NASA

Technical Memorandum 80676

AN AUTOMATED APPROACH TO THE DESIGN OF DECISION TREE CLASSIFIERS

P. Argentiero, R. Chin and P. Beaudet

MARCH 1980

National Aeronautics and Space Administration

Goddard Space Flight Center
Greenbelt, Maryland 20771

AN AUTOMATED APPROACH TO THE DESIGN OF DECISION TREE CLASSIFIERS

P. Argentiero
Goddard Space Flight Center

R. Chin and P. Beaudet
Business and Technological Systems, Inc.

March 1980

Abstract

The classification of large dimensional data sets arising from the merging of remote sensing data with more traditional forms of ancillary data causes a significant computational problem. Decision tree classification is a popular approach to the problem. This type of classifier is characterized by the property that samples are subjected to a sequence of decision rules before they are assigned to a unique class. If a decision tree classifier is well designed, the result in many cases is a classification scheme which is accurate, flexible and computationally efficient.

This paper provides an automated technique for effective decision tree design which relies only on *apriori* statistics. This procedure utilizes a set of two dimensional canonical transforms and Bayes table look-up decision rules. An optimal design at each node is derived based on the associated decision table. A procedure for computing the global probability of correct classification is also provided.

An example is given in which class statistics obtained from an actual LANDSAT scene are used as input to the program. The resulting decision tree design has an associated probability of correct classification of .76 compared to the theoretically optimum .79 probability of correct classification associated with a full dimensional Bayes classifier.

Recommendations for future research are included.

1. Introduction

The tendency in remote sensing technology is toward the merging of remote sensing information with collateral data to form high dimensional data sets. The classification of such data to produce conventional thematic maps creates problems of two kinds. First, the computational expense associated with classification increases steeply with dimension. Second, there is the familiar fact[1,2,3] that for a fixed number of training samples, classification accuracy can actually decline with an increase in dimension. A conventional solution to these problems is to reduce dimensionality by means of a feature extraction transformation the coefficients of which are chosen so as to optimize a class separability measure.[4] A decision rule is then applied to assign the reduced dimensional samples to the available classes. But the feature extraction represents a compromise since the separability measure to be optimized must take into account the overlap of each class with each other class. Hence, classification accuracy is not always satisfactory.

This paper presents an alternative approach to dimensionality reduction and classification which involves a procedure for the automated design of a *decision tree classifier.*[5] This approach to classification will be described in more detail in section 2, but basically it is characterized by the fact that samples are subjected to a sequence of decision rules before they are assigned to a unique class. Each decision rule can leave ambiguity with regard to the precise class assignment of a sample. If the ambiguity is unacceptable for a particular application it can be removed by subsequently applied decision rules. When the structure is diagrammed to show the heirarchy among the decision rules it exhibits a characteristic tree-like aspect – thus the rubric "decision tree classifier."

There are numerous advantages to decision tree classification. Most importantly, the decision rules can be designed to be both inexpensive and effective since each rule is required to take into account only a small subset of the original classes and it is not required to remove all ambiguities. Also, there is considerable generality and flexibility associated with this type of classification. For instance, collateral data of a categorical nature such as soil type or political boundaries can be readily incorporated within the framework of a decision tree classifier. Also, it is easy to avoid situations in

which computer time is spent in removing ambiguities which are irrevelant to a particular application such as distinguishing among confuser crops in an agricultural scene. Of course, for these benefits to be realized it is important for decision trees to be well designed. This paper presents an automated technique for designing effective decision tree classifiers predicated only on *apriori* class statistics. The procedure relies on a set of two dimensional feature extractions and Bayes table look-up decision rules. Associated error matrices are computed and utilized to provide an optimal design of the decision tree at each so-called "node." A byproduct of this procedure is a simple algorithm for computing the global probability of correct classification assuming the statistical independence of the decision rules.

