \Big/ \left[\sum_{i=1}^{s} p_i f_i(z)\right]^2 \quad \text{and} \quad \nu_{l0}(z) = 2 p_l f_l(z) \sum_{\substack{i=1 \\ i \neq l}}^{s} p_i f_i(z) \Big/ \left[\sum_{i=1}^{s} p_i f_i(z)\right]^2.$$

In view of Theorem 2.1, $p_i \overset{\text{a.s.}}{=} \xi_i$ and (1) we have

$$\Pi_l^{(1)}(z) = \nu_l(z) + \tfrac{1}{2}\nu_{l0}(z) = \eta^2(\omega_l \mid z) + \eta(\omega_l \mid z)(1 - \eta(\omega_l \mid z)) = \eta(\omega_l \mid z).$$

Now the limiting PMC's of the first-stage RNN are derived as follows:

$$\alpha_{lj}^{(1)} = \lim_{n \to \infty} P(\text{decide } Z \in \omega_l \mid Z \in \omega_j,\ l \neq j) = \int \Pi_l^{(1)}(z) f_j(z)\, dz = \int \eta(\omega_l \mid z) f_j(z)\, dz. \tag{13}$$

The minimal Bayes error rate (for the 0–1 loss matrix case) is:

$$R^* = \int \min\left(\sum_{i \neq 1} \xi_i f_i(z), \sum_{i \neq 2} \xi_i f_i(z), \ldots, \sum_{i \neq s} \xi_i f_i(z) \right) dz$$

$$= \int \min\{(1 - \eta(\omega_1 \mid z)), (1 - \eta(\omega_2 \mid z)), \ldots (1 - \eta(\omega_s \mid z))\} g(z)\, dz. \tag{14}$$

When the training sample is from a mixture of s populations, we may take $p_l = \xi_l$ and then using (1), (3) and (13) we have the limiting TPMC of the 1st-stage rule in the following form:

$$R^{(1)} = \sum_{l=1}^{s} \sum_{\substack{j=1 \\ j\neq l}}^{s} \xi_j \alpha_{lj}^{(1)} = \int \sum_{l=1}^{s} \sum_{\substack{j=1 \\ j\neq l}}^{s} \eta(\omega_l \mid z)\eta(\omega_j \mid z) g(z)\, \mathrm{d}z$$

$$= \mathrm{E}\left(\sum_{l=1}^{s} \sum_{\substack{j=1 \\ j\neq l}}^{s} \eta(\omega_l \mid z)\eta(\omega_j \mid z) \right). \tag{15}$$

It is clear from (15) that the limiting TPMC for the 1-URNN rule in s populations is exactly the same as the limiting risk of the 1-NN rule of Cover and Hart (1967) in the s-population case. From this we deduce the following theorem.

Theorem 2.2 (Bagui, 1993). *Suppose the training sample is from a mixture of populations ω_l $(l = 1, \ldots, s)$ with ξ_l the prior probability of ω_l. Then $R^{(1)}$ has the following bounds:*

$$R^* \leqslant R^{(1)} \leqslant R^* \left(2 - \frac{s}{s-1} R^* \right)$$

where R^* is the Bayes error rate defined in (14).

Now we will derive the limiting TPMC of the m-stage RNN rule.

Let $\Pi_l^{(m)}(z; n_1, \ldots, n_s)$ be the conditional probability that the m-URNN rule classifies Z in ω_l, given $Z = z$. By the definition of $\phi_l^{(m)}$ (see (4)), we have

$$\begin{aligned}
\Pi_l^{(m)}(z; n_1, \ldots, n_s) &= \mathrm{P}(\text{both } U^{(m)} \text{ and } V^{(m)} \text{ are from the } l\text{th population}) \\
&\quad + \tfrac{1}{2}\mathrm{P}(\text{one of } U^{(m)} \text{ and } V^{(m)} \text{ is from the } l\text{th population}) \\
&= \mathrm{P}\left(\phi_l^{(1)} = 1 \mid z\right) + \mathrm{P}\left(\phi_l^{(1)} = \tfrac{1}{2},\ \phi_l^{(2)} = 1 \mid z\right) + \cdots + \mathrm{P}\left(\phi_l^{(1)} = \tfrac{1}{2}, \ldots, \phi_l^{(m)} = 1 \mid z\right) \\
&\quad + \tfrac{1}{2}\mathrm{P}\left(\phi_l^{(1)} = \tfrac{1}{2}, \ldots, \phi_l^{(m)} = \tfrac{1}{2} \mid z\right) \\
&= \mathrm{P}\left(\phi_l^{(1)} = 1 \mid z\right) + \sum_{i=2}^{m} \mathrm{P}\left(\phi_l^{(1)} = \tfrac{1}{2}, \ldots, \phi_l^{(i-1)} = \tfrac{1}{2},\ \phi_l^{(i)} = 1 \mid z\right) \\
&\quad + \tfrac{1}{2}\mathrm{P}\left(\phi_l^{(1)} = \tfrac{1}{2}, \ldots, \phi_l^{(m)} = \tfrac{1}{2} \mid z\right).
\end{aligned} \tag{16}$$

