Complexity of Classification Problems and Comparative Advantages of Combined Classifiers

Tin Kam Ho

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

Abstract. We studied several measures of the complexity of classification problems and related them to the comparative advantages of two methods for creating multiple classifier systems. Using decision trees as prototypical classifiers and bootstrapping and subspace projection as classifier generation methods, we studied a collection of 437 two-class problems from public databases. We observed strong correlations between classifier accuracies, a measure of class boundary length, and a measure of class manifold thickness. Also, the bootstrapping method appears to be better when subsamples yield more variable boundary measures and the subspace method excels when many features contribute evenly to the discrimination.

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 refer to combination strategies of the first kind as *decision optimization* methods and the second kind as *coverage optimization* methods.

Theoretical claims of optimality for single or combined classifiers are often dependent on an assumption of infinite sample size. In practice, limits in training data often mean that models and heuristics are needed to generalize decisions to unseen samples. Practical systems use specialized classifiers to include domain-specific heuristics in different ways, and their cooperation is thus desirable to provide a more balanced coverage of the domain. Such a balance is explicitly sought after in coverage optimization methods, though a heuristic choice is still needed to specify a family of component classifiers, such as specific kinds of kernels or trees. In either case, sparse training samples and the biases of the heuristics often leave the optimality goals unfulfilled.

Just like with a single classifier, when a combined system performs at suboptimal accuracy, the reasons are often unclear. Theoretically[4] and empirically, there is a general understanding that different types of data require different

kinds of classifiers. Similar arguments can be made for 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 [20].

Deeper investigation of methods for creating multiple classifiers and their behavior points back to many fundamental questions in pattern recognition. Competence of classifiers is closely related to the sufficiency of the chosen features and metrics for separating the classes. Generalization ability and independence of classifiers also closely resemble similar properties of individual features. This suggests that studies of behavior of combined classifiers will benefit from a better understanding of the nature of difficulties of a recognition problem. In this paper we attempt to characterize such difficulties and their effects on classifier behavior.

2 Sources of Error of a Classification Problem

We begin with a brief analysis of the sources of error of a classification problem. We isolate these sources for convenience of discussion, though we understand that real world problems often include difficulties from more than one sources. For simplicity we assume that each problem is on discrimination of two classes. Multiclass problems are reduced to dichotomies, and we note that it is not trivial to recombine such pairwise decisions to a final decision. We assume that a problem is defined with a given set of samples described as points in a vector (feature) space, and each point is labeled with a class from a given set.

Class ambiguity. Some problems are known to have nonzero Bayes error [9]. That is, samples of two different classes may have identical feature values. This can happen regardless of the shape of the class boundary and feature space dimensionality. While certain problems may be intrinsically ambiguous, others may be so because of poor feature selection. The ambiguity is intrinsic if the given features are complete for reconstruction of the patterns. Otherwise, it is possible that the ambiguity can be removed by redefining the features. The Bayes error is a measure of difficulty in this aspect and it sets a lower bound on the achievable error rate.

Imperfectly modeled 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) [14]. An example is a set of randomly located points arbitrarily labeled as one of two classes. Classifiers need to have a matching capacity to model the boundary, otherwise error will occur. This is independent of sampling density, class ambiguity, and feature space dimensionality.

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. This happens easily in a high dimensional space where 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 [22] and is also discussed in Kleinberg's arguments on M-representativeness [12], Berlind's hierarchy of indiscernibility [2], and Raudys' and Jain's practical considerations [18]. It is also discussed in many studies of error rate estimation [6] [11] [21].

Imperfect accuracy of classifiers may be due to a combination of these 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. Our discussion is to be qualified in the context of statistical estimation, as sample size constrains what can be learned about either class ambiguity or boundary complexity of a given problem.

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 may lead to unnecessarily weak results. Most classifier designs are based on 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 consider a number of measures 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.

Certain practical problems possess a structure or regularity so that there exists a transformation with which samples can be mapped to a new space where class discrimination becomes easier. The existence of such a transformation is not always obvious for an arbitrary problem (i.e., Kolmogorov complexity is not effectively computable[15]), so our discussion of problem complexity will be simplified by refering to a fixed, given feature space.

Length of Class Boundary

To measure the length of the boundary between two classes, we consider a method proposed by Friedman and Rafsky [5]. Given a metric, a minimum spanning tree (MST) is constructed that connects all sample points regardless of class. Thus some edges will connect points belonging to two different classes. The length of boundary is then given as a count of such edges (Figure 1(a)). With n points there are $n-1$ edges in the MST, so the count can be simply normalized as a percentage of n.

This measure is sensitive to both the separability of the classes and the clustering properties 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 dense and close within each class.

Space Covering by ϵ-Neighborhoods

The local clustering properties of a point set can be described by an ϵ-neighborhood pretopology [13]. Here we consider a reflexive and symmetrical binary relation $\mathcal{R}$ of two points x and y in a set F. $\mathcal{R}$ is defined by $x\mathcal{R}y \Leftrightarrow d(x,y) < \epsilon$, where $d(x,y)$ is a given metric and ϵ is a given nonzero constant. Define $\Gamma(x) = \{y \in F | y\mathcal{R}x\}$ to be the $\epsilon - neighborhood$ of x, an adherence mapping ad from the power set $\mathcal{P}(F)$ to $\mathcal{P}(F)$ is such that

$$\begin{cases} ad(\phi) = & \phi \\ ad(\{x\}) = & \{x\} \cup \Gamma(x) \\ ad(A) = & \bigcup_{x \in A} ad(\{x\}) \qquad \forall A \subseteq F. \end{cases}$$

Adherence subsets can be grown from a singleton $\{x\}$: $\{x\} = ad^0(\{x\})$, $ad(\{x\}) = ad^1(\{x\})$, ..., $ad(ad^n(\{x\})) = ad^{n+1}(\{x\})$, where j is called the *adhesion order* in $ad^j(\{x\})$. From a point of each class one can grow successive adherence subsets to the highest order n such that $ad^n(\{x\})$ includes only points of the same class but $ad^{n+1}(\{x\})$ includes points of other classes.

We keep for each point only the highest order n adherence subset such that all elements of $ad^n(\{x\})$ are within the class of x, and eliminate any adherence subsets that are strictly included in others. Using the ϵ-neighborhoods with Euclidean distance as d, each retained adherence subset associated with a point is the largest hypersphere that contains it and no points from other classes, in units of the chosen ϵ (Figure 1(b)). In our experiments we used $\epsilon = 0.55\delta$ where δ is the distance between two closest points of opposite classes. We chose 0.55 arbitrarily just so that it is larger than 0.5 and the lowest adhesion order is always zero, occurring at the points closest to the class boundary.

A list of such ϵ-neighborhoods needed to cover the two classes is a composite description of the shape of the classes. The count and order of the retained adherence subsets show the extent to which the points are clustered in hyperspheres or distributed in thinner structures. In a problem where each point is closer to points of the other class than points of its own class, each adherence subset is retained and is of a low order. We normalize the count by the total number of points.

Feature Efficiency

With a high dimensional problem we are concerned about how the discriminatory information is distributed across the features. Here we consider a measure of

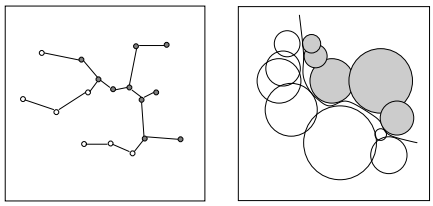


Fig. 1. (a) A minimum spanning tree where the thicker edges connect two classes. (b) Retained adherence subsets for two classes near the boundary.

efficiency of individual features that describes how much each feature contributes to the separation of the two classes [10].

We consider a local continuity heuristic such that all points of the same class may have values of each feature falling between the maximum and minimum of that class. If there is an overlap in the feature values of two classes, we consider the classes ambiguous in that region along that dimension. Given that, a problem is easy (i.e. linearly separable) if there exists one feature dimension where the ranges of values spanned by each class do not overlap. For other problems that are globally unambiguous, one may progressively remove the ambiguity between the two classes by separating only those points that lie outside the overlapping region in each chosen dimension.

We define the *individual feature efficiency* to be the fraction of all remaining points separable by that feature. *Maximum feature efficiency* is the maximum individual feature efficiency computed using the entire point set. In this procedure we consider only separating hyperplanes perpendicular to the feature axes, i.e., joint effects of the features are not accounted for.

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. Two typical methods of this category construct a collection of classifiers systematically by varying either the training samples or the features used. These are respectively represented by the bootstrapping (or *bagging*) method [3] and the random subspace method [7][8]. In bagging, subsets of the training points are independently and randomly selected 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 training points are projected onto a randomly chosen coordinate subspace in which a classifier is derived. 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 (at different levels of the tree) and parallel (with different trees) combinations of arbitrary discriminators. Decisions at the internal nodes can be simple

splits on a single feature or any other linear or nonlinear discriminators. Other classifiers are special cases of decision forests.

In this study the decision trees use oblique hyperplanes to split the data at each internal node [16]. The hyperplanes are derived using a simplified Fisher's method [7], i.e., by looking for the error minimizing hyperplane perpendicular to a line connecting the centroids of two classes. Assuming no class ambiguity, the tree can always be fully split, and trees constructed this way are usually small.

5 The Collection of Problems

We investigated these complexity measures and classifiers' behavior using two collections of problems. The first consists of 14 datasets from the UC-Irvine Machine Learning Depository, selected from those containing at least 500 points and no missing values: *abalone, car, german, kr-vs-kp, letter, lrs, nursery, pima, segmentation, splice, tic-tac-toe, vehicle, wdbc, and yeast.* Categorical features were all numerically coded. With each dataset, we took every pair of classes to be a problem. Of the 844 two-class problems, 452 were found to be linearly separable[1]. We used the remaining 392 problems in this study. The second collection consists of 10,000 handwritten digit images from the NIST special database 3. There are 1000 images for each of the 10 digits, and we took each pair of digits as an individual problem. Each image is represented as gray levels in [-8,8] in 28×28 pixels, so the feature space has 784 dimensions. Despite the dimensionality, all these 45 problems were found to be linearly separable – recall that this is an apparent complexity from limited samples.

6 Results and Discussions

Figure 2 shows some pairwise scatter plots of the complexity and accuracy measures. The measures we studied include:

1. error rate of the 1-nearest-neighbor classifier using Euclidean distance, estimated by the leave-one-out method;
2. error rate of the decision tree classifier, estimated by two-fold cross validation with 10 random splits;
3. error rate of the subsampling decision forests using 100 trees, estimated by two-fold cross validation with 10 random splits;
4. error rate of the subspace decision forests using 100 trees, estimated by two-fold cross validation with 10 random splits;
5. percent improvement of subsampling decision forests over single trees (reduction in error rate normalized by the single tree error rate);
6. percent improvement of subspace decision forests over single trees (reduction in error rate normalized by the single tree error rate);
7. percent points on boundary estimated by the MST method;
8. percent points with associated adherence subsets retained; and
9. maximum (individual) feature efficiency.

Figure 2 (a)-(d) plot the error rates of 1NN, decision tree, subsampling forests and subspace forests against % points on boundary estimated by the MST method. In each plot we see a strong positive correlation and an almost linear relationship, which suggests that % points on boundary is a good indicator of problem difficulty. This measure seems to set an accuracy limit on both the individual and combined classifiers. Say, if over 50% of the points are on boundary, none of the classifiers can get to lower than 20% error rate. So no matter how much gain we may have by combining multiple classifiers of this kind and in these ways, we should not expect them to get beyond this intrinsic limit determined by the data. Though, a few off-diagonal points in each plot suggest that accuracy is affected by some other factors as well.

Figure 2 (e)-(h) plot the same error rates against % points with associated adherence subsets retained. In these plots, we see that problems with few adherence subsets retained are easier for each classifier. All those problems with a high error rate have close to 100% points retaining their own highest order adherence subset, though the converse is not necessarily true. This suggests that difficult problems tend to have classes forming long and thin structures along the class boundary. Combining with the plots in Figure 2 (a)-(d), we can see that those thin structures may still be dense enough that few points are closer to the opposite class than other members of their own (those problems have many retained adherence subsets but very few points on cross-boundary edges of the MST). The classifiers perform well on such cases.

Figure 2 (i)(j) plot the % improvement (reduction in error rate) of the decision forests over the single tree classifiers against the maximum feature efficiency. We observe (in (j)) that the subspace method tends to have larger improvement for those problems when the maximum feature efficiency is lower (an indication that the contribution to discrimination is more evenly distributed across the feature dimensions). Plot (i) shows that the subsampling method is better for some cases with higher maximum feature efficiency, but for other such cases it can also be worse than the single tree classifier. Notice that these improvement measures are normalized by the error rate of the single tree classifier, so that a large percentage (either positive or negative) may happen when the single tree error rate is small. Plot (k) shows that for most cases that have thicker classes (less retained adherence subsets), the subsampling method does not offer a substantial advantage (say, over 20%) and may even be worse. Interestingly, for the same cases, though the subspace method also yields only minor improvements, it does not perform worse than the single tree classifier (plot (l)). Both (k) and (l) show that the combined systems offer significant improvements when the classes are thin, despite that those cases are hard for a single tree classifier ((g) and (h)).

Figure 2 (m)-(p) compare the single classifiers and combined classifiers against one another. In (m) we see that the two classifiers 1NN and decision tree differ by much for some cases (off diagonal points). In (n) we compare the subsampling method vs. the subspace method in terms of improvement over single trees, and we observe two off diagonal clusters suggesting that both types of cases exist for which one method is substantially better than the other. In (o) we notice

that for many cases when 1NN does not perform well, the subsampling method can yield an improved classifier. Compared to (p) we see that for those cases subsampling are also better than the subspace method. These correspond to the lower cluster in (n).

Finally, we calculate the standard deviation in the boundary measure as we take subsamples or subspace projections of the training set, and examine their correlation with the advantages of the decision forests. These are shown in Figure 2 (q)-(t). In (r) we see that the subspace method is good essentially when the boundary characteristics are similar in different subsamples (small standard deviation). For the more variable cases the subsampling method is preferable (q). When the subspace variation is small, there are more cases for which the subspace method yields over 40% improvement ((s) and (t)).

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 data characteristics. Such dependences may serve as a guide for the expectation and direction of future efforts for optimizing classifiers and their combinations.

Our main observations are:

- there exist complementary advantages between nearest neighbor and decision tree classifiers, as well as between subsampling and subspace methods for generating multiple classifiers;
- intrinsic characteristics of the data set a limit on achievable accuracy of either single or combined classifiers that are based on these simple geometrical models;
- classes with long boundaries or thin structures are harder for all these classifiers;
- when the discrimