A Data Complexity Analysis of Comparative Advantages of Decision Forest Constructors *

Tin Kam Ho
Bell Laboratories, Lucent Technologies
700 Mountain Avenue, 2C425, Murray Hill, NJ 07974, USA
tkh@bell-labs.com

October 18, 2001

Abstract

Using a number of measures for characterizing the complexity of classification problems, we studied the comparative advantages of two methods for constructing decision forests – bootstrapping and random subspaces. We investigated a collection of 392 two-class problems from the UCI depository, and observed that there are strong correlations between the classifier accuracies and measures of length of class boundaries, thickness of the class manifolds, and nonlinearities of decision boundaries. We found characteristics of both difficult and easy cases where combination methods are no better than single classifiers. Also, we observed that the bootstrapping method is better when the training samples are sparse and the subspace method is better when the classes are compact and the boundaries are smooth.

keywords: data complexity, classifier combination, decision tree, decision forest, bagging, random subspace method.

1 Introduction

Since the early 1990's many methods have been developed for classifier combination. These methods are results of two parallel lines of study: (1) assume a given, fixed set of carefully designed and highly specialized classifiers, attempt to find an *optimal combination* of their decisions; and (2) assume a fixed decision combination function, *generate* a set of mutually complementary, generic classifiers that can be combined to achieve optimal accuracy. We will refer to combination strategies of the first kind as *decision optimization* methods and the second kind as *coverage optimization* methods.

Just like with a single classifier, when a combined classifier achieves 100% accuracy with given test data, it is accepted that the solution is optimal. When observed accuracy is less than that, it is often unclear what the reasons are. Theoretically[5] and empirically, there is a general understanding that different types of data require different kinds of classifiers. Similar arguments can be made for

*To appear, Pattern Analysis and Applications

combined systems. However, we believe that it is possible and desirable to make more explicit the nature of such a dependency on data characteristics. A recent attempt is a step in this direction [32].

In [17] we presented a number of measures useful for characterizing the complexity of classification problems. A collection of real world problems are found to be distributed in a continuum in a space spanned by these measures, with extremely difficult ones (random labeling) and extremely simple (linearly separable) ones occupying opposite ends. Thus the measurement space is effective for problem characterization.

Given such a space, one can then ask whether classifiers differ in performance for problems located in different regions. A similar question can be asked of classifier combination methods. Thus, other than the two questions of extremes

1. does this method work?
2. does this method work for this particular problem?

we are now ready to investigate a third question

3. what type of problems does this method work for?

In this paper we focus on two coverage optimization methods for classifier combination. We use oblique decision trees [13] [28] as the basic classifier, and bootstrapping (*bagging*) [4] and the random subspace method [13][14] for constructing decision forests. A decision forest is a collection of decision trees together with a decision combination function. In both types of forests, individual tree decisions are combined by averaging the estimated posterior probabilities at the leave nodes.

2 Sources of difficulty of a classification problem

We begin with a brief analysis of the sources of difficulty of a classification problem. We isolate these sources for convenience of discussion, though it is understood that real world problems often include difficulties from each of these sources. For simplicity we assume each problem is on discrimination of two classes. Multiclass problems will be reduced to discrimination of each pair of classes, and we note that it is not trivial to recombine such pairwise decisions to a final decision.

- **Class ambiguity.** Some problems are known to have nonzero Bayes error [15]. Even if labeled samples are available at every point of the given feature space, no classifiers can be trained to perform perfectly. These problems need sufficiently abundant samples that can reflect accurately the prior and posterior probability distributions of the classes. This is independent of the shape of the class boundary and feature space dimensionality. While certain problems may be intrinsically ambiguous, others may be so because of a lack of discriminating features due to poor design. The Bayes error is a measure of difficulty in this aspect and it sets a lower bound on the achievable error rate.
- **Boundary complexity.** Some problems have a long, geometrically or topologically complicated (Bayes) optimal decision boundary. These problems are complex by Kolmogorov's

notion (the boundary needs a long description or a long algorithm to reproduce, possibly including an enumeration of all points of each class) [25]. An example is a set of randomly located points each labeled arbitrarily as one of two classes. Such a problem can be very difficult even if every point in the space is covered by a sample, that there is no ambiguity in the class assignment, and even if it resides in a one-dimensional feature space.

- **Small sample effect and feature space dimensionality.** The danger of having a small training set is that it may not reflect the full complexity of the underlying problem, so from the available samples the problem appears deceptively simple (Figure 1). This happens easily in a high dimensional space [8] [20] [27] [30] as the class boundary can vary with a larger degree of freedom. The representativeness of a training set is related to the generalization ability of classifiers, which is a focus of study in Vapnik's statistical learning theory [34] and is also discussed in Kleinberg's arguments on M-representativeness [22], Berlind's hierarchy of indiscernibility [2], and Raudys' and Jain's practical considerations [30]. It is also intensively discussed in many studies of error rate estimation [11] [12] [21] [33].

Imperfect accuracy of classifiers may be due to a combination of many reasons. Attempts to improve classifiers have to deal with each of them in some way. Among these, class ambiguity is either a nature of the problem or requires additional discriminatory features, and little can be done using classifiers after feature extraction. On the other hand, most classifiers are designed with the goal of finding a good decision boundary. So in this paper we will focus on the boundary complexity. But since the sample size constrains what can be known about either class ambiguity or boundary complexity of a given problem, we can address only the *apparent* complexity of a problem based on a given training set.

3 Measures of problem complexity

Practical classification problems involve geometrical characteristics of the classes in the feature space coupled with probabilistic events in the sampling processes. Some theoretical studies focus on distribution-free or purely combinatorical arguments without taking into account the geometrical aspects of the problems. This often leads to unnecessarily weak results. Most classifier designs build upon simple geometrical heuristics such as proximity, convexity, and globally or locally linear boundaries. We believe that such elementary geometrical properties of the data distributions are of central importance in pattern recognition, so we emphasize these in this study.

We assume that we do not know the true data distribution of each problem, so that we have to estimate the problem complexity from a given training sample, i.e., a set of points labeled with two classes. We consider a number of measures that have been proposed in the literature for characterizing geometrical complexity. These measures give empirical estimates of the *apparent* complexity of a problem, which may or may not be close to the true complexity depending on the sparsity of training data.

We consider the complexity of a problem in a given, fixed feature space. This restriction is introduced for the following reason. Certain practical problems possess a structure or regularity so that there exists a transformation with which the points can be mapped to a new space where class discrimination becomes easier. While such transformation is not necessarily obvious for an

arbitrary problem, discussion of problem complexity will be complicated without reference to a fixed space.

For example, consider a two-dimensional discrete space where two classes are interleaved in a fashion like the black and white cells on a checkerboard. Suppose the size of the space is fixed. A grid of a finer scale will yield a longer boundary between the classes than one of a coarser scale. At the limit, if class labels alternate at every other point, every point lies on the boundary. Nevertheless, if one knows the spatial periodicity of the point distribution, one can use a very simple transformation to map the points to another space where the classes can be easily separated. Both the mapping and the discriminator in the new space can be described in a simple algorithm, so the problem has very low Kolmogorov complexity. One cannot exclude the possibility that such mappings can be found with an arbitrary dataset. But nor can one have an algorithm that always finds the mapping giving the shortest description. This is known as the mathematical fact that Kolmogorov complexity is not effectively computable [26].

For simplicity we consider only two-class discrimination problems. Among a number of useful measures we investigated in [17], we chose those describing the following aspects of boundary complexity of a given problem:

1. length of class boundary
2. thickness of class manifolds
3. maximum Fisher's discriminant ratio
4. training set size relative to feature space dimensionality
5. ratio of average intra/inter class nearest neighbor distances
6. nonlinearity of nearest neighbor or linear classifiers.

Length of class boundary
This is given as percent points on an edge connecting two opposite classes in the (class-indifferent) minimum spanning tree (MST) connecting all training samples. The use of this measure was motivated by the test proposed by Friedman and Rafsky [10] for whether two multivariate samples are from the same distribution. We chose Euclidean distance to be the metric in constructing the MST. For n training points there are $n-1$ edges in the MST, so the count is normalized as a percentage of n. This measure is sensitive to both the separability of the classes and the clustering tendency of the points of each class. A linearly separable problem with wide margins (relative to the intra-class distances) may have only one edge going across the classes. But another linearly separable problem may have many such edges if the points of the same class happen to be farther apart than they are from those of the other class. On the other hand, a problem with a complicated nonlinear class boundary may still have only one boundary-crossing edge as long as the points are compact within each class.

Thickness of class manifolds
This is given as percent points with a retained adherence subset. A (highest order) adherence subset is a pretopological concept[23] which in this context is realized as the largest hyperspherical neighborhood centered at a specific training point of a class c that does not include a point not in

c, using again the Euclidean metric. We eliminate any adherence subset associated with a point if it is strictly included in one of another point. Then we count how many points have their adherence subsets retained, and represent that as a percentage of the total number of points. If we consider the support of the class-conditional distributions as manifolds in the feature space, this percentage conveys some information about the thickness of the manifolds. If all points of one class are distributed within a hypersphere centered at one of the points, only the adherence subset of that point will be retained. On the other hand, if the points spread over a long and thin structure along the class boundary, many points will have their adherence subset retained.

Maximum Fisher's discriminant ratio
A classical measure of the discriminative power of the features is Fisher's discriminant ratio:

$$f = \frac{(\mu_1 - \mu_2)^2}{{\sigma_1}^2 + {\sigma_2}^2}$$

where μ_1, μ_2, ${\sigma_1}^2$, ${\sigma_2}^2$ are the means and variances of the two classes respectively. An estimate of f is given by using the sample means and variances instead.

f as defined above is specific to one feature dimension. For a multidimensional problem, not necessarily all features have to contribute to class discrimination. A problem is easy as long as there exists at least one discriminating feature. So only the maximum f over all features would matter. However, a zero maximum f does not mean that the classes are inseparable, it could just be that the separating boundary is not parallel to an axis in the given feature space.

Training set size relative to feature space dimensionality
This is given as the number of training samples divided by the number of features. This measure shows the sparsity of the samples relative to the size of the feature space. A problem with sparse samples is not necessarily intrinsically difficult, but there could be a large difference between the apparent complexity observed from the samples and the intrinsic complexity of the underlying problem.

Ratio of intra versus inter class nearest neighbor distances
This measure compares the dispersion within the classes to the separation between the classes, and is calculated as follows. First we compute the distances from each point to its nearest neighbor within or outside the class. Then we take the average of all the distances to intra-class nearest neighbors and the average of all those to inter-class nearest neighbors. The ratio of the two averages is used as a measure. A close to one ratio may mean that the classes are completely mixed, but it could also occur in a case where two well separated classes are distributed each in a long, thin, and sparse structure along the boundary.

Nonlinearity of nearest neighbor or linear classifiers
A measure of nonlinearity of a classifier w.r.t. a given dataset is proposed by Hoekstra and Duin [18]. The convex hull containing the training set of each class is approximated by linear interpolation (with random coefficients) between randomly drawn pairs of points from the same class. Then the error rate of the classifier (trained by the same training set) on this enhanced set is measured. This measure is sensitive to both the smoothness of the classifier's decision boundary and the overlap of the convex hulls of the two classes. High nonlinearity could be due to a severe overlap of the (randomly filled) convex hulls, or a poorly chosen or trained classifier unable to match the shape of the class boundary, or both reasons. We consider the nonlinearity of a linear classifier (minimizing error by linear programming [1]) and that of a nearest neighbor classifier.

Joint usage of the measures

Each of these measures captures some aspect of the geometrical complexity of the problem, and individually is rarely sufficient for characterizing the difficulty of the problem. For example, a large percentage of points on boundary can arise from at least two conditions: either the two classes are heavily intermixed, so that the nearest neighbor of almost every point is of a different class; or that the classes are well separated but the points are distributed far apart relative to the separation between the two classes and that they are all close to the class boundary. Therefore the size of the boundary alone does not determine the difficulty of a problem – in the case of long and thin classes with small separation, if there is sufficient structural simplicity that a chosen classifier can match, very low error rate can be achieved.

However, there is evidence that jointly these measures provide a reasonable measurement space in which problems as difficult as random noise or as simple as linearly separable ones are well separated[17]. Thus we consider these measures useful indicators into some deeper underlying geometrical, topological, and combinatorial properties of the point set assumed to represent a problem. Before we can develop a powerful enough language to describe such properties, we hope these will provide helpful insights into the structure of the class distributions.

4 Classifiers and their combination

Accuracies of decision optimization methods are limited by the given set of classifiers, which makes it difficult to discuss the effects of data characteristics on their behavior. Therefore we focus our study on the coverage optimization methods that specify a strategy to construct a collection of classifiers systematically. Two typical methods of this category are by varying the training sets and by varying the features (or metric) used. These are respectively represented by the bootstrapping (or *bagging*) method [4] and the random subspace method [13][14]. In the bagging method, subsets of the training points are selected independently and randomly with replacement according to a uniform probability distribution. A classifier is constructed using each selected subset. In the random subspace method, in each pass all the training points are projected onto a randomly chosen coordinate subspace in which a classifier is derived. The projections can also be made to a coordinate subspace of an augmented feature space where simple combinations (sums, differences, or products) of the raw features are included. Both methods are known to work well using decision trees as component classifiers, and the combined classifier is called a *decision forest*.

A decision forest is the most general form of classifiers since it allows both serial and parallel combinations of arbitrary discriminators. Serial combinations are realized by placing the discriminators at different levels of each tree, and parallel combinations occur when multiple trees are combined to form a forest. A decision node can be as simple as a split by thresholding one particular feature, or as sophisticated as a support vector machine, i.e., a linear discriminator with maximum margin in a kernel-transformed space. Other classifiers are special cases of decision forests where there may be only one tree, or some trees may have only one level. For example, nearest neighbor matching to a number of prototypes can be considered as a special case where the tree has only one level with multiple branches corresponding to the Voronoi cells. Therefore though our choices of the types of forests does not include the full generality, we believe that a study of their behavior can give important insights into the combination of other classifiers.

The decision trees constructed in this study use oblique hyperplanes to split the data at each internal node [28]. The hyperplanes are derived using a simplified Fisher's method [13]. First the centroids of each class are found. Then the hyperplanes perpendicular to the line connecting the centroids are evaluated by the count of points falling on the wrong side (the right side being the one containing the majority of points of that class). The hyperplane that minimizes the sum of these counts is chosen for that node. Assuming no ambiguity in the class labels, the tree can always be fully split, and trees constructed this way are usually small.

5 The collection of problems

We investigated the complexity measures and classifiers' behavior using a collection of problems arising from 14 datasets from the UC-Irvine Machine Learning Depository. The 14 datasets were selected from those containing at least 500 points with no missing values: *abalone, car, german, kr-vs-kp, letter, lrs, nursery, pima, segmentation, splice, tic-tac-toe, vehicle, wdbc, and yeast.* For those sets containing categorical features, the values were numerically coded. With each dataset, we took every pair of classes to be a problem. Totally there are 844 two-class discrimination problems. Of the 844 problems, 452 were found to be linearly separable by a linear programming procedure [1]. We used the remaining 392 problems in this study. The main reason for omission of the linearly separable problems is that many of them are too small (say, with less than 10 points) for useful estimates of classifier accuracy. Also since these problems are relatively easy, there is less interest in improving accuracy. Their complexity characterization is included in [17]. Recall that since we used only the training sets, we studied only *apparent* complexity of the underlying problems.

We used the following measures to describe the accuracies of the classifiers and combinations:

1. error rate of the decision tree classifier, estimated with a two-fold cross validation with 10 random splits;
2. error rate of the subsampling decision forests using 100 trees, estimated with a two-fold cross validation with 10 random splits;
3. error rate of the subspace decision forests using 100 trees, estimated with a two-fold cross validation with 10 random splits;
4. % improvement of subsampling decision forests over single trees (reduction in averaged error rate normalized by the single tree averaged error rate);
5. % improvement of subspace decision forests over single trees (reduction in averaged error rate normalized by the single tree averaged error rate);

The training set sizes are not uniform and some of them are rather small, so the observed differences in accuracies between classifiers were tested for statistical significance. We used a paired comparison procedure as follows. For each training set and each of the three classifiers (single tree, subsampling forest, and subspace forest), the error rate on each of the (same) 10 random splits is calculated. To test if two classifiers are significantly different, we calculate the 10 differences d_i (i=1..10) from which we compute estimates of the mean $\bar{d}$ and standard error of the mean $s/\sqrt{10}$ (where s is the

standard deviation among the 10 differences). These are used to test against a null hypothesis that the true mean of the differences is zero via the statistic

$$t = \frac{\bar{d} - 0}{(s/\sqrt{10})}$$

that follows a Student's-t distribution with 9 degrees of freedom. The two classifiers are said to differ significantly if the p-value of t is smaller than a two-sided significance level of 0.05 (2.262).

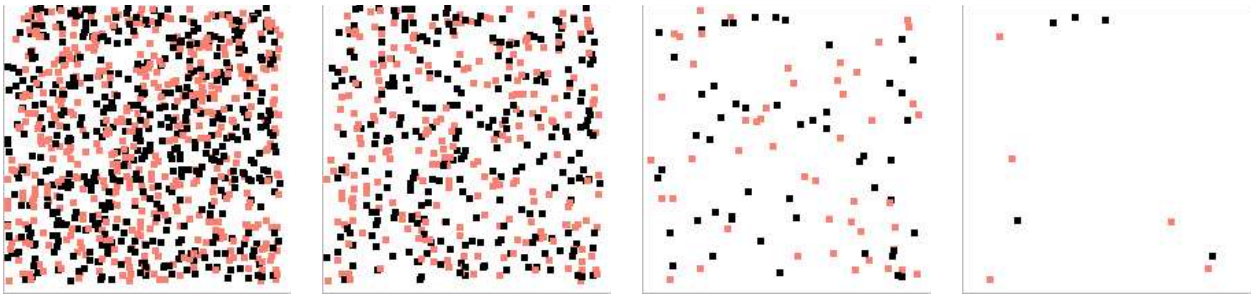
Using this procedure, among the 392 problems, for 49 there are no significant accuracy differences by either type of forests from that of a single tree. For 78 problems, at least one of the forests is significantly different from a single tree but there is no significant difference between the two types of forests. For 66 problems, subsampling forests are found to be significantly better than subspaces, and for 199 problems, subspace forests are found to be better than subsampling.