Again,

$$\begin{aligned}
\mathrm{P}\left(\phi_l^{(1)} = \tfrac{1}{2}, \ldots, \phi_l^{(i)} = 1 \mid z\right) &= \mathrm{P}\left(\phi_l^{(1)} = \tfrac{1}{2} \mid z\right) \prod_{j=2}^{i-1} \mathrm{P}\left(\phi_l^{(j)} = \tfrac{1}{2} \mid \phi_l^{(j-1)} = \tfrac{1}{2}, \ldots, \phi_l^{(1)} = \tfrac{1}{2},\ z\right) \\
&\quad \times \mathrm{P}\left(\phi_l^{(i)} = 1 \mid \phi_l^{(i-1)} = \tfrac{1}{2}, \ldots, \phi_l^{(1)} = \tfrac{1}{2},\ z\right)
\end{aligned} \tag{17}$$

and

$$\mathrm{P}\left(\phi_l^{(1)} = \tfrac{1}{2},\ \phi_l^{(2)} = \tfrac{1}{2}, \ldots, \phi_l^{(m)} = \tfrac{1}{2} \mid z\right) = \mathrm{P}\left(\phi_l^{(1)} = \tfrac{1}{2} \mid z\right) \prod_{j=2}^{m} \mathrm{P}\left(\phi_l^{(j)} = \tfrac{1}{2} \mid \phi_l^{(j-1)} = \tfrac{1}{2}, \ldots, \phi_l^{(1)} = \tfrac{1}{2},\ z\right). \tag{18}$$

We shall show that

$$\lim_{n\to\infty} \mathrm{P}\left(\phi_l^{(i)} = 1 \mid \phi_l^{(i-1)} = \tfrac{1}{2}, \ldots, \phi_l^{(1)} = \tfrac{1}{2},\ Z\right) = \lim_{n\to\infty} \mathrm{P}\left(\phi_l^{(1)} = 1 \mid Z\right) = \nu_l(Z), \quad \text{a.s.} \tag{19}$$

and

$$\lim_{n\to\infty} \mathrm{P}\big(\phi_l^{(j)} = \tfrac{1}{2} \mid \phi_l^{(j-1)} = \tfrac{1}{2}, \ldots, \phi_l^{(1)} = \tfrac{1}{2}, Z\big) = \lim_{n\to\infty} \mathrm{P}\big(\phi_l^{(1)} = \tfrac{1}{2} \mid Z\big) = \nu_{l0}(Z), \quad \text{a.s.} \tag{20}$$

where ν_l and ν_{l0} are given in Theorem 2.1.

Thus, using (16) to (20) we arrive at

$$\Pi_l^{(m)}(z) = \nu_l(z) \sum_{i=1}^{m-1} \nu_{l0}^i(z) + \tfrac{1}{2}\nu_{l0}^m(z). \tag{21}$$

Assume $\phi_l^{(1)} = \frac{1}{2}$. Remove the observations corresponding to $U^{(1)}$ and $V^{(1)}$ from the pooled ordered training sample. Denote the remaining $(\sum_{i=1}^{s} n_i - 2)$ observations by $X_{ij}^{(2)}$.

Lemma 2.3. *Given $Z = z$, $\phi_l^{(1)} = \frac{1}{2}$, $U^{(1)} = u_1$ and $V^{(1)} = v_1$ (suppose one of u_1 and v_1 is from the lth population), then the conditional distributions of $X_{lj}^{(2)}$ $(j = 1, \ldots, n_l - 1)$ and $X_{kj}^{(2)}$ $(k \neq l)$ $(j = 1, \ldots, n_k - 1)$ satisfy:*

(i) *$X_{ij}^{(2)}$ are conditionally mutually independent.*

(ii) *The density of $X_{lj}^{(2)}$ is*

$$f_l^{(2)}(x) = f_l(x)/\big[1 - \big(F_l(v_1) - F_l(u_1)\big)\big] \quad \text{on } [u_1, v_1]^c. \tag{22}$$

(iii) *The density of $X_{kj}^{(2)}$ is*

$$f_k^{(2)}(x) = f_k(x)/\big[1 - \big(F_k(v_1) - F_k(u_1)\big)\big] \quad \text{on } [u_1, v_1]^c. \tag{23}$$

Lemma 2.3 can be extended in a similar manner inductively. Let $\phi_l^{(m)} = \frac{1}{2}$ $(m = 1, 2, \ldots, i-1)$. Delete the observations corresponding to $U^{(m)}$ and $V^{(m)}$ $(m = 1, 2, \ldots, i-1)$ and denote the remaining $(\sum_{i=1}^{s} n_i - 2(i-1))$ observations by $X_{\alpha\beta}^{(i)}$.

