OCTOBER 1984 LIDS-P-1411

GENERATION AND TERMINATION OF BINARY DECISION TREES FOR NONPARAMETRIC MULTICLASS CLASSIFICATION

S. Gelfand

S.K. Mitter

Department of Electrical Engineering and Computer Science

and

Laboratory for Information and Decision Systems

Massachusetts Institute of Technology

Cambridge, MA 02139

This research has been supported by the U.S. Army Research Office under Grant DAAG29-84-K-0005.

Abstract

A two-step procedure for nonparametric multiclass classifier design is described. A multiclass recursive partitioning algorithm is given which generates a single binary decision tree for classifying all classes. The algorithm minimizes the Bayes risk at each node. A tree termination algorithm is given which optimally terminates binary decision trees. The algorithm yields the unique tree with fewest nodes which minimizes the Bayes risk. Tree generation and termination are based on the training and test samples, respectively.

I. Introduction

We state the nonparametric multiclass classification problem as follows. M classes are characterized by unknown probability distribution functions. A data sample containing labelled vectors from each of the M classes is available. A classifier is designed based on the training sample and evaluated with the test sample

Friedman [1] has recently introduced a 2-class recursive partitioning algorithm, motivated in part by the work of Anderson [2], Henderson and Fu [3], and Meisel and Michalopoulos [4]. Friedman's algorithm generates a bindary decision tree by maximizing the Komolgorov-Smirnov (K-S) distance between marginal cumulative distribution functions at each node. In practice, an estimate of the K-S distance based on a training sample is maximized. Friedman suggests solving the M-class problem by solving M 2-class problems. The resulting classifier has M binary decision trees.

In this note we give a multiclass recursive partitioning algorithm which generates a single binary decision tree for classifying all classes. The algorithm minimizes the Bayes risk at each node. In practice an estimate of the Bayes risk based on a training sample is minimized. We also give a tree termination algorithm which optimally terminates binary decision trees. The algorithm yields the unique tree with the fewest nodes which minimizes the Bayes risk. In practice an estimate of the Bayes risk based on a test sample is minimized.

The research was originally done in 1981-82 [5]. The recent book of Breiman et al [6] has elements in common with this paper but we believe the approach presented here is different.

The note is organized as follows. In Section 2 we give binary decision tree notation and cost structure for our problem. In Section 3 and 4 we discuss tree generation and termination, respectively.

II. Notation

We shall be interested in classifiers which can be represented by binary decision trees. For our purposes, a binary decision tree T is a collection of nodes $\{N_i\}_{i=1}^{K}$ with the structure shown in Fig. 2.1. The levels of T are ordered monotonically as 0, 1,...,L-1 going from bottom to top. The nodes of T are ordered monotonically as 1,2,...,K going from bottom to top, and for each level from left to right. We shall find it convenient to denote the subtree of T with root node N_i and whose terminal nodes are also terminal nodes of T as T(i) (see Fig. 2.1).

We associate a binary decision tree and a classifier in the following way. For each node $N_i \varepsilon T$ we have at most five decision parameters: k_i, α_i, s_i, r_i, and c_i. Suppose $\underline{\alpha} \varepsilon \mathbb{R}^d$ is to be classified. The root node N_K is where the decision process begins. At N_i the $k_i{}^{th}$ component of $\underline{\alpha}$ will be used for discrimination. If $\alpha^k < \alpha_i$ the next decision will be made at N_{s_i} .* If $\alpha^k \geq \alpha_i$ the next decision will be made at N_{r_i} . If N_i is a terminal node then $\underline{\alpha}$ is labelled class c_i. It is easily seen that a binary decision tree with these decision parameters can represent a classifier which partitions R^d into d-dimensional intervals. The algorithms we shall discuss generate binary decision trees as partitioning proceeds.

Let H_j be the hypothesis that the vector under consideration belongs to the j^{th} class, j=1,...,M. We denote be l_j the misclassification cost for H_j

and π_j the prior probability of H_j. The Bayes risk (of misclassification) is then given by $\sum_{j=1}^{M} \ell_j\pi_j(1 - \Pr\{\text{decide } H_j|H_j\})$.

III. Tree Generation

In this section generation of binary decision trees is discussed. An algorithm is given which generates a single binary decision tree for classifying all classes. The algorithm minimizes the Bayes risk at each node. In practice an estimate of the Bayes risk based on a training sample is minimized.

We first review Friedman's 2-class algorithm. Friedman's algorithm is based on a result of Stoller's [5] concerning univariate nonparametric classification (d=1). We assume $l_1\pi_1 = l_2\pi_2$.

