Estimating Feature Discriminant Power in Decision Tree Classifiers*

I. Gracia[1], F. Pla[1], F. J. Ferri[2] and P. García[1]

[1] Departament d'Informàtica. Universitat Jaume I
Campus Penyeta Roja, 12071 Castelló. Spain
gracia@inf.uji.es, +34 64 345676
[2] Dept. d'Informàtica i Electrònica. Universitat de València
Dr Moliner, 50 46100 Burjassot (València). Spain
ferri@uv.es, +34 6 3864768

Abstract. Feature Selection is an important phase in pattern recognition system design. Even though there are well established algorithms that are generally applicable, the requirement of using certain type of criteria for some practical problems makes most of the resulting methods highly inefficient. In this work, a method is proposed to rank a given set of features in the particular case of Decision Tree classifiers, using the same information generated while constructing the tree. The preliminary results obtained with both synthetic and real data confirm that the performance is comparable to that of sequential methods with much less computation.

1 Introduction

From a theoretical point of view, adding new features always improve the performance of any classifier. However, when using finite sets of prototypes to train a classifier, recognition accuracy usually decreases above a certain threshold in the number of features actually used. This effect is known as the *peaking phenomenon* [6]. The situation can be even worse due to the so-called *curse of dimensionality* or combinatorial explosion of most of the algorithms involved.

In order to avoid the above mentioned problems, dimensionality reduction methods have been proposed and studied since long time ago. Feature Selection (FS) methods [4] constitute a particular case in which a small or moderate number of features are selected from the whole set of measurements.

Recent contributions to the field [10] point out that no general satisfactory solutions to the FS problem have been found mainly because two well-know facts. First, solving the FS problem is NP-hard, therefore no optimal method can be used for realistic problems. And second, good feature selection criteria need so much computation that even polinomic-time algorithms are inapplicable for some problems.

* This work has been partially supported by project P1A94-23 Fundació Caixa Castelló, and project GV-2110/94 Conselleria d'Educació i Ciència, Generalitat Valenciana.

Hlaváč, Šára (Eds.): CAIP '95 Proceedings, LNCS 970

Decision Trees (DT) are a particular and interesting type of classifier which allows dealing with many different types of feature sets, and even problems where the feature set is not fixed (uncertainty in some measurements).

This paper presents an attempt to do FS in DT classifiers. Instead of using classical FS methods that would require constructing and evaluating a large number of trees, we propose to extract the information about feature effectiveness from the same tree construction process.

2 Feature Selection Methods

FS methods require the use of a criterion function used to evaluate tentative inclusion/exclusion of features in/from the selected set. Assuming that a suitable criterion has been adopted, FS is reduced to a search problem in which a number of techniques can be used. There exists an optimal branch-and-bound method [8] apart from exhaustive search but, it requires excessive computation for realistic problems. Moreover, it can only be used with monotonic criterion functions.

Sequential search [4] constitutes a valid alternative to optimal procedures for real problems. This family of heuristic methods consists of a greedy strategy to explore the possible feature subsets. Provided that k features have been selected, the method considers the inclusion of non-used features in turn and selects the one that gives a higher value of the criterion. This particular method is called Sequential Forward Selection (SFS) because we start from the empty set and keep including features at each step. Starting from the whole set and considering exclusion at each step leads to the Sequential Backward Selection (SBS) method.

Further modifications [6] of the original idea lead to the (l, r) method, in which a prefixed number of forward (l) and backward (r) steps are performed in order to correct premature erroneous decisions. This idea has been further generalized by considering a non fixed number of forward and backward steps [10]. Among other approaches to FS, it is worth mentioning the use of genetic algorithms to perform feature subset search [12].

3 Decision Trees

DT are binary trees where each leave (*terminal node*) has a class label assigned, and each non-leave (*decision node*) represents a (simple) decision rule. During the classification process, each pattern is dropped into the tree at the root node. The pattern follows a path through the tree according to the decision nodes and it is thus assigned to the class of the terminal node reached.

In the most usual case, the rule at each decision node involves only one feature (axis parallel splitting rules). In this case, deterministic search can be used to find the parameters of the decision rule [2]. Other heuristic methods are used in the case of linear combination of features at each node [7] or other approaches like using small neural nets at each node [5].

DT's are usually built expanding each decision node until all of them are pure (only prototypes from one class). As this tree overfits the data in the training set

used, it is pruned using some heuristic technique [11, 2] considering prototypes non used in the tree expanding process.

Other methods include stopping rules during the tree growing process so they do not expand the tree completely [1, 2]. Although pruning (ascending) methods are considered in the literature to give better results, other descending methods [9] can give as good results than some pruning methods.