Lemma 2.4. *Given $Z = z$, $U^{(m)} = u_m$, $V^{(m)} = v_m$ and $\phi_l^{(m)} = \frac{1}{2}$ $(m = 1, 2, \ldots, i-1)$, the conditional distributions of $X_{l\beta}^{(i)}$ $(\beta = 1, 2, \ldots, n_l - i + 1)$ and $X_{k\beta}^{(i)}$ $(k \neq l)$ $(\beta = 1, 2, \ldots, n_l - i_k + 1$, where i_k is such that $\sum_{k\neq l} i_k = i)$, satisfy:*

(i) *$X_{\alpha\beta}^{(i)}$'s are mutually independent.*

(ii) *The density of $X_{l\beta}^{(i)}$ is*

$$f_l^{(i)}(x) = f_l^{(i-1)}(x)/\big[1 - \big(F_l^{(i-1)}(v_{i-1}) - F_l^{(i-1)}(u_{i-1})\big)\big] \quad \text{on } [u_{i-1}, v_{i-1}]^c, \tag{24}$$

where $F_l^{(i-1)}$ is c.d.f. corresponding to $f_l^{(i-1)}$ defined inductively.

(iii) *The density of $X_k^{(i)}\beta$ is*

$$f_k^{(i)}(x) = f_k^{(i-1)}(x)/\big[1 - \big(F_k^{(i-1)}(v_{i-1}) - F_k(u_{i-1})\big)\big] \quad \text{on } [u_{i-1}, v_{i-1}]^c, \tag{25}$$

where $F_k^{(i-1)}$ is c.d.f. corresponding to $f_k^{(i-1)}$, defined inductively.

Proof. The proofs of the above two lemmas are similar to the proofs of Lemmas 3.1 and 3.2 of (Das Gupta and Lin, 1980), and hence they are omitted. □

Note from (19) that

$$\mathrm{P}\big(\phi_l^{(i)} = 1 \mid \phi_l^{(i-1)} = \tfrac{1}{2}, \ldots, \phi_l^{(1)} = \tfrac{1}{2}, z\big) = \mathrm{E}\big(\mathrm{P}\big(\phi_l^{(i)} = 1 \mid \phi_l^{(i-1)} = \tfrac{1}{2}, \ldots, \phi_l^{(1)} = \tfrac{1}{2}; U^{(i)}, V^{(i)}, z\big)\big) \tag{26}$$

with

$$\mathrm{P}\big(\phi_l^{(i)} = 1 \mid \phi_l^{(i-1)} = \tfrac{1}{2}, \ldots, \phi_l^{(1)} = \tfrac{1}{2}, U^{(i)}, V^{(i)}, z\big) = C_l^{(i)}/B^{(i)}, \tag{27}$$

where $C_l^{(i)}$ and $B^{(i)}$ can be obtained from C_l and B (given in (10) and (12) respectively) by replacing n_l, u, v, f_l by $n_l - i + 1$, u_i, v_i, $f_l^{(i)}$ respectively.

Since U_i, $V_i \overset{\text{a.s.}}{\to} z$ (by Lemma 2.2), using Lemmas 2.3, 2.4, and the continuity of f_i, we have for $f_l^{(i)}(u_i)$, and $f_l^{(i)}(v_i) \overset{\text{a.s.}}{\to} f_l(z)$ a.s. Therefore, the limiting value of $C_l^{(i)}/B^{(i)}$ would be C_l/B. So by (26), (27) and the Dominated Convergence theorem (DCT), we get

$$\lim_{n\to\infty} \mathrm{P}\big(\phi_l^{(i)} = 1 \mid \phi_l^{(i-1)} = \tfrac{1}{2}, \ldots, \phi_l^{(1)} = \tfrac{1}{2}, Z\big) = C_l/B = \nu_l(Z), \quad \text{a.s.} \tag{28}$$

and by similar arguments one can show that

$$\lim_{n\to\infty} \mathrm{P}\big(\phi_l^{(i)} = \tfrac{1}{2} \mid \phi_l^{(i-1)} = \tfrac{1}{2}, \ldots, \phi_l^{(1)} = \tfrac{1}{2}, Z\big) = \nu_{l0}(Z), \quad \text{a.s.} \tag{29}$$

where $\nu_l(Z)$, and $\nu_{l0}(Z)$ are given in Theorem 2.1.

Now we state our main theorem.