6 Results and Discussions

Figure 2(a) shows a scatter plot of the percent improvement in accuracy given by the subspace or subsampling forests over a single tree. Cases are marked with different symbols based on which of the following groups they belong to: (1) cases where a forest is no better than a single tree (49 cases, marked with crosses); (2) cases where either type of forests improves over a single tree but the two types do not differ significantly (78 cases, marked with circles); (3) cases where a subsampling forest is better (66 cases, marked with squares); and (4) cases where a subspace forest is better (199 cases, marked with triangles). The definitions of these markings are maintained throughout all the plots displayed in this section. The four groups are most visible as four partially overlapping clusters in Figure 2(a).

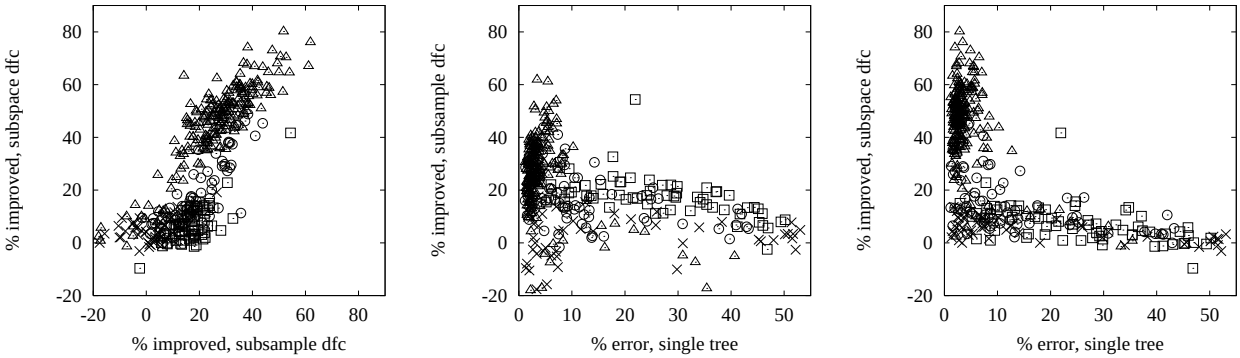
Table 1 shows the averaged values of various complexity and accuracy measures from each of the four groups. Comparing groups (1) and (2), we see that the most obvious difference is in training set sparsity: group (2) has denser training sets relative to dimensionality. A smaller boundary size also helps make these problems easier and thus yielding lower error rates. At the same time a higher percentage of retained adherence subsets means that all the three groups (2),(3), and (4) contain thinner class manifolds. Other than that, groups (1) and (3) seem to share most characteristics, only with crucial differences on the nonlinearity measures. Both of these two groups contain the more difficult cases, but subsampling forests are capable of improving on single trees when the decision boundary is rough (least smooth). Group (4) contains most distinguishing characteristics such as densest training sets, smallest boundaries, and lowest nonlinearities. They are the easier problems but interestingly accuracies can still be improved with subspace forests. Arguably these are just overall impressions given by the averaged values, and individual cases vary in all directions along these factors. We will examine these factors in greater details in the following plots.

Next we look closer at the groups (2),(3), and (4) where a combined classifier has an advantage. From Figure 2(b) and (c) we can see that these cases cover the full range of error rates incurred by a single tree. Due to the way we calculate the improvement (((Error (tree) - Error (forest))/ Error (tree))$\times$ 100%) it is understandable that when the single tree error rate is low, a larger improvement is observed. But it is worth mentioning that even in some very difficult cases where a single decision tree has close to 50% error, a forest can still cut down the error by 10%. Comparing



(a) 1000 points (b) 500 points (c) 100 points (d) 10 points

Figure 1: A set of randomly located points labeled randomly with two classes. The problem appears simpler as the sample size is reduced.



(a) (b) (c)

Figure 2: (a) Four groups of cases defined by comparative advantages of a particular forest contruction method (×: no forest improvements; ○: a forest improves but no difference between two types; □: subsampling forests are better; △: subspace forests are better); (b) Subsample forest improvement versus error rate of single tree; (c) Subspace forest improvement versus error rate of single tree.

Group	(1)	(2)	(3)	(4)
No. of cases	49	78	66	199
No. of pts/No. of dim.	36.01	61.36	38.88	92.97
(%) pts on boundary	25.59	16.96	38.46	3.80
(%) pts with retained ad. subset	85.21	93.54	96.50	95.13
Intra/Inter class NN dist	0.44	0.43	0.66	0.33
Max. Fisher's discrim. ratio	2.58	1.87	0.49	2.57
Nonlinearity of NN classifier (%)	8.67	6.30	16.17	2.34
Nonlinearity of linear classifier (%)	9.78	6.44	17.15	1.78
Error, single tree (%)	18.06	11.77	26.00	4.42
Error, subsamping forest (%)	17.46	10.19	21.96	3.51
Error, subspace forest (%)	17.64	10.47	24.65	2.81
Markers in figures	×	○	□	△

Table 1: Averaged values of various statistics by four performance groups.

	dataset, class pair			
measure	car, acc vs good	kr-vs-kp, nowin vs won	nursery, priority vs spec_prior	splice, IE vs N
No. of dimensions	21	73	27	480
No. of points	453	3196	4000	2423
No. of pts/No. of dim.	21.57	43.78	148.15	5.05
(%) pts on boundary	33.11	20.49	22.90	32.15
(%) pts with retained ad. subset	100.00	100.00	100.00	99.92
Max. Fisher's discrim. ratio	0.47	0.54	0.38	3.04
Intra/Inter class NN dist	0.87	0.72	0.96	0.87
Nonlinearity of NN classifier (%)	0.0	0.0	0.0	0.02
Nonlinearity of linear classifier (%)	0.93	0.79	5.14	0.01
Error, single tree (%)	8.96	7.32	7.05	3.34
Error, subsampling forest (%)	7.93	4.32	3.24	3.03
Error, subspace forest (%)	7.31	4.35	2.50	2.66
(%) Improvement by subsampling forest	11.58	41.05	53.98	9.38
(%) Improvement by subspace forest	18.47	40.58	64.57	20.49

Table 2: Four easy cases despite high fractions of boundary points.

the locations of the $\triangle$'s and $\square$'s in these two plots, we can see that the subsampling forests tend to be better than the subspace forests when the problem is more difficult for a single tree.