Section 2 provides a more precise definition of decision tree classification. Section 3 gives mathematical details on the technique for automated decision tree design. Section 4 gives an example of a simple application of the procedure using class statistics acquired from an actual LANDSAT scene. Section 5 summarizes results and discusses directions for future research.

2. A Mathematical Description of Decision Tree Classification

The purpose of this section is to give a rigorous description of decision tree classification.

Consider a classification problem with sample set M to be assigned to K classes indexed by a K dimensional index set $\mathcal{J} = \{1, 2, \ldots K\}$. Let $\tau_j \subset M$ be the set of samples properly associated with the class indexed by j. Also define $\Pi = \{\mathcal{J}_1, \mathcal{J}_2, \ldots \mathcal{J}_n\}$ as a set of nonempty subsets of $\mathcal{J}$ which satisfies conditions $\bigcup_i \mathcal{J}_i = \mathcal{J}$ and $\mathcal{J}_i \cap \mathcal{J}_{i'} = \phi$ if $i \neq i'$. A *generalized decision rule* is defined as a transformation $f: M \rightarrow \Pi$ from the sample set M to a set Π of disjoint subsets of $\mathcal{J}$. An element $x \epsilon \tau_j \subset M$ is considered to be correctly classified by decision rule f if $f(x) = \mathcal{J}_i$ and $j \epsilon \mathcal{J}_i$.

It is convenient to express a generalized decision rule as a pair (D, Π) where D is a set of parameters which define the transformation from M to Π and where Π is an explicitly expressed set of subsets of $\mathcal{J}$. Since a decision tree classifier will be defined as a set of generalized decision rules which collectively satisfy certain conditions we shall refer to a pair (D, Π) as a *node*. Because a sample is subjected to a given decision rule if and only if another decision rule has mapped the sample into a certain index subset, it is important to describe the partial ordering that must exist on a set of generalized decision rules for that set to constitute a decision tree classifier.

Definition: Let $\eta_i = (D_i, \Pi_i)$, $\Pi_i = \{\mathcal{J}_{i,1}, \mathcal{J}_{i,2} \ldots \mathcal{J}_{i,Nj}\}$ and $\eta_j = (D_j, \Pi_j)$, $\Pi_j = \{\mathcal{J}_{j,1}, \mathcal{J}_{j,2} \ldots \mathcal{J}_{j,Nj}\}$ be two nodes. Then η_i is a *parent node* of η_j if there exists a $K \leq N_i$ such that $\bigcup_\varrho \mathcal{J}_{j,\varrho} = \mathcal{J}_{i,K}$. In this case η_j will be refered to as an *offspring node* of η_i

A Set which consists of just one element will be called a *unitary set* and nodes whose index subsets consist entirely of unitary sets will be called *simple nodes*. A node which does not have a parent will be called a *root node*. To complete our terminology, a node without offspring will be called a *terminal node.*

Definition: A *decision tree classifier* is a set of nodes statisfying the following conditions

(a) There is just one root node

(b) Every node which is not a root node has a single parent

In addition to conditions (a) and (b), a *complete decision tree classifier* satisfies condition (c):

(c) For every element i in the original index set $\mathcal{J}$ there exists a node $\eta = (D, \Pi)$, $\Pi = \{\mathcal{J}_1, \mathcal{J}_2, \ldots \mathcal{J}_N\}$ and a $K \leq N$ such that $\mathcal{J}_K$ is a unitary set and $i \epsilon \mathcal{J}_k$.