Theorem 2.3. *Let z be a continuity point of f_i $(i = 1, \ldots, s)$. Then the limiting conditional probability of classifying z as belonging to ω_l using the m-URNN is given by*

$$\Pi_l^{(m)}(z) = \lim_{n\to\infty} \Pi_l^{(m)}(z; n_1, \ldots, n_s) = \nu_l(z) \sum_{i=0}^{m-1} \nu_{l0}^i(z) + \tfrac{1}{2}\nu_{l0}^m(z).$$

Proof. Introducing the set A_1 as in Theorem 2.1 and arguing as for (5) to (7) we get the result in view of (16) to (21). □

The limiting PMC's of the m-URNN rule using the DCT are given by

$$\alpha_{lj}^{(m)} = \lim_{n\to\infty} \mathrm{P}\big(m\text{-URNN rule decides } Z \in \omega_l \mid Z \in \omega_j\big) = \int \Pi_l^{(m)}(z) f_j(z)\, \mathrm{d}z. \tag{30}$$

In the limiting case we may take $p_i = \xi_i$ $(i = 1, 2, \ldots, s)$ so that the limiting TPMC for the m-URNN rule is given by

$$R^{(m)} = \sum_{l=1}^{s} \sum_{\substack{j=1 \\ j\neq l}}^{s} \xi_j \alpha_{lj}^{(m)}. \tag{31}$$

Finally, we show that $R^{(m)}$ is monotone decreasing with m.

Theorem 2.4. *For $R^{(m)}$ as defined in* (31), *we have*:

(i) $R^{(m)} \leqslant R^{(m-1)}$, $m = 2, 3, 4, \ldots$, *and*

(ii) $R^* \leqslant R^{(\infty)}$, *where R^* is given in* (14) *and $R^{(\infty)}$ is obtained from* (31).

Proof. Using (1), (21), (30), (31), and after a great deal of simplification, one will have

$$R^{(m)} - R^{(m-1)} = -2^{m-2} \int \sum_{l=1}^{s} \eta(\omega_l \mid z)^{m-1} \big(1 - \eta(\omega_l \mid z)\big)^{m-2} \big(1 - \eta(\omega_l \mid z)\big)^2 \big(1 - 2\eta(\omega_l \mid z)\big) g(z)\, \mathrm{d}z.$$

Now using the fact that $-(1 - \eta(\omega_l \mid z))^2 \leqslant -(1 - 2\eta(\omega_l \mid z))$ in the above, we get

$$R^{(m)} - R^{(m-1)} \leqslant -2^{m-2} \int \sum_{l=1}^{s} \eta(\omega_l \mid z)^{m-1} \big(1 - \eta(\omega_l \mid z)\big)^{m-2} \big(1 - 2\eta(\omega_l \mid z)\big)^2 g(z)\, \mathrm{d}z,$$

which implies $R^{(m)} \leqslant R^{(m-1)}$.

From (30) and (31) as $m \to \infty$, we have

$$R^{(\infty)} = \int \sum_{l=1}^{s} \sum_{\substack{j=1 \\ j \neq l}}^{s} \Pi_l^{(\infty)} \xi_j f_j(z)\, \mathrm{d}z$$

$$= \int \left[\Pi_1^{(\infty)} \sum_{j \neq 1}^{s} \xi_j f_j(z) + \Pi_2^{(\infty)} \sum_{j \neq 2}^{s} \xi_j f_j(z) + \cdots + \Pi_s^{(\infty)} \sum_{j \neq s}^{s} \xi_j f_j(z) \right] \mathrm{d}z$$

$$\geqslant \int \min\left(\sum_{j \neq 1}^{s} \xi_j f_j(z), \sum_{j \neq 2}^{s} \xi_j f_j(z), \ldots, \sum_{j \neq s}^{s} \xi_j f_j(z) \right) \mathrm{d}z = R^*. \qquad \square$$

Remark 2.1. Theorem 2.4 shows that as the number of stages (m) increases, the limiting error rate of the m-stage RNN rule decreases, which agrees with our intuition. Thus the m-stage rule is just an improvement over the previous stage.

3. Monte Carlo simulation for m-URNN rule

This section investigates the performance of the m-URNN rule empirically and compares it with that of the k-NN rule. Our first experiments used random training samples $\{x_{ij}\}$ of equal size ($n_1 = n_2 = n_3 = 50$) generated from equiprobable mixtures of triples of univariate distributions, namely (normal, normal, normal), (lognormal, lognormal, lognormal), (gamma, gamma, gamma), (Weibull, Weibull, Weibull) and (gamma, Weibull, gamma). For each mixture we then drew 1000 random observations from each of ω_1, ω_2, and ω_3. The 150 x_{ij}'s were then used to classify these 3000 points using the m-URNN and m-NN rules, varying m from 1 to 5. The proportions among the 3000 z's that were misclassified by the m-URNN and k-NN ($k = m$) rules are given in Tables 2, 3, and 4. Entries in the mth row of each of these three tables correspond to the situation where all 3000 sample points have been classified by the m-URNN. We remind readers that this means that all stages up to and including m have been used, as described in Section 1. For the m-NN rule, of course, each row of these tables corresponds to the fixed number of m of nearest neighbors used as shown in column 1.