The misclassification rate in the training set (resubstitution estimate) decreases as the tree grows. Conversely, the "true" error rate (estimated using different prototypes) gets to a minimum before increasing as the tree overfits the data in the training set. Ascending methods try to find this minimum by pruning the fully expanded tree while descending methods use a minimum error expanding criterion [9] which briefly consist of expanding at each step the node which produce a maximum decrease in the (holdout) estimated tree error rate. Both methods generate a succession of trees of consecutive sizes.

4 Feature Selection using Decision Trees

The FS method proposed in this work is based on the ideas described in the previous section about descending methods to build a DT. Each one of the trees of consecutive sizes obtained, has an error estimate associated. As the growing step from a tree to the next is due to feature x, which is used to split a node, the difference between the estimates of the two trees can be used as a partial measure of the discrimination power of feature x.

To obtain a measure of the discriminant power of each feature used in the whole tree, these partial measures can be accumulated for each feature along the tree building process. Thus, the features which produce a maximum overall decrease in the error rate will be considered as the most discriminant ones.

The classification error was estimated using holdout for the tree growing process. Holdout has the advantage that the discriminant information obtained is more related to the final performance of the DT classifier.

Using the approach described, the algorithm to asses the discriminant power of each feature for a given classification problem could be expressed as follows:

1. Set to 0 the discriminant power of every feature. Initialize the tree: build the root node. Separate the set of available prototypes, $\mathcal{L}$, into two subsets, $\mathcal{L}_1$ and $\mathcal{L}_2$.
2. Find the best split for each terminal node according to the impurity criterion measured using the first subset $\mathcal{L}_1$.
3. Select the one which leads to a maximum decrease in the holdout error estimated using the second subset, $\mathcal{L}_2$.
4. If the error increases (it is a minimum), go to step 6.
5. Else, increase the discriminant power of the feature used in the split and go to step 2.
6. Sort the features in descending order according to their discriminant powers.

The algorithm returns a list of the D features sorted according to their discriminant power, considering that the more discriminant is a particular feature, the bigger is the decrease produced by this feature every time this feature is used.

5 Experiments and Discussion

From the ranking of features obtained with the proposed method, a chain of nested subsets is obtained and evaluated by estimating the misclassification rate of the corresponding classifier. The SFS method is used as well for comparison purposes because this method also produce a ranking of features and is preferable to other sequential methods from a computational point of view. The accuracy of the k-Nearest Neighbour classifier is selected as the criterion function for this sequential method. For the two experiments presented a value of $k = 3$ is used.

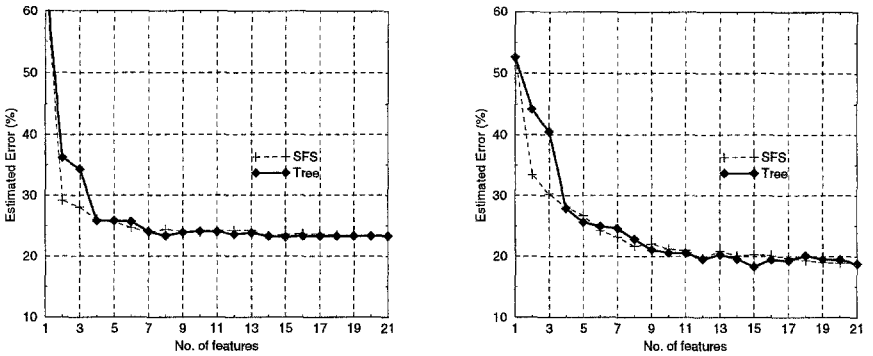
In the following experiments both DT and k-NN classifiers are used to evaluate the feature subsets obtained. The DT's are built using the Gini index as impurity measure and the cost complexity pruning method [2] to choose the right tree. A value of $k = 3$ is selected again for the k-NN classifier for the same reasons.

synthetic data: The problem consist of a waveform recognition experiment [2] which has already been used for benchmarking purposes [5, 7]. In this problem we have 21-dimensional vectors of three different classes. For this problem, 2600 samples were drawn to be used as a training set, both for tree generation (2000) and tree selection (600), and another independent, identically generated set of 6000 samples was used to evaluate the feature sets obtained.

real data: A real problem about texture recognition is used to evaluate the methods. Thirteen 512x512 images from the Brodatz album [3] were used to extract texture measurements from random 64x64 windows. Each sample consisted of 15-dimensional vectors extracted from the Fourier spectrum of each window. A set of 3250 samples was used as training set (2145+1105) and, a different one with the same number of samples to evaluate the feature subsets.

Figure 1 shows the classification error rate for the waveform recognition problem with respect to the number of features involved in the construction of a DT classifier (a) and a 3-nearest neighbour classifier (b). The curve labeled as *Tree* denotes estimated errors found using the feature subsets obtained with the proposed method, and the curve labeled as *SFS* denotes the estimated errors found using the SFS method for feature selection. The results show that the proposed method and SFS give, in general, similar results.