Now we try to relate the error rates and their differences to the complexity measures. First we examine the effects of boundary size measured by percentage of number of points lying on the class boundary. Figure 3(a)(b) display its effects on single tree error rates. From these plots we can see there is almost a linear correlation between the fraction of boundary points and the single tree error rate. But there are notable exceptions. In particular, four cases seem to be outside the general trend, and for them the decision tree works well despite a high fraction of points lying on the boundary. We show more details about these four cases in Table 2. Checking on values such as number of points, number of dimensions, % points with retained adherence subsets, maximum Fisher's discriminant ratio, and especially the nonlinearity of the nearest neighbor and linear classifiers, we find that these problems have large training sets and reside in a high dimensional space, such that although there is a long boundary, the boundary is smooth (low nonlinearity of both nearest neighbor and linear classifiers) and the convex hulls of the two classes are well separated (low nonlinearity of linear classifiers). In all these cases forest accuracies are better than single trees. In two out of the four cases subspace forests are preferable and in the remaining cases the two types of forests do not differ significantly.

In Figure 3(a)(b) we can also observe a difference between the subspace and subsampling forests. Most notably in Figure 3(b) where the axes are on logarithmic scales, there is a group of cases with low fraction of boundary points and for which subspace forests perform better. Tracing those cases in Figure 3(c), we find that those cases typically contain large number of samples relative to feature space dimensionality. These suggest that with a dense training set and well separated classes (small boundary size), subspace method is preferable and subsampling method does not offer an advantage. For almost all those cases the subspace forests yield a large improvement over single trees (on average, 49% reduction in error rate).

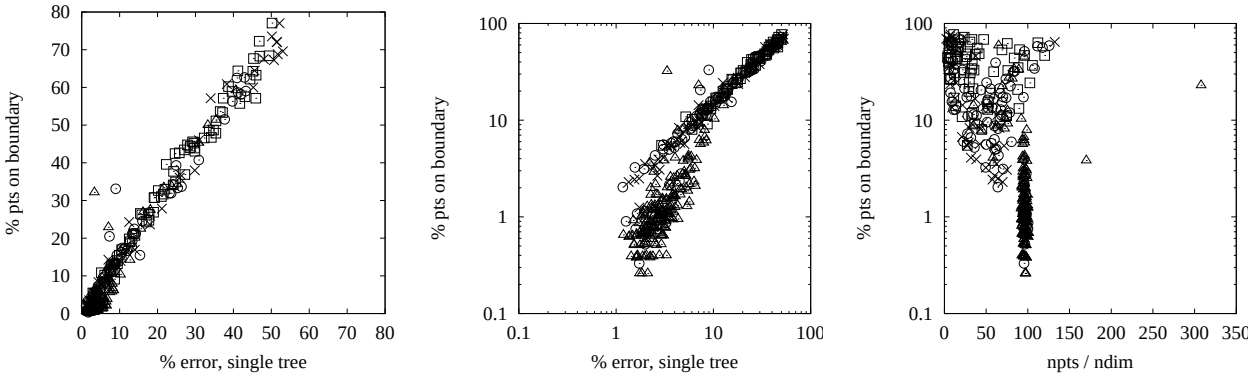
A similar analysis on the maximum Fisher's discriminant ratio versus fraction of boundary points and training set density (4(a)-(c)) reveals that the comparative advantages of the two forest construction methods tend to correlate strongly with these measures. This is most obvious in 4(a) where along the y-axis are the cases for which the subspace forests excel, and along the x-axis are winning cases of subsampling forests. That is, when the classes are well separated (larger Fisher's ratio) and when training sample is dense (larger number of points per dimension), subspace forests perform better than the subsampling forests. This distinction is also visible in 4(b) on logarithmic scales. In 4(c) we see a confirmation that subsampling forests are better when the classes are interleaved (extremely low Fisher's ratio) and when the training set is sparse relative to dimensionality. Cross-checking with Figure 3(c), we find that the few exceptions with dense samples for which subsampling forests are better (the boxes towards the lower-right corner of the left half of Figure 4(c)) correspond to the cases where the class boundary is long (more than 30% of all points are on boundary). The more so, the subsampling forests become better.

Some relatively easy problems can be found from the pretopological measure of % retained adherence subsets. Figure 5(a)-(c) show the values together with the single tree error rate and improvements by each type of forests. Those with lower values are cases with more training samples collected within the same hyperspherical clusters. Understandably they are easier problems as the correct decision regions are more compact. They invariably have low error rates by a single tree, and for most of these cases forests are not significantly better than single trees. Cases with a high fraction of adherence subsets retained are those with long and thin class distributions, so that as a hypersphere centered at a training sample grows, it will likely touch a point of the other class before touching another point of the same class. For many of these cases, both types of forests can yield large improvements, and cross-checking with plots in Figure 5(d)-(f) shows that those more improvable cases typically have larger training sets relative to dimensionality.

Figure 6(a)-(i) show more plots of selected measures. In these as well as all previous plots, we can observe that there is substantial spread among the collection of problems in terms of each combination of measures, which means that a problem's complexity needs to be described jointly by several factors. Further checking of these plots confirms that the subspace method excels in cases when the class distributions are more compact (low pretopological measure and low ratio of intra/inter class nearest neighbor distances) and the training sets are dense (a)(b)(c)(h). Also it can be seen that the boundary size together with the ratio of intra/inter class nearest neighbor distance are the most effective predictors of whether accuracy can be improved by forest methods (g). Nonlinearity of either the nearest neighbor or the linear classifier also indicates strongly the difficulty of a problem as shown by their correlation to single tree error rates (j)(k). Lower nonlinearity means a smoother boundary between the classes. It appears that subspace forests can better take advantage of such smoothness (l).