Tables 2–4 exhibit the behavior of the m-URNN rule across changes in the shape and mean separations $\{|\mu_i - \mu_j|\}$ of the components of g. Observe that the mean separations of A, B and C in Tables 2 and 3 increase from 1 to 2 to 3 while the variance in all cases remains fixed (at 1). In A, mean separation and standard deviations are equal, and error rates hover around 45%, regardless of m or $f(x \mid \omega_i)$. As $|\mu_i - \mu_j|$ increases to the right in Tables 2 and 3, the error rate drops; as m increases, the error rate also drops. Both of these trends

Table 2
Error rates (%) for the m-URNN and m-NN rules: equiprobable mixtures of 3 normals

m	A		B		C	
	m-URNN	m-NN	m-URNN	m-NN	m-URNN	m-NN
1	51.1	50.3	29.7	29.0	12.8	12.8
2	47.2	50.4	26.3	29.7	12.1	13.1
3	45.5	48.3	23.7	27.3	11.8	11.4
4	44.9	48.7	23.8	27.1	11.7	11.8
5	44.7	45.3	23.8	23.1	11.7	12.0

A = $(N(0, 1) + N(1, 1) + N(2, 1))/3$; B = $(N(0, 1) + N(2, 1) + N(4, 1))/3$; C = $(N(0, 1) + N(3, 1) + N(6, 1))/3$
$N(a, b)$ = Normal distribution with mean a and standard deviation b

Table 3
Error rates (%) for the m-URNN and m-NN rules: equiprobable mixtures of 3 lognormals

m	A		B		C	
	m-URNN	m-NN	m-URNN	m-NN	m-URNN	m-NN
1	48.9	47.9	36.6	35.9	12.6	12.6
2	45.6	48.9	33.9	36.7	9.4	12.4
3	44.9	46.2	32.4	34.5	9.4	9.7
4	45.4	45.7	32.5	34.4	9.0	10.2
5	45.4	44.5	32.1	32.5	8.9	10.0

A $= (L(0, 1) + L(1, 1) + L(2, 1))/3$; B $= (L(0, 1) + L(2, 1) + L(4, 1))/3$; C $= (L(0, 1) + L(3, 1) + L(6, 1))/3$
$L(a, b)$ = Lognormal distribution (with mean and standard deviation of the underlying normal distribution being a and b respectively)

agree with theory. Table 4 lists results of using the m-URNN rule on mixtures whose components have Gamma and/or Weibull distributions. The same trends and remarks just noted apply to Table 4. Note that C, Table 4 is a mixture of two Gammas with a Weibull that has non-uniform mean separations. In summary, Tables 2–4 show that the m-URNN and m-NN rules yield quite similar, very comparable results on equiprobable mixtures of univariate distributions. Excluding ties, 34 of the 43 unequal error rate pairs in these three tables (about 79.1% of the tests) favor the m-URNN rule; but the differences in error rates are not significant enough to claim a great advantage for m-URNN. Nonetheless, this indicates that m-URNN is a bonafide competitor to the standard m-NN rule for classifier design on a wide variety of mixtures.

In Tables 2–4 we compared m-URNN with m-NN rules. One might argue that m-URNN should be compared to $2m$-NN rules as the m-URNN rule might use $2m$ data points. The intuitive justification for not doing this lies in the fact that m-URNN can, in the worst case, use $2m$ data points and in the best case decision may be made only with two data points. On an average, decision is expected to be made with $m/2$ neighbors on either side, i.e., m-URNN is expected to use on an average roughly m neighbors only. However, for completeness we compare in Table 5, the $2m$-NN rule with the m-URNN rule as an illustration, which also shows comparable results. Results of Table 5 correspond to the mixtures used in Table 3. For other mixtures reported in Tables 2 and 4, one can view the behavior of m-URNN and $2m$-NN rules (for $m \leqslant 2$) from Tables 2 and 4 itself.