Stoller solves the following problem: find α^* which minimizes the Bayes risk based on the classifier

$$\begin{array}{ll} \alpha<\alpha^* & \text{decide } H_1 \text{ or } H_2 \\ \alpha\geq\alpha^* & \text{decide } H_2 \text{ or } H_1 \end{array}$$

Let $F_1(\alpha)$, $F_2(\alpha)$ be the cumulative distribution functions (c.d.f.'s) for H_1, H_2 respectively, and let

$$D(\alpha) = |F_1(\alpha) - F_2(\alpha)| \tag{3.1}$$

Stoller shows that

$$\alpha^* = \arg\max_{\alpha} D(\alpha) \tag{3.2}$$

($D(\alpha^*)$ is the Komolgorov-Smirnov distance between F_1 and F_2). This procedure can be applied recursively until all intervals in the classifier meet a termination criterion. A terminal interval I is then assigned the class label

$$c^* = \arg \max_{j=1,2} \Pr\{\alpha\varepsilon I | H_j\} \tag{3.3}$$

Friedman extends Stoller's algorithm to the multivariate case ($d\geq 2$) by solving the following problem: find k^* and α^* which minimize the Bayes risk of the classifier

$$\alpha^{k*} < \alpha^* \qquad \text{decide } H_1 \text{ or } H_2$$

$$\alpha^{k*} \geq \alpha^* \qquad \text{decide } H_2 \text{ or } H_1$$

Let $F_{1,k}(\alpha)$, $F_{2,k}(\alpha)$ be the marginal c.d.f.'s on coordinate k for H_1,H_2 respectively, and let

$$D_k(\alpha) = |F_{1,k}(\alpha) - F_{2,k}(\alpha)| \tag{3.4}$$

In view of (3.2) we have

$$\alpha^*(k) = \arg\max_{\alpha} D_k(\alpha)$$

$$k^* = \arg\max_k D_k(\alpha^*(k)) \tag{3.5}$$

$$\alpha^* = \alpha^*(k^*)$$

As with the univariate case, Friedman's procedure can be applied recursively until all (d-dimensional) intervals in the classifier meet a termination criterion. A terminal interval is then assigned class label

$$c^* = \arg\max_{j=1,2} \Pr\{\underline{\alpha}\varepsilon I \,|\, H_j\} \tag{3.6}$$

To apply Friedman's algorithm to the nonparametric classification problem we must estimate $F_{j,k}(\alpha)$ and $\Pr\{\underline{\alpha}\varepsilon I|H_j\}$. Let $\underline{\alpha}_{1,1},\ldots,\underline{\alpha}_{1,n_1}$, $\alpha_{2,1},\ldots,\underline{\alpha}_{2,n_2}$ be the training sample vectors where $\underline{\alpha}_{j,i}$ is the i^{th} vector from the j^{th} class. Suppose we have arranged the sample such that $\alpha^k_{j,1} \leq \alpha^k_{j,2} \leq \ldots \leq \alpha^k_{j,n}$. We estimate $F_{j,k}(\alpha)$ by

$$F_{j,k}(\alpha) = \begin{cases} 0 & \alpha < \alpha^k_{j,1} \\ i/n_j & \alpha^k_{j,i} \leq \alpha < \alpha^k_{j,i+1} \\ 1 & \alpha \geq \alpha^k_{j,n_j} \end{cases}$$

and $\Pr\{\underline{\alpha}\varepsilon I|H_j\}$ by the fraction of training sample vectors in class j which land in I.

Note that Friedman's algorithm generates a binary decision tree as partitioning proceeds by appropriately identifying the decision parameters of Section 2.

Friedman extends his algorithm to the M-class case by generating M binary decision trees, where the j^{th} tree discriminates between the j^{th} class and all the other classes taken as a group. We next propose an extension which has the advantage of generating a single binary decision tree for classifying all classes. At the same time we relax the constraint that all the ${}_j\pi_j$'s are equal.

Consider the following problem: find the k^*, α^*, m^* and n^* which minimize the Bayes risk based on the classifier

$$\alpha^{k^*} < \alpha^* \qquad \text{decide } H_{m^*} \text{ or } H_{n^*}$$

$$\alpha^{k^*} \geq \alpha^* \qquad \text{decide } H \text{ or } H$$

Let

$$\begin{aligned} R_{m,n,k}(\alpha) = \min\{&\ell_m\pi_m(1-F_{m,k}(\alpha)) + {}_n\pi_n F_{n,k}(\alpha), \\ &\ell_n\pi(1-F_{n,k}(\alpha)) + {}_m\pi_m F_{m,k}(\alpha)\} \\ &-\sum_{j\neq m,n} \ell_j\pi_j \end{aligned} \qquad (3.7)$$