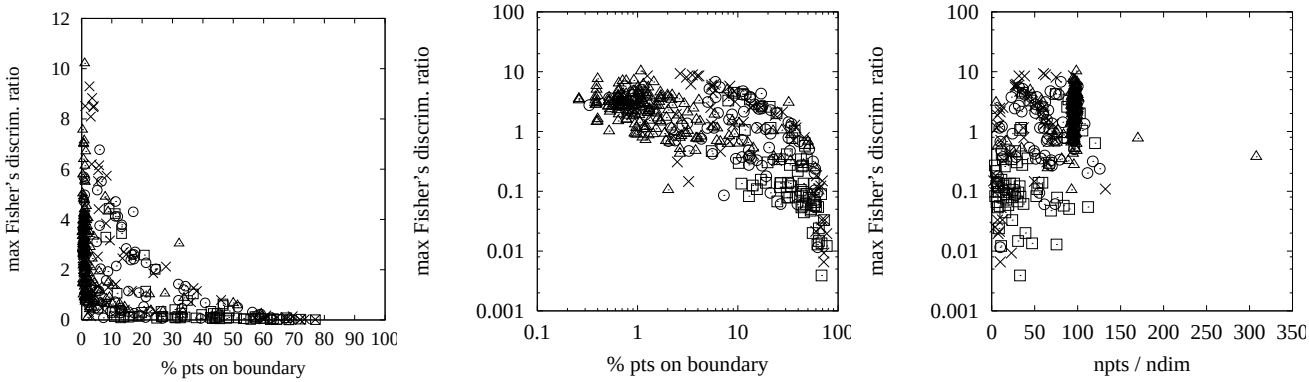
We summarize our main observations below.

- Common behavior of both types of forests:
  - Both types of forests are capable of improving over a single tree for problems of a large range of difficulties, i.e., improvements are observed in problems with very low, very high, and all intermediate single tree error rates.
  - Very difficult cases are identified by several conditions occurring jointly: very high fractions of boundary points (70% or above), a close to 1 ratio of intra/inter class nearest



(a) (b) (c)

Figure 3: (a) % Points on boundary versus single tree error rate; (b) same plot with both axes on log scale; (c) (log) % points on boundary versus average number of points per dimension.



(a) (b) (c)

Figure 4: (a) Maximum Fisher's discriminant ratio versus % points on boundary; (b) same plot with both axes on log scale; (c) (log) maximum Fisher's discriminant ratio versus number of points per dimension.

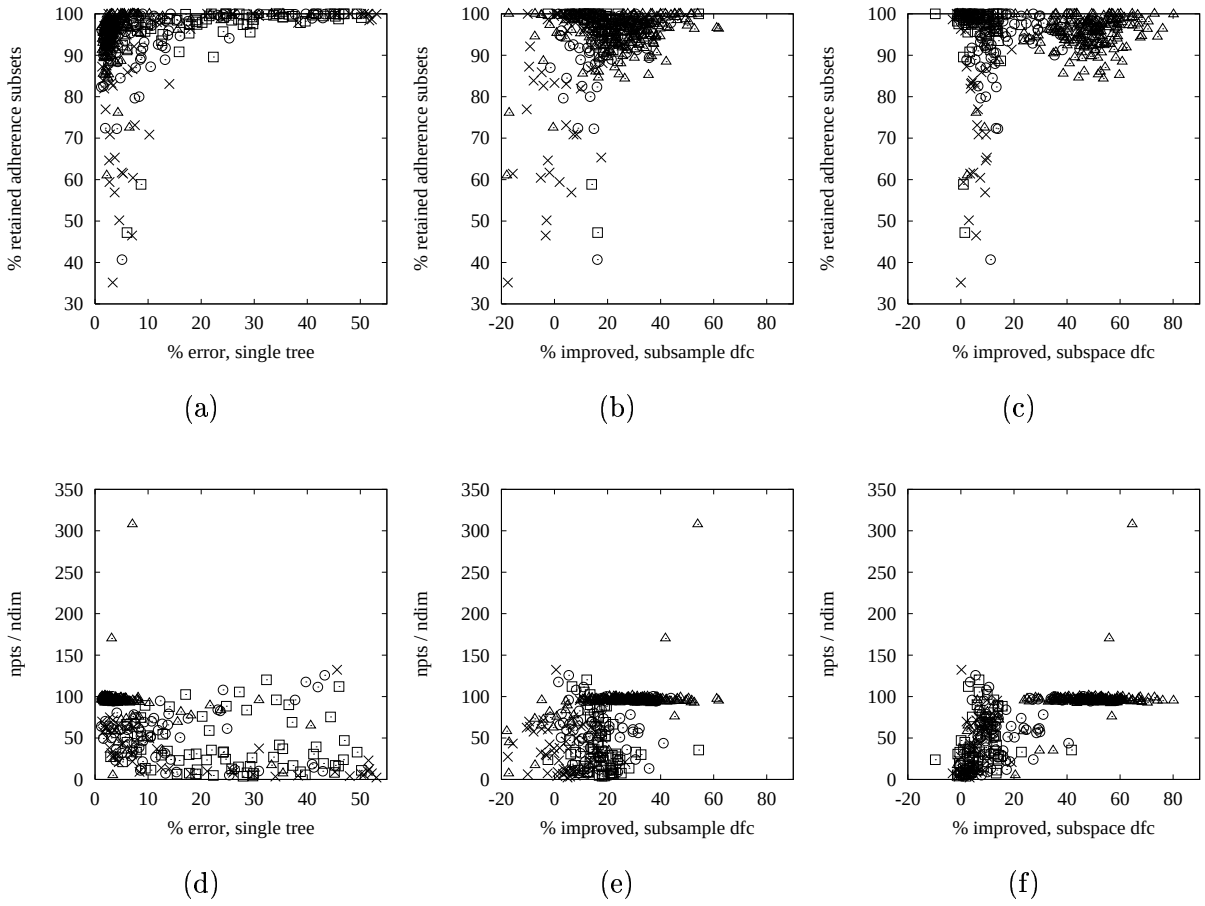


Figure 5: % Retained adherence subsets versus (a) single tree error rate; (b) improvement by subsampling forests; (c) improvement by subspace forests. Average number of points per dimension versus (d) single tree error rate; (e) improvement by subsampling forests; (f) improvement by subspace forests.

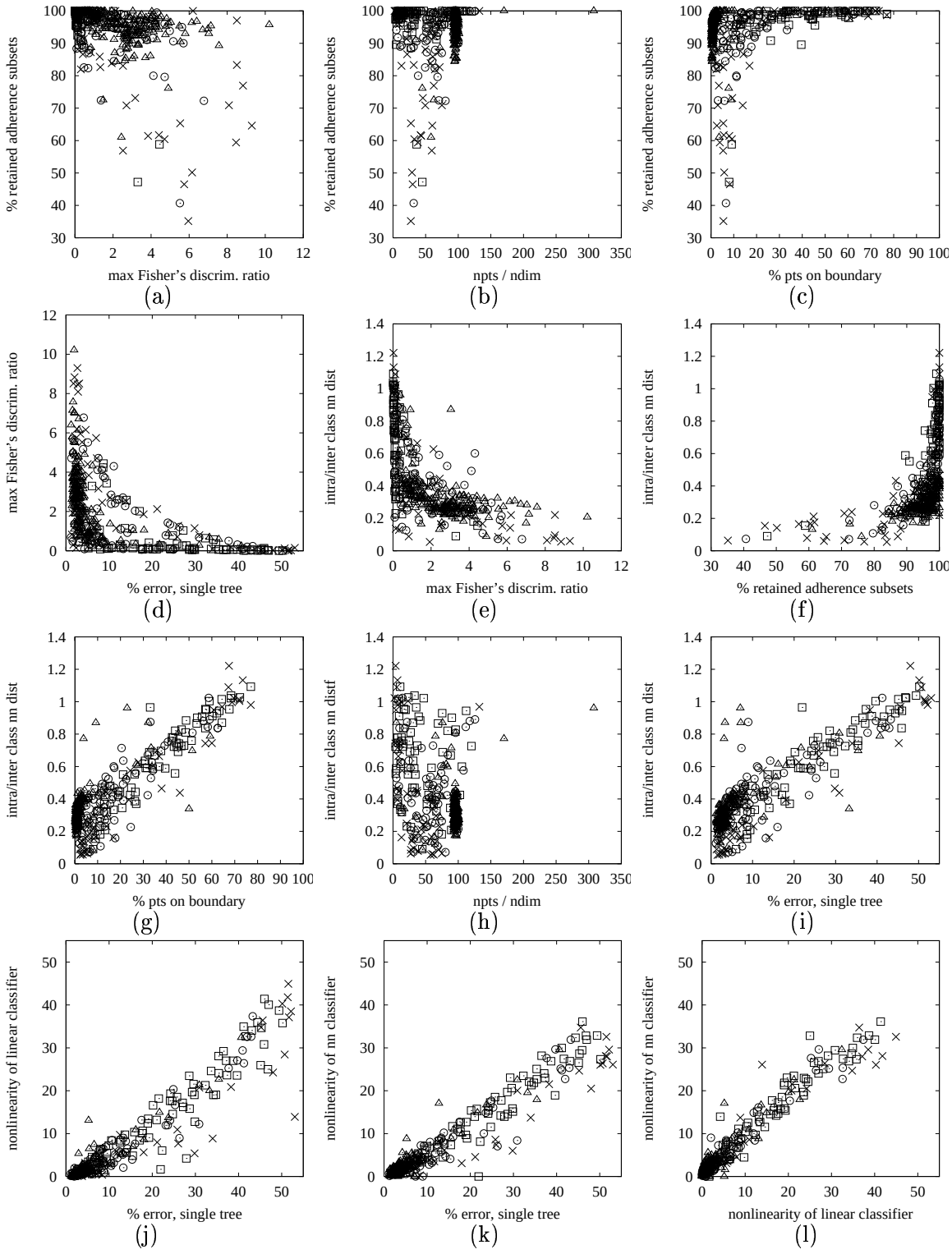


Figure 6: % Retained adherence subsets versus (a) maximum Fisher's discriminant ratio, (b) average number of points per dimension, and (c) % points on boundary. (d) Maximum Fisher's discriminant ratio versus single tree error rate. Intra/inter class nearest neighbor distances versus (e) maximum Fisher's discriminant ratio, (f) % retained adherence subsets, (g) % points on boundary, (h) average number of points per dimension, and (i) single tree error rate. (j) Nonlinearity of linear classifier versus single tree error rate. Nonlinearity of nearest neighbor classifier versus (k) single tree error rate, and (l) nonlinearity of linear classifier.

neighbor distances, very low maximum Fisher's discriminant ratio (0.05 or below), and high nonlinearity of both nearest neighbor and linear classifiers (25% or above). For these cases a single tree performs poorly, and forest methods do not offer much help.

- Easier cases are those with relatively compact classes, i.e., when the pretopological measure is lower than 80%. For these cases improvements by forests over single trees are less certain.

- Comparative advantages of each type of forests:
  - The subsampling method is preferable when the training set is very sparse relative to dimensionality, especially when coupled with a close-to-vanishing maximum Fisher's ratio (0.3 or below), and when the class boundary is highly nonlinear.
  - Subspace forests perform better when the class boundary is smoother (both nearest neighbor and linear classifiers display low nonlinearity).
  - If the training set is large relative to dimensionality, the subspace method is more preferable, even if the class distributions are long and thin.

7 Conclusions

We presented some empirical observations of the relationship between classifier and combined classifier accuracies and several measures of problem complexity. We conclude that there exist obvious dependences of classifiers' behavior on those measurable data characteristics. Such dependencies can serve as a guide for the expectation and direction of future efforts for optimizing classifiers and their combinations.

Detailed examinations of the dependences confirm the multi-facet nature of a problem's complexity, in the sense that exceptions are found to the apparent rules relating each single complexity measure to accuracy and improvements. Thus we believe that our approach of complexity characterization is useful and could even be mandatory for such studies before a more powerful language is developed to describe the geometrical, topological, and combinatorical characteristics of multidimensional point sets.

While we have not tried to exclude problems with any special characteristics, this collection is nevertheless small and may not be representative. On the other hand it is impossible to have a representative collection of realistic problems because of the ill-definition of the set. Therefore the evidence we presented regarding classifier behavior is existential by nature and is not conclusive even in a statistical sense. It will be interesting if future studies turn up exceptions to the apparent rules observed from this collection.

It should also be noted that for many of these measures we have used the number of training samples for normalization. In most of these problems the sample size is determined by convenience or resource limitations rather than by a rigorous sampling rule. As a result, the sampling density may be very different across different problems, which may introduce a hidden source of variance in the values of the measures.

There are other potentially useful measures of problem complexity. Some of those are investigated in [7][17]. Also, there are other ways of generating multiple classifiers that are also important. One

is a localized scheme, where the feature space is partitioned and a different classifier or combination function is derived in each element of the partition. The partitions can be computed using some characteristics of the input or the classifiers' decisions [3] [19] [24] [35] and the procedure can be performed systematically. Others involve a sequential optimization that search for classifiers that can complement existing ones (e.g. boosting [9]). It will be interesting future work to continue this analysis on such methods.

Acknowledgements

The author thanks George Nagy and Don X. Sun for helpful discussions.

References

[1] Basu M., Ho TK., The learning behavior of single neuron classifiers on linearly separable or nonseparable input, *Proceedings of the 1999 International Joint Conference on Neural Networks*, Washington, DC, July 1999.

[2] Berlind R., *An Alternative Method of Stochastic Discrimination with Applications to Pattern Recognition*, Doctoral Dissertation, Department of Mathematics, State University of New York at Buffalo, 1994.

[3] Bottou L., Vapnik V., Local learning algorithms, *Neural Computation*, **4**, 6, 1992, 888-900.

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

[5] Devroye L., Any discrimination rule can have an arbitrarily bad probability of error for finite sample size, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **4**, 2, March 1982, 154-157.

[6] Devroye L., Automatic pattern recognition: a study of the probability of error, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **10**, 4, July 1988, 530-599.

[7] Duin RPW., Compactness and Complexity of Pattern Recognition Problems, in: Perneel C. (eds.), *Proc. Int. Symposium on Pattern Recognition, In Memoriam Pierre Devijver*, Royal Military Academy, Brussels, Feb 12, 1999, 124-128.

[8] Foley DH., Considerations of sample and feature size, *IEEE Transactions on Information Theory*, **18**, 5, September 1972, 618-626.

[9] Freund Y., Schapire RE., Experiments with a New Boosting Algorithm, *Proceedings of the Thirteenth International Conference on Machine Learning*, Bari, Italy, July 3-6, 1996, 148-156.

[10] Friedman JH., Rafsky LC., Multivariate generalizations of the Wald-Wolfowitz and Smirnov two-sample tests, *The Annals of Statistics*, **7**, 4, 1979, 697-717.

[11] Fukunaga K., Kessell DL., Estimation of classification error, *IEEE Transactions on Computers*, **20**, 12, December 1971, 1521-1527.

[12] Hand, DJ., Recent advances in error rate estimation, *Pattern Recognition Letters*, **4**, October 1986, 335-346.

[13] Ho TK., Random decision forests, *Proceedings of the 3rd International Conference on Document Analysis and Recognition*, Montreal, August 14-18, 1995, 278-282.

[14] Ho TK., The random subspace method for constructing decision forests, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **20**, 8, August 1998, 832-844.

[15] Ho TK., Baird HS., Large-scale simulation studies in image pattern recognition, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **19**, 10, October 1997, 1067-1079.

[16] Ho TK., Baird HS., Pattern classification with compact distribution maps, *Computer Vision and Image Understanding*, **70**, 1, April 1998, 101-110.

[17] Ho TK., Basu M., Measuring the Complexity of Classification Problems, *Proceedings of the 15th International Conference on Pattern Recognition*, Barcelona, September 3-8, 2000, 43-47.

[18] Hoekstra A., Duin RPW., On the nonlinearity of pattern classifiers, *Proc. of the 13th ICPR*, Vienna, August 1996, **D**271-275.

[19] Ho TK., Hull JJ., Srihari SN., Decision Combination in Multiple Classifier Systems, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **16**, 1, January 1994, 66-75.

[20] Kanal L., Chandrasekaran B., On dimensionality and sample size in statistical pattern classification, *Pattern Recognition*, **3**, 1971, 225-234.

[21] Kittler J., Devijver PA., Statistical properties of error estimators in performance assessment of recognition systems, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **4**, 2, March 1982, 215-220.

[22] Kleinberg EM., An overtraining-resistant stochastic modeling method for pattern recognition, *Annals of Statistics*, **4**, 6, December 1996, 2319-2349.

[23] Lebourgeois F., Emptoz H., Pretopological approach for supervised learning, *Proceedings of the 13th International Conference on Pattern Recognition*, Vienna, 1996, 256–260.

[24] Lee DS., *A Theory of Classifier Combination: The Neural Network Approach*, Doctoral Dissertation, Department of Computer Science, State University of New York at Buffalo, 1995.

[25] Li M., Vitanyi P., *An Introduction to Kolmogorov Complexity and Its Applications*, Springer-Verlag, 1993.

[26] Maciejowski JM., Model discrimination using an algorithmic information criterion, *Automatica*, **15**, 1979, 579-593.

[27] Mehrotra KG., Mohan CK., Ranka S., Bounds on the number of samples needed for neural learning, *IEEE Transactions on Neural Networks*, **2**, 6, November 1991, 548-558.

[28] Murthy S., Kasif S., Salzberg S., A System for Induction of Oblique Decision Trees, *Journal of Artificial Intelligence Research*, **2**, 1, 1994, 1-32.

[29] Raudys S., On dimensionality, sample size, and classification error of nonparametric linear classification algorithms, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **19**, 6, June 1997, 667-671.

[30] Raudys S., Jain AK., Small sample size effects in statistical pattern recognition: Recommendations for practitioners, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **13**, 3, 1991, 252-264.

[31] Raudys S., Pikelis V., On dimensionality, sample size, classification error, and complexity of classification algorithm in pattern recognition, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **2**, 3, May 1980, 242-252.

[32] Sohn SY., Meta analysis of classification algorithms for pattern recognition, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **21**, 11, 1999, 1137-1144.

[33] Toussaint GT., Bibliography on estimation of misclassification, *IEEE Transactions on Information Theory*, **20**, 4, July 1974, 472-479.

[34] Vapnik V., *Statistical Learning Theory*, John Wiley & Sons, 1998.

[35] Woods K., Kegelmeyer KP. Jr., Bowyer K., Combination of multiple classifiers using local accuracy estimates, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **19**, 4, 1997, 405-410.