Figure 2 represents the same results shown in Figure 1 for the texture classification problem. In (b) we can notice that the SFS method detects a feature which increases the error rate more than 10% (15th feature). Using the proposed method, this feature is added as the 13th feature, and it also increases the error rate at this point. The reason because the proposed method does not detect such a noisy feature could be due to the fact that DT's have the property to be quite robust to noisy data, so it seems natural that DT's are not too much affected



(a) DT classifier (b) 3NN classifier

Fig. 1. Performance of the feature subsets obtained with the proposed method (Tree) and SFS in the waveform recognition problem.

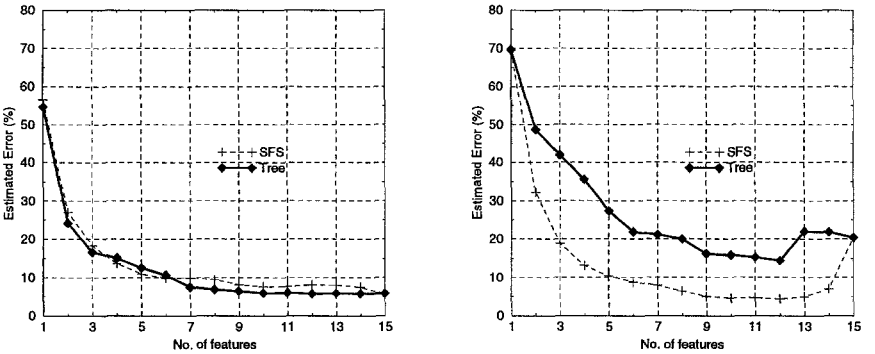
with the introduction of this feature. The better result of the SFS in (b) is clearly consequence of the fact that the same classifier has been used for feature selection and for evaluation. On the other hand, we can notice that there is no such a difference in the results shown in (a) where DT's were used to evaluate the feature subsets. In this case the proposed method gives similar or better results in some cases. These results are also comparable to the best results from (b).

6 Conclusions and Further Work

An attempt to use DT's in feature selection has been presented. These first results show that the method gives comparable results to sequential feature selection approaches when using a DT as classifier. Therefore, the use of the proposed method for feature selection in DT's appears to be a valid starting point to develop feature selection methods for binary trees.

The main advantage of the proposed method, in its current implementation, is the significant reduction of the computational cost with respect to sequential methods, in which inclusion or exclusion of features needs the construction of a different classifier for each possible decision, leading to a heavy computational burden.

Further improvements of the proposed method could be done by using DT's with linear splits, involving several features at each node, and therefore, expanding the current work to this type of DT classifiers.



(a) DT classifier (b) 3NN classifier

Fig. 2. Performance of the feature subsets obtained with the proposed method (Tree) and SFS in the texture recognition problem.

References

1. Boswell, R.; *Manual for NewID*, version 5.1, The Turing Institute, Ref. TI/P2154/RAB/4/2.4.
2. Breiman L. et. al., *Classification and Regression Trees*, Chapman & Hall, 1984
3. Brodatz, *Textures: A Photographic Album for Artists and Designers*, Dover Publications, New York, 1966
4. Devijver, P.A. and Kittler, J. *Pattern Recognition: a Statistical Approach*, Prentice-Hall International, 1982
5. Guo, H. and Genlfand, S.B. "Classification Trees with Neural Network Feature Extraction", IEEE Trans. on Neural Networks, Vol. **3**, No. 6, pp. 923-933,1992.
6. Kittler, J. "Feature Selection and Extraction", *Handbook of Pattern Recognition and Image Processing*, 1986
7. Murthy, S.K.; Kasif, S. and Salzberg, S.; "A System for Induction of Oblique Decision Trees", Journal of Artificial Intelligence Research, **2**, 1994, pp. 1-32.
8. Narendra, P.M. and Fukunaga, K. "A Branch and Bound Algorithm for Feature Subset Selection", IEEE Trans. Comput., vol. **C-26**, pp 917-922, Sept. 1977
9. Pla, F. *Estudios de Técnicas de Análisis de Imagen en un Sistema de Visión para la Recolección Robotizada de Cítricos*, (in Spanish) Ph.D. Thesis, Universitat de València, 1993.
10. Pudil, P.; Ferri, F., Novoviçová, J and Kittler J. "Floating Search Methods for Feature Selection with Nonmonotonic Criterion Functions", in Proc. of the 12th Intl. Conf. on Pattern Recognition, Jerusalem, 1994.
11. Quinlan, J.R. "Simplifying decision trees", International Journal of Man-Machine Studies **27**, pp. 221-234, 1987.
12. Siedlecki, W. and Sklansky, J. "On Automatic Feature Selection", International Journal of Pattern Recognition and Artificial Intelligence, Vol. **2**, pp. 197-220, 1988.