Then it can easily be shown that

$$\alpha^*(m,n,k) = \arg\min_{\alpha} R_{m,n,k}(\alpha)$$

$$k^*(m,n) = \arg\min_{k} R_{m,n,k}(\alpha^*(m,n,k))$$

$$(m^*,n^*) = \arg\min_{m,n} R_{m,n,k^*(m,n)}(\alpha^*(m,n,k^*(m,n)))$$

$$k^* = k^*(m^*,n^*)$$

$$\alpha^* = \alpha^*(m^*,n^*,k^*) \tag{3.8}$$

Furthermore, if $\ell_1\pi_1 = \ldots = \ell_M\pi_M$ the minimizations over $R_{m,n,k}(\alpha)$ reduce to maximizations over

$$D_{m,n,k}(\alpha) = |F_{m,k}(\alpha) - F_{n,k}(\alpha)| \tag{3.9}$$

If we now replace the double maximization (3.5) in Friedman's algorithm with the triple minimization (3.8) we get the proposed multiclass recursive partitioning algorithm. Of course (3.6) should be replaced by

$$c^* = \arg\max_{j=1,\ldots M} {}_j\pi_j \Pr\{\underline{\alpha}\varepsilon I|H_j\} \tag{3.10}$$

Otherwise the algorithms are the same. In particular the multiclass algorithm generates a single bindary decision tree as partitioning proceeds by appropriately identifying the decision parameters of Section 2. Note that m^* and n^* are <u>not</u> decision parameters.

IV. Tree Termination

In this section termination of binary decision trees is discussed. An algorithm is given for optimally terminating a binary decision tree. The algorithm yields the unique tree with fewest nodes which minimizes the Bayes risk. In practice an estimate of the Bayes risk based on a test sample is minimized.

Suppose we generate a binary decision tree with the multiclass recursive partitioning algorithm of Section 3. Partitioning can proceed until terminal nodes only contain training sample vectors from a single class. In this case the entire training sample is correctly classified. But if class distributions overlap the optimal Bayes rule should not correctly classify the entire training sample. Thus we are led to examine termination of binary decision trees.

Friedman introduces a termination parameter k = minimum number of training sample vectors in a terminal node. The value of k is determined by minimizing the Bayes risk. In practice an estimate of the Bayes risk based on a test sample is minimized. In the sequel we will refer to the binary decision tree with terminal nodes only containing training sample vectors from a single class as the "full" tree. What Friedman's method amounts to is minimizing the Bayes risk over a subset of the subtrees of the full tree with the same root node. At this point the following question arises: is there a computationally efficient method of minimizing the Bayes risk over all subtrees of the full tree with the same root node? The answer is yes as we shall now show.

We first state a certain combinatorial problem. Suppose we have a binary decision tree and with each node of the tree we associate a cost. We define the cost of each subtree as the sum of the costs of its terminal nodes. The problem is to find the subtree with the same root node as the original tree which maximizes cost. More precisely, let $T_o = \{N_i\}_{i=1}^{K}$ be a binary decision tree with L levels and K_i nodes at level i as described in Section 1, g_i the cost associated with node N_i, and G(T) the cost of subtree T. Then

$$G(T) = \sum_{i=1}^{K} 1_i(T)g_i \tag{4.1}$$

where

$$1_i(T) = \begin{cases} 1 & N_i \text{ is a terminal node of } T \\ 0 & \text{else} \end{cases}$$

Now let F be the set of subtrees of T_o withe the same root node N_K. The problem can then be stated as:

$$\max_{T\varepsilon F} G(T) = \operatorname*{Max}_{T\varepsilon F} \sum_{i=1}^{K} 1_i(T)g_i \tag{4.2}$$

Next consider the following simple algorithm. Going from first to last level and for each level from left to right, if deleting descendents of current node does not decrease cost, delete descendents and go to next node, etc. In view of (4.1) the algorithm becomes:

```
For $i = 1,\ldots, L-1$ do:
    $T_i \leftarrow T_{i-1}$
    For $j = K_{i-1} + 1,\ldots, K_i$ do:
        If $g_j \geq G(T_i(j)$:
            $T_i(j) \leftarrow \{N_j\}$
```

Define $T^* = T_{L-1}$. We claim that T^* solves (4.2).

Theorem: $G(T^*) \geq G(T)$ for all $T \varepsilon F$. Furthermore, if $G(T^*) = G(T)$ for some $T \varepsilon F$, $T \neq T^*$, then T^* has fewer nodes than T.

Proof: See Appendix.