4. Generalization to the multivariate case and an empirical study

Multivariate observations cannot be ranked uniquely because $x \in \mathbb{R}^p$ can at best be partially ordered. Consequently, it is difficult to make a natural extension of the m-URNN rule to multivariate observations. For

Table 4
Error rates (%) for the m-URNN and m-NN rules: equiprobable mixtures of 3 distributions

m	A		B		C	
	m-URNN	m-NN	m-URNN	m-NN	m-URNN	m-NN
1	41.1	41.6	51.3	51.2	40.6	40.1
2	38.1	41.1	49.8	52.4	36.4	39.3
3	37.1	38.4	48.7	50.4	35.6	36.3
4	36.7	37.9	48.5	50.1	34.9	35.3
5	36.6	36.8	48.3	48.7	34.9	35.5

A $= (G(1, 1) + G(3, 1) + G(5, 1))/3$; B $= (W(1, 1) + W(3, 1) + W(5, 1))/3$; C $= (G(1, 1) + W(2, 1) + G(4, 1))/3$
$G(a, b)$ = Gamma distribution with shape and scale parameters a and b respectively
$W(a, b)$ = Weibull distribution with shape and scale parameters a and b respectively

Table 5
Error rates (%) for the m-URNN and $2m$-NN rules: equiprobable mixtures of 3 lognormals.

m	A		B		C	
	m-URNN	$2m$-NN	m-URNN	$2m$-NN	m-URNN	$2m$-NN
1	48.9	48.9	36.6	36.7	12.6	12.4
2	45.6	45.7	33.9	34.4	9.4	10.2
3	44.9	44.5	32.4	32.4	9.4	9.5
4	45.4	44.7	32.5	31.8	9.0	9.7
5	45.4	43.8	32.1	31.5	8.9	9.9

A = $(L(0, 1)+L(1, 1)+L(2, 1))/3$; B = $(L(0, 1)+L(2, 1)+L(3, 1))/3$; C = $(L(0, 1)+L(3, 1)+L(6, 1))/3$.

literature on ordering multivariate observations see (Anderson, 1966; Barnett, 1976). But here we propose a rule which classifies multivariate observations using the m-URNN rule first on each feature; and then integrates the feature-wise results to arrive at a final labeling decision for each multivariate observation. The rule is specified as follows: suppose we have s p-variate ($p \geqslant 1$) populations $\omega_1, \omega_2, \ldots, \omega_s$, and let $(x_1^{(l)}, x_2^{(l)}, \ldots, x_{n_l}^{(l)})$ be the training data from the lth population ω_l with sample size n_l, where $x_j^{(l)} = (x_{j1}^{(l)}, x_{j2}^{(l)}, \ldots, x_{jp}^{(l)})^T$ for all $j = 1, 2, \ldots, n_l$ and $l = 1, 2, \ldots, s$, and T stands for the transpose of a vector. Let $z = (z_1, z_2, \ldots, z_p)^T$ be an unknown observation to be classified into one of the s populations. First we classify z_k for all $k = 1, 2, \ldots, p$ by applying the m-URNN rule using $\{x_{jk}^{(l)}\}$ for $j = 1, 2, \ldots, n_l$ and $l = 1, 2, \ldots, s$ as training data. Recalling $\phi_l^{(m)}$ of Eq. (4), we define $\phi_{lk}^{(m)}$ to be 1 or $\frac{1}{2}$ or zero when z_k is classified to the lth population ω_l, z_k is randomized between the lth and jth population ($j \neq l$), and not classified to ω_l respectively. Define $m_l = \sum_{k=1}^{p} \phi_{lk}^{(m)}$. Thus m_l can be viewed as the feature-wise total number of counts to classify z in favor of the class ω_l.

m-stage Multivariate Rank Nearest Neighbor (m-MRNN) rule

m-MRNN(a): If $m_i = m^*$ where m^* is the unique maximum of $\{m_l: 1 \leqslant l \leqslant s\}$, then classify z to the population ω_i.

m-MRNN(b): If $m^* = m_{i_1} = m_{i_2} = \cdots = m_{i_J}$, then classify z to ω_i with probability of being correct equal to $1/J$ for $i = i_1, i_2, \ldots, i_J$.

In other words, we poll the *votes* (cast by the m-URNN rule in each feature) for each of the s possible labels which might be assigned to z, and classify z using a simple majority scheme. Ties in m-URNN(b) are resolved arbitrarily. We shall call this algorithm the m-stage Multivariate Rank Nearest Neighbor (m-MRNN) rule. The term m-stage is appropriate because for $p = 1$, the proposed rule reduces to the m-URNN rule.

To see how the m-MRNN rule performs on multivariate data, we use Anderson's IRIS data (Johnson and Wichern, 1988) as an experimental data set. Let S be the IRIS data. S contains 50 (labeled) vectors in 4-dimensional space for each of the 3 classes of the IRIS sub-species. Properties of this data set are well known in the literature and have become a benchmark to illustrate various clustering and classifier designs. In order to use S in the present context, it is necessary to create training (S_D) and test (S_T) data sets from S. We partition S randomly into two (sub) data sets S_D and S_T, such that $S_D \cap S_T = \emptyset$; $S_D \cup S_T = S$; and $|S_D| = |S_T| = 75$ (cardinality), with 25 points being drawn randomly from each of the three classes. For any partition $(S_D : S_T)$, we first use S_D as the training set and S_T as the test set, and then we switch the data sets and repeat the experiment. We refer to the former run as the forward phase, while the latter run is referred to as the reverse phase. We have generated two such partitions P_1 and P_2 where $P_1 = S_{D_1} \cup S_{T_1}$ and $P_2 = S_{D_2} \cup S_{T_2}$. Table 6 illustrates the results of a typical forward–reverse run on partition P_1 of the IRIS data using the m-URNN and m-NN rules separately on each feature. Error rates in Table 6 are averages of forward–reverse phases.