There are a variety of intuitively pleasing properties possessed by complete decision tree classifiers. For instance, one can show that all terminal nodes are simple nodes. To complete the definition we must indicate precisely how a decision tree structure is used to classify samples. Let $\{\eta_i\}_I$ be a set of nodes forming a decision tree with reference to a sample set M and a set of classes indexed by index set $\mathcal{I}$. For every sample $x \epsilon M$ there is a unique *decision sequence* S(x) composed of an ordered set of nodes in $\{\eta_i\}_I$ with the following properties:

(a) The first node in the sequence S(x) is the root node of $\{\eta_i\}_I$

(b) If $\eta_j = (D_j, \Pi_j)$, $\Pi_j = \{\mathcal{I}_{j,1}, \mathcal{I}_{j,2} \cdots \mathcal{I}_{j,N_j}\}$ is the i^{th} element in S(x), then

$$\eta_\varrho = (D_\varrho, \Pi_\varrho), \Pi_\varrho = \{\mathcal{I}_{\varrho,1}, \mathcal{I}_{\varrho,2}, \ldots \mathcal{I}_{\varrho,N_\varrho}\}$$

is the i + 1 at element of the sequence if η_j maps x onto $\mathcal{I}_{j,k} \epsilon \Pi_j$ and $\bigcup_k \mathcal{I}_{\varrho,k} = \mathcal{I}_{j,k}$. If no available node satisfies this condition, the decision sequence terminates at η_j.

In effect, S(x) is the unique set of generalized decision rules used by the decision tree $\{\eta_i\}_I$ to classify $x \epsilon M$. The index set to which x is assigned is understood to be the image of x under the mapping defined by the terminating node in S(x). If $\{\eta_i\}_I$ is a complete decision tree one can show that each sample is mapped onto a unitary index set. Hence, a complete decision tree classifier maps samples onto unique class indexes.

3. Decision Tree Design

A complete decision tree can be designed in a natural fashion by first defining a structure for the root node. Next, for every non unitary index subset associated with the root node it is necessary to define another node which performs a further decomposition into smaller index subsets. The defining process continues until simple nodes are achieved and no further decomposition is possible. At each step of the process we are confronted with the problem of decomposing a certain index set $\mathcal{I}$ into a set of subsets $\Pi = \{\mathcal{I}_1, \mathcal{I}_2, \ldots \mathcal{I}_N\}$ and developing a computationally efficient generalized decision rule which maps samples into Π in a way that provides adequate classification accuracy. We now concentrate on a solution to this problem.

Let $f: M \rightarrow \mathcal{I}$ be a decision rule which unambiguously assigns samples $x \epsilon M$ to elements in index set $\mathcal{I}$. Associated with f is an *error matrix* $\mathcal{E}_f$ defined as

$\mathcal{E}_f(i,j) \rightarrow$ conditional probability that a sample from a class indexed by i is assigned by decision rule f to the class indexed by j

Let $\Pi = \{\mathcal{I}_1, \mathcal{I}_2, \ldots \mathcal{I}_N\}$ be a decomposition of $\mathcal{I}$ into disjoint subsets. The decision rule f uniquely defines a generalized decision rule $f': M \rightarrow \Pi$ in the following natural way. Assume that $f(x) = i$ and let $\mathcal{I}_{j(i)}$ be the element of Π which contains i. Then $f'(x) \triangleq \mathcal{I}_{j(i)}$. Furthermore, given error matrix $\mathcal{E}_f$ one can readily compute the probability of correct classification for f' which is

$$P_{f'} = \sum_{i \epsilon \mathcal{I}} a_i \sum_{\ell \epsilon \mathcal{I}_{j(i)}} \mathcal{E}_f(i, \ell) \qquad (1)$$

where a_i is the *apriori* class probability associated with the class indexed by i. The right side of equation one represents a very simple and efficient algorithm for computing classification accuracy. Hence, even for relatively large index sets it is possible to investigate every generalized decision rule generated by f and to choose the one associated with maximum classification accuracy. In this fashion it is possible to generate from a decision rule f with mediocre classification accuracy a generalized decision rule f' with very high classification accuracy.

For the procedure outlined above to be a feasible approach to decision tree design, two requirements must be satisfied:

(a) a decision rule f must be available which affords a satisfactory compromise between accuracy and computational efficiency.

(b) the error matrix E_f must be computable.

These conditions can be met with a two dimensional linear feature extraction matched with Bayesian table look-up decision rule as suggested by Mobasseri and McGillem.[6] The feature extraction is performed by means of a canonical analysis approach as suggested by Merembeck and Turner.[7] That is, if m is the dimension of the sample set M, we first seek an m dimensional row vector A_1 which maximizes the F-ratio

$$F_A = \frac{ABA^T}{AWA^T} \tag{2}$$

where B is the between class covariance matrix as defined from class mean vectors and W is the pooled within class covariance matrix. The required A_1 is the eigenvector associated with the largest eigenvalue of $W^{-1}B$. Having determined A_1 we seek a row vector A_2 in the orthogonal compliment of A_1 which maximizes the F-ratio defined by equation 2. It, in turn, is the eigenvector associated with the second largest eigenvalue of $W^{-1}B$. Our required two dimensional transformation matrix has A_1 as the first row and A_2 as the second row.

The first step in designing the associated table look-up classifier is to specify a location and dimensions of a rectangle in the transformed two dimensional feature space such that the rectangle contains at least 99% of the probability associated with each class density function under the usual normality assumption. Next the rectangle is divided into 256 equal area grid elements. Each element is associated with a class index according to the maximum likelihood classification of its midpoint. The resulting decision rule simply assigns to each transformed sample the class index of the grid element in which it is contained. Samples which fall outside of the rectangle are assigned to the nearest grid element.

It remains to define how the error matrix $\mathcal{E}$ of the above defined table look-up decision rule is computed. A given element $\mathcal{E}(i,j)$ of $\mathcal{E}$ can be obtained by summing the integrals of the transformed normal density function associated with the i^{th} class over each grid element indexed by j. We have found that a good approximation can be obtained if the density functions are first represented in each grid by a two dimensional second order Taylor series expanded about the grid midpoint. The Taylor series representations rather than the density functions are then integrated over the appropriate grid elements.

An important byproduct of this approach to decision tree design is a convenient method for computing the associated global probability of correct classification under the assumption that the decision rules employed at each node are statistically independent. To see how the computation is performed, we first determine the conditional probability Pi that a sample from a class indexed by i is correctly classified by a complete decision tree $\{\eta_j\}_j$. Let $\{\eta_\ell\}_{\ell \in R_i}$ be a set of nodes

$$\eta_\ell = (D_\ell, \Pi_\ell), \quad \Pi_\ell = \{\mathcal{J}_{1,\ell}, \mathcal{J}_{2,\ell}, \ldots \mathcal{J}_{N_\ell,\ell}\}$$

such that for some $j \leq N_\ell$, $i \in \mathcal{J}_{j,\ell}$. The probability that a sample from the class indexed by i is properly classified at node η_ℓ is

$$P_{i,\ell} = \sum_{k \in \mathcal{J}_{j,\ell}} \mathcal{E}_\ell(i, k) \tag{3}$$

where $i \in \mathcal{J}_{j,\ell}$ and $\mathcal{E}_\ell$ is the error matrix of the table look-up decision rule employed at node η_ℓ. As usual, let $\mathcal{J}$ be the class index set and let $\{a_i\}$ $i \in \mathcal{J}$ be a set of *apriori* class probabilities. Then assuming the statistical independence of decision rules, the global probability of correct classification is

$$P = \sum_{i \in \mathcal{J}} a_i \prod_{\ell \in R_i} P_{i,\ell} \tag{4}$$

4. An Example

The procedure outlined in section 3 for automated decision tree design was incorporated into a FORTRAN program which now resides on an IBM 360/91 computer at the Goddard Space Flight Center. The input is a set of class mean vectors and covariance matrices. The output is a description of each node of a decision tree design. Each node description consists of a decomposition of an index set into index subsets, the coefficients of the two dimensional linear feature extraction, a decision table for the table look-up classifier and its associated confusion matrix. The output also includes a computation of the global probability of correct classification as obtained from equations 3 and 4. As described in section 3, the program designs the decision tree from the top down starting from the root node. The design terminates when each of the original class indexes appears in a unitary index subset. A simple flow chart for the program is included in figure 1.