Finally, we show that the problem of minimizing the Bayes risk over all subtrees of the full tree with the same root node has form (4.2). Let T_o be the full tree and

$$g_i = \ell_{c_i} \pi_{c_i} \Pr\{\underline{\alpha} \varepsilon N_i | H_{c_i}\} \quad i=1,\ldots,K \tag{4.3}$$

where c_i is the class label of N_i if N_i becomes a terminal node, i.e.,

$$c_i = \arg \max_{j=1\ldots,M} \ell_j \pi_j p_{ij} \tag{4.4}$$

where p_{ij} is the fraction of training sample vectors in class j which land in N_i. Then by direct computation the Bayes risk of $T \varepsilon S$ is given by

$$R(T) = \sum_{j=1}^{M} \ell_j \pi_j - \sum_{i=1}^{K} 1_i(T) g_i = \sum_{j=1}^{M} \ell_j \pi_j - G(T) \tag{4.5}$$

Hence, minimizing R(T) is equivalent to maximizing G(T). In practice an estimate of R(T) based on a test sample is minimized. In this case

$$g_i = {}_{c_i}\pi_{c_i} q_{ic_i} \qquad i = 1,\dots,K \tag{4.6}$$

where q_{ij} is the fraction of test sample vectors in class j which land in N_i.

APPENDIX

Proof of Theorem Section IV: Let S_i be the set of subtrees of T_o with the same root node N_K and which only have nodes missing from levels i-1,...,0 (or equivalently, every terminal node on levels i,...,L-1 is also a terminal node of T_o). We shall say that T_i is optimal over S_i if the theorem holds with T^* and S replaced by T_i and S_i, respectively. We show that T_i is optimal over S_i for i = 1,...,L-1. Since $T^* = T_{L-1}$ and $S = S_{L-1}$ the theorem follows. We proceed by induction. T_1 is clearly optimal over S_1. We assume T_i is optimal over S_i and want to show that T_{i+1} is optimal over S_{i+1}. Let $T \varepsilon S_{i+1}$ and $T \neq T_{i+1}$. There are four cases to consider.

Suppose there exists a terminal node $N_j \varepsilon T_{i+1}$ which is a nonterminal node of T and N_j is on some level $\leq$ i. Construct $T' \varepsilon S_{i+1}$ from T by terminating T at N_j. Since N_j is a terminal node of T_{i+1} it is also a terminal node of T_i and it follows from (4.1) and the optimality of T_i that $g_j \leq G(T(j))$ so that $G(T') \leq G(T)$, and since T' has fewer nodes than T, T cannot be optimal over S_{i+1}.

Next, suppose there exists a terminal node $N_j \varepsilon T$ which is a nonterminal node of T_{i+1} and N_j is on some level $\leq i$. Contruct $T' \varepsilon S_{i+1}$ from T by augmenting T with $T_{i+1}(j)$ at N_j. Since $T_{i+1}(j) = T_i(j)$ it follows from (4.1) and the optimality of T_i that $G(T'(j)) < g_j$ so that $G(T') < G(T)$, and consequently T cannot be optimal over S_{i+1}.

Next, suppose there exists a terminal node $N_j \varepsilon T_{i+1}$ which is a nonterminal node of T and N_j is on level i+1. If $T(j) = T_i(j)$ construct $T' \varepsilon S_{i+1}$ from T by terminating T at N_j. Since $g_j \leq G(T_i(j)) = G(T(j))$ it follows from (4.1) that $G(T') \leq G(T)$, and since T' has fewer nodes than T, T cannot be optimal over S_{i+1}. If $T(j) \neq T_i(j)$ construct $T' \varepsilon S_{i+1}$ from T by replacing T(j) with $T_i(j)$. At this point we essentially are in one of the preceding cases (with T_{i+1} replaced by T').

Finally, suppose there exists a terminal node $N_j \varepsilon T$ which is a nonterminal node of T_{i+1} and N_j is on level i+1. Construct $T' \varepsilon S_{i+1}$ from T by augmenting T with $T_{i+1}(j)$ at N_j. Since $T_{i+1}(j) = T_i(j)$ we have $g_j > G(T_i(j)) = G(T_{i+1}(j)) = G(T'(j))$ and it follows from (4.1) that $G(T) > G(T')$, and consequently T cannot be optimal over S_{i+1}. QED

REFERENCES

[1] J.H. Friedman (1977), "A Recursive Partitioning Decision Rule for Nonparametric Classification," IEEE Trans. Computers, Vol. C-26, pp. 404-408.

[2] T.W. Anderson (1969), "Some Nonparametric Multivariate Procedures Based on Statistically Equivalent Blocks," in Multivariate Analysis (ed. P.R. Krishnaiah), New York: Academic Press.

[3] E.G. Henrichon and K.S. Fu (1969), "A Nonparametric Partitioning Procedure for Pattern Classification," IEEE Trans. Computers, Vol. C-18, pp. 614-624.

[4] W.S. Meisel and D.R. Michalopoulos (1973), "A Partitioning Algorithm with Application in Pattern Classification and the Optimization of Decision Trees," IEEE Trans. Computers, Vol. C-22, pp. 93-103.

[5] S. Gelfand (1982), "A Nonparametric Multiclass Partitioning Method for Classification," S.M. Thesis, MIT, Cambridge, MA.

[6] L. Breiman, et al: Classification and Regression Trees, Wadsworth International Group, California, 1984.