Table 7 shows results similar to Table 6, but with partition P_2.

From Tables 6 and 7 note that the m-URNN and m-NN rules produce comparable error rates on each feature, just as they did in the Monte Carlo simulations discussed earlier. Further, features 3 and 4 are clearly superior to features 1 and 2, since the rates for either of these features are in the 5% range, (roughly 1/10 of rates achieved using feature 2). The point of tabulating these outputs is for purposes of comparison with the multivariate rules.

Table 8 illustrates the results of applying the m-MRNN and m-NN rules to the same data sets $S = S_D \cup S_T$ as used for Tables 6 and 7, considering each point to be a vector in 4-space. Error rates in Table 8 are again for averages of forward–reverse runs. Observe that using the multivariate data reduces these rates considerably, as one would expect. Distances for the m-NN rule were computed using the Euclidean norm (the 2-norm) in columns 3 and 5, and the Manhattan norm (the 1-norm) in columns 2 and 4. Observe first that the error rates are again comparable, but the m-NN rule is found to work a little better than the m-MRNN rule for this data set and particular partitions. Note that for the m-NN rule one needs to make a choice for d, the metric used by the m-NN rule to assess "nearest". The choice of d is usually a matter of trial and error, and in practice, system design with the m-NN rule must account for this. On the other hand, just as in Tables 6 and 7, the results in

Table 6
Error rates (%) using each feature of the IRIS data (P_1)

Feature	$m \rightarrow$	1	2	3	4	5
1	m-URNN	39.0	34.3	32.3	32.7	32.7
	m-NN	36.0	35.3	35.1	34.3	33.0
2	m-URNN	51.7	46.0	46.0	43.7	42.0
	m-NN	48.7	52.3	50.0	54.0	46.3
3	m-URNN	7.0	5.7	5.7	5.7	5.7
	m-NN	5.3	5.4	4.7	5.0	4.7
4	m-URNN	8.7	8.0	8.0	8.0	8.0
	m-NN	5.3	4.3	4.7	4.7	4.7

Table 7
Error rate (%) using each feature of the IRIS data (P_2)

Feature	$m \rightarrow$	1	2	3	4	5
1	m-URNN	4.0	35.0	33.7	32.7	33.3
	m-NN	31.3	34.7	36.0	38.8	38.0
2	m-URNN	54.0	48.3	48.3	45.7	45.7
	m-NN	52.7	51.7	47.8	46.7	47.0
3	m-URNN	8.7	6.3	6.3	6.3	6.0
	m-NN	4.7	6.0	5.3	5.0	4.7
4	m-URNN	6.7	6.7	6.7	6.7	6.7
	m-NN	3.7	4.0	3.3	3.3	3.3

Table 8
Comparison between m-MRNN and m-NN on the IRIS data in terms of error rate (%)

m	P_1		P_2	
	m-MRNN	m-NN	m-MRNN	m-NN
1	6.7	4.0	8.7	5.3
2	7.0	2.0	9.0	18.7
3	8.0	6.7	9.7	3.3
4	8.3	6.7	10.0	4.7
5	8.7	4.0	10.0	3.3

Table 8 for the m-MRNN rule are the only ones that can be found for partitions P_1 and P_2 of S. In other words, the use of the m-MRNN rule is independent of the *need* to specify a distance measure. Thus, an essential difference between these multivariate rules is the way "evidence" for class labels produced by each feature is aggregated; the m-MRNN rule essentially uses the 1-norm and voting; whereas the metric chosen is the aggregation mechanism for the m-NN rule.

Computational complexities of the m-MRNN and the m-NN rules

Let $|S_D| = a$, $|S_T| = b$, and $S = S_D \cup S_T \subset \mathbb{R}^p$. We assume that the cost of multiplication and comparisons are equal, and we ignore the cost of addition and subtraction. For the m-MRNN rule we need to sort the values of each feature for points in S_D. The total computational overhead for this sort is equal to $(pa \log_2 a)$. Note that this sorting is required only once. Now the computational cost for finding the class of a p-variate observation using m-MRNN rule is $(2mp + p \log_2 a)$ – we assumed the worst case when the decision is made at stage m. Hence, roughly the total computational cost (TC_{MRNN}) for classifying the set S_T is

$$TC_{\text{MRNN}} = (2mp + p \log_2 a)b + (pa \log_2 a). \tag{32}$$

In (32) $\log_2 a$ is the number of comparisons required to find the positions (rank) of an element in a sorted sequence of length a. So $p \log_2 a$ is the cost of finding positions of all p components of z in the respective p sorted sequences each of length a; while $2mp$ is the maximum number of comparisons that may be required to classify all p components of z.

On the other hand, for the m-NN rule we need to compute a distances for each data point in S_T. Thus the total computational cost (TC_{NN}) for processing the entire set S_T is

$$TC_{\text{NN}} = (a \log_2 a + pa + m)b. \tag{33}$$

In (33) pa is the total number of multiplications required for computing a Euclidean distances and m is the number of comparisons for finding class labels of m nearest neighbors.

In (32) $pa \log_2 a$ is a fixed overhead and does not change with b, the number of points to be classified. Now $(a \log_2 a + pa + m)b$ is much greater than $(2mp + p \log_2 a)b$ unless a is very small. So in general, unless a and b are very small, TC_{NN} will be greater than TC_{MRNN}.

5. Conclusions

We have generalized the (univariate) 1-stage URNN rule to an m-URNN rule for s populations, and investigated its theoretical properties. The asymptotic error rates R $(R^{(1)})$ of the 1-NN (1-URNN) rules both lie in the optimal Bayes interval $[R^*, R^*(2 - (s/(s-1))R^*)]$. As the number of training samples increases, the performance of these two rules becomes nearly identical. Their implementational characteristics, however, are different. The asymptotic TPMC of the m-URNN rule is shown to decrease as m increases.

The m-URNN rule enjoys some advantages over the m-NN rule. First, it may lessen the computational burden of computing the distances $\{d(x_{ij}, z)\}$, and second, decisions by the m-URNN rule are less ambiguous, because ties can occur between only two classes (regardless of s), whereas the m-NN rule can suffer an s-way tie. Moreover, the k-NN rules require additional computation to find the class with the highest vote. The m-URNN rule is particularly useful when the observations are available in terms of their ranks.

Finally, we extended (generalized) the univariate m-RNN rule to multivariate observations by defining the m-MRNN rule and empirically investigated its performance. The theoretical properties of the m-MRNN rule need to be investigated.

Acknowledgements

The authors are grateful to Dr. Jim Bezdek for his helpful suggestions and comments made during the preparation of this manuscript. Thanks are also due to an anonymous referee for his/her constructive suggestions on the previous version which led to the present improved version.

References

Anderson, T.W. (1966). Some nonparametric multivariate procedures based on statistical equivalent blocks. In: P.R. Krishnaiah, Ed., *Proc. 1st Internat. Symp. Analysis.* Academic Press, New York.

Bagui, S.C. (1989). Nearest Neighbor Classification Rules for Multiple Observations. Ph.D. thesis, University of Alberta.

Bagui, S.C. (1993). Classification using first stage rank nearest neighbor rule for multiple classes. *Pattern Recognition Lett.* 14, 537–544.

Barnett, V. (1976). The ordering of multivariate data. *J. Roy. Statist. Soc. A* 139 (3), 318.

Cover, T.M. and P.E. Hart (1967). Nearest neighbor pattern classification. *IEEE Trans. Inform. Theory* 13, 21–26.

Das Gupta, S. and H.E. Lin (1980). Nearest neighbor rules of statistical classification based on ranks. *Sankhyá A* 42, 419–430.

Devijver, P.A. and J. Kittler (1982). *Pattern Recognition: A Statistical Approach.* Prentice-Hall, Englewood Cliffs, NJ.

Devroye, L. (1981a). On the inequality of Cover and Hart in nearest neighbor discrimination. *IEEE Trans. Pattern Anal. Mach. Intell.* 3, 75–78.

Devroye, L. (1981b). On the asymptotic probability of error in nonparametric discrimination. *Ann. Statist.* 9, 1320–1327.

Fix, E. and J.L. Hodges (1951). Nonparametric discrimination: consistency properties. U.S. Air Force School of Aviation Medicine. Report No. 4, Randolph Field, TX.

Fritz, J. (1975). Distribution-free exponential error bound for nearest neighbor pattern classification. *IEEE Trans. Inform. Theory* 21, 552–557.

Gessaman, M.P. (1970). A consistent nonparametric multivariate density estimator based on statistically equivalent blocks. *Ann. Math. Statist.* 41, 1344–1346.

Johnson, R. and Wichern, D. (1988). *Applied Multivariate Statistical Analysis.* Prentice-Hall, Englewood Cliffs, NJ.

Wagner, T.J. (1971). Convergence of nearest neighbor rule. *IEEE Trans. Inform. Theory* 17, 566–571.