To provide an example of the application of the program, class statistics were obtained from a LANDSAT 2 scene taken over Finney County, Kansas during May of 1975.[8] The five classes consisted of two types of winter wheat and three confuser crops. The class statistics were obtained from well known sites in Finney County. The four channels are those of the Multispectral Scanner on board the LANDSAT 2. The sizes of the training sample sets range from about one hundred to about three hundred. The class statistics are shown in Table 1. The information in Table 1 was used as input to the program and the resulting decision tree design is shown as a tree diagram in figure 2. Table 2 shows the part of the program output which describes the root node. Similar information about the other nodes is also made available. Table 3 lists all possible designs for the root node and the associated probabilities of correct classification. Design number 5 is seen to have the optimal probability of correct classification and is employed in the tree design as shown in figure 2.

The global probability of correct classification for the decision tree shown in figure 2 was computed to be .76. From the results of a previous study[9] it is known that the theoretically optimal 4 dimensional Bayes decision rule provides an accuracy of 79% when applied to this problem. Hence, for this application the more efficient and more flexible decision tree approach provides a classification which is nearly optimum.

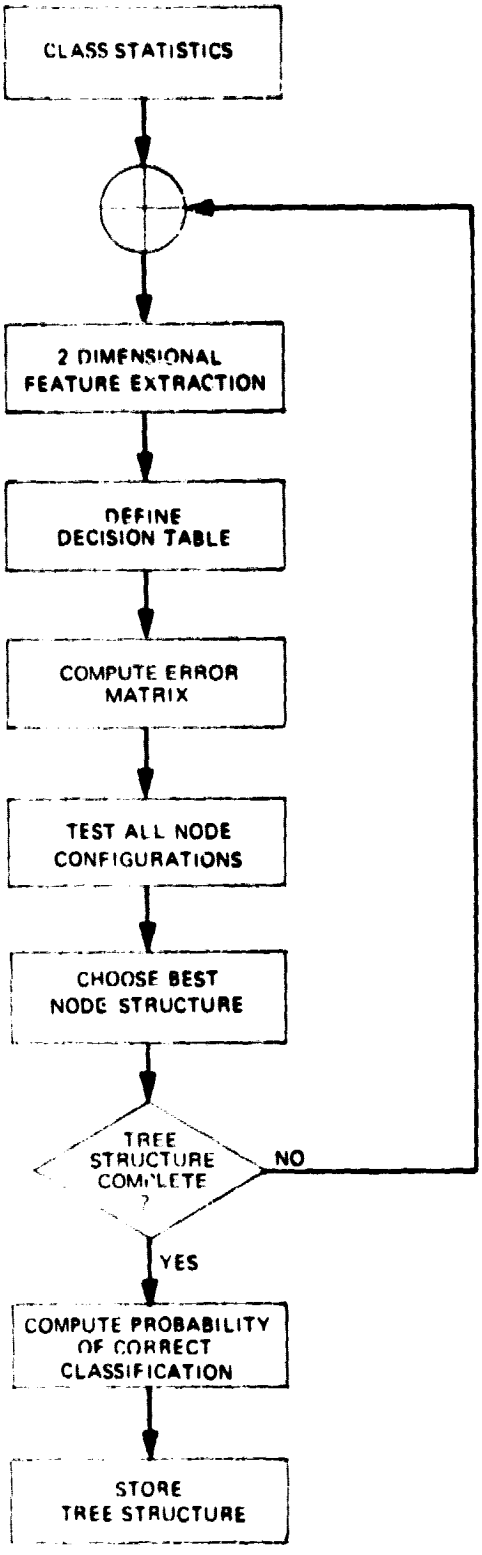

Figure 1. Flow Chart for Design of Decision Tree Classifier

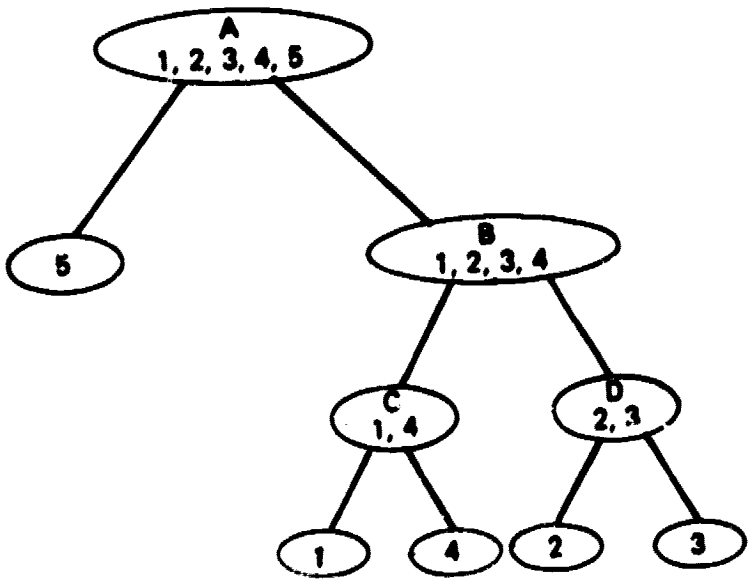

Figure 2. Design of Decision Tree Classifier Associated with Statistics Shown in Table 1

Table 1
Statistics of LANDSAT-2 MSS Signatures Acquired
May 1975 Over Finney County, Kansas

Channel	Mean	Std. Dev.	Covariance Matrix 1	2	3	4
(1) 184 Pixels of Non-Wheat						
1	27.7	3.6	12.7		SYMMETRIC	
2	24.5	8.0	25.0	63.4		
3	75.1	20.4	-51.4	-140.7	415.5	
4	37.4	12.0	-30.8	-84.2	242.1	143.4
(2) 333 Pixels of Non-Wheat						
1	34.7	3.6	12.7			
2	40.4	5.5	17.2	30.0		
3	47.0	5.2	8.8	9.9	27.3	
4	19.7	2.5	0.6	-1.2	10.4	6.0
(3) 324 Pixels of Non-Wheat						
1	33.3	1.6	2.6			
2	38.5	2.7	2.6	7.2		
3	44.1	6.4	4.3	2.5	41.2	
4	18.7	3.3	1.9	0.3	19.9	11.1
(4) 106 Pixels of Winter Wheat						
1	28.5	2.4	5.8			
2	27.5	4.0	7.4	16.2		
3	51.2	5.2	-6.0	-14.4	26.7	
4	24.0	3.0	-4.3	-8.9	14.1	9.0
(5) 127 Pixels of Winter Wheat						
1	21.5	2.7	7.3			
2	16.7	4.2	10.3	18.0		
3	54.9	5.1	4.1	4.9	26.0	
4	29.1	2.8	-1.0	-2.8	11.4	8.1

Table 2
Information Associated with the Root Node of Decision Tree Shown in Figure 2

Transformation Matrix →

$$\begin{bmatrix} .48 & .71 & -.38 & -.35 \\ -.02 & -.06 & .58 & -.81 \end{bmatrix}$$

Error Matrix

$$\begin{bmatrix} .883 & .011 & .056 & .045 & .005 \\ 0 & .473 & .512 & .015 & 0 \\ 0 & .192 & .808 & 0 & 0 \\ .012 & .042 & .178 & .758 & .009 \\ .005 & 0 & 0 & 0.004 & 0.991 \end{bmatrix}$$

Decision Table

5	5	5	5	5	5	1	1	1	1	1	1	1	1	1	
5	5	5	5	5	5	1	1	1	1	1	1	1	1	1	
5	5	5	5	5	5	5	1	1	1	1	1	1	1	1	
5	5	5	5	5	5	5	1	1	1	1	1	1	1	1	
5	5	5	5	5	5	5	1	1	1	1	1	1	1	1	
5	5	5	5	5	5	5	5	1	1	1	1	1	1	1	
5	5	5	5	5	5	5	1	1	1	1	1	1	1	1	
5	5	5	5	5	5	4	1	1	1	1	1	1	1	1	
5	5	5	4	4	4	4	4	1	1	1	1	1	1	1	1
4	4	4	4	4	4	4	4	1	1	1	1	1	1	1	1
4	4	4	4	4	4	4	4	1	1	1	1	1	1	1	1
4	4	4	4	4	4	4	2	3	3	3	3	3	3	3	3
4	2	2	2	2	3	3	3	3	3	3	3	3	3	3	2
2	3	3	3	3	3	3	3	3	2	2	2	2	2	2	2
3	3	3	3	3	3	2	2	2	2	2	2	2	2	2	2
3	3	3	3	2	2	2	2	2	2	2	2	2	2	2	2

Table 3
All Possible Configurations of the Root Node and their Probabilities of Correct Classification

Case	Node Structure	Probability
1	(1) (2,3,4,5)	0.97827
2	(2) (1,3,4,5)	0.77264
3	(3) (1,2,4,5)	0.75595
4	(4) (1,2,3,5)	0.96345
5	(5) (1,2,3,4)	0.99722
6	(1,2) (3,4,5)	0.75465
7	(1,3) (2,4,5)	0.75341
8	(1,4) (2,3,5)	0.95945
9	(1,5) (2,3,4)	0.97829
10	(2,3) (1,4,5)	0.96223
11	(2,4) (1,3,5)	0.75343
12	(2,5) (1,3,4)	0.76986
13	(3,4) (1,2,5)	0.75467
14	(3,5) (1,2,4)	0.75317
15	(4,5) (1,2,3)	0.96343
16	(1) (2) (3,4,5)	0.75278
17	(1) (3) (2,4,5)	0.74382
18	(1) (4) (2,3,5)	0.95058
19	(1) (5) (2,3,4)	0.97689
20	(2) (3) (1,4,5)	0.74541
21	(2) (4) (1,3,5)	0.74476
22	(2) (5) (1,3,4)	0.76986
23	(3) (4) (1,2,5)	0.73703
24	(3) (5) (1,2,4)	0.75317
25	(4) (5) (1,2,3)	0.96205
26	(1) (2,3) (4,5)	0.95197
27	(1) (2,4) (3,5)	0.74243
28	(1) (2,5) (3,4)	0.75140
29	(1) (2,3) (4,5)	0.74474
30	(1) (2,4) (3,5)	0.74262
31	(2) (1,5) (3,4)	0.75280
32	(3) (1,2) (4,5)	0.73701
33	(4) (1,2) (3,5)	0.73563
34	(5) (1,2) (3,4)	0.75327
35	(3) (1,4) (2,5)	0.74263
36	(3) (1,5) (2,4)	0.74383
37	(4) (1,3) (2,5)	0.74336
38	(5) (1,3) (2,4)	0.75203
39	(4) (1,5) (2,3)	0.95198
40	(5) (1,4) (2,3)	0.95945
41	(1) (2) (3) (4,5)	0.73514
42	(1) (2) (4) (3,5)	0.73376
43	(1) (2) (5) (3,4)	0.75140
44	(1) (3) (4) (2,5)	0.73376
45	(1) (3) (5) (2,4)	0.74243
46	(1) (4) (5) (2,3)	0.95058
47	(2) (3) (4) (1,5)	0.73516
48	(2) (3) (5) (1,4)	0.74263
49	(2) (4) (5) (1,3)	0.74336
50	(3) (4) (5) (1,2)	0.73563
51	(1) (2) (3) (4) (5)	0.73376

5. Summary and Recommendation for Future Research

The classification of large dimensional data sets arising from the merging of remote sensing data with more traditional forms of ancillary data causes a significant computational problem. Decision tree classification is an increasingly popular approach to the problem. This type of classifier is characterized by the property that samples are subjected to a sequence of decision rules before they are assigned to a unique class. If a decision tree classifier is well designed, the result in many cases, is a classification scheme which is accurate, flexible, and computationally efficient.

It is useful to have available an automated procedure for effective decision tree design which relies only on *apriori* class statistics. The procedure described in this paper utilizes a set of two dimensional feature extractions and Bayes table look-up decision rules. An optimal design at each node is derived based on the associated error matrix. A procedure for computing the global probability of correct classification is also provided.

An example is provided in which class statistics obtained from an actual LANDSAT scene are used as input to the program. The resulting decision tree design shown in figure 2 has an associated probability of correct classification of .76 which compares reasonably to the theoretically optimum .79 probability of cover classification associated with a full dimensional Bayes classifier.

The work documented in this report represents a promising depiction in the exploitation of decision tree classification. An obvious next step is to test the procedure on large dimensional merged data sets with results compared to ground truth information. Also, monte carlo studies are in order to validate the computational procedure for determining the global probability of correct classification as given in equations 3 and 4. This is particularly important for rather deep decision tree structures where samples can be subjected to many decision rules before being finally classified. It is possible in this situation that the independence assumption can lead to error.

It is also clear that the automated procedure described in section 3 should be modified to include greater flexibility. For instance, it should be possible to permit a user to employ collateral data of a catagorical nature in defining certain mode structures of decision tree. Also, it should be possible to insure that a decision tree design reflect the fact that for a certain application, certain ambiguities among classes are irrevelent. As an example, for the case presented in section 4, classes 1, 2, and 3 are confuser crops in an agricultural scene. Hence, node D as represented in figure 2 can be deleted from the tree structure with no loss of useful information.

References

(1) Hughes, G., "On the Mean Accuracy of Statistical Pattern Recognizers," IEEE Trans. on Information Theory, Vol. IT-14, No. 1, pp 55-63, January, 1968.

(2) Boullion, P., Odell, P., Duran, B. "Estimating the Probability of Misclassification and Variate Selection," Pattern Recognition, Vol. 7, pp 139-145, 1975.

(3) Wahl, P., Kronmal, R., "Discriminant Functions when Covariances are Unequal and Sample Sizes are Moderate," Biometrics 33, 479-484, September, 1977.

(4) Decell, H., Guseman, L., "Linear Feature Selection with Applications," Pattern Recognition, Vol. 11, pp. 55-63, 1979.

(5) Wu, C., Landgrebe, D., Swain, P., "The Decision Tree Approach to Classification," LARS Information Note 090174, 1974.

(6) Mobasseri, B., McGillem, "Multiclass Bayes Estimation By a Feature Space Sampling Technique," IEEE Transactions on Systems, Man, and Cybernetics, Vo. SMC-9, No. 10, Oct 1979.

(7) Merembeck, B., Turner, B., "Directed Canonical Analysis and the Performance of Classifiers Under its Associated Linear Transformations," Proc. 1979 Symp. on Machine Processing of Remotely Sensed Data, Purdue University, West Lafayette, In, June 1979.

(8) Σ^2 Study, Capability Demonstration for Landsat-D System, Final Report, by General Electric, Contract NAS5-23412, Mod. 30.

(9) Argentiero, P., Strong, Jr., Koch, D., "Inventory Estimation on the Massively Parallel Processor," Proc. 1980 Symp. on Machine Processing of Remotely Sensed Data, Purdue University, West Lafayette, In., June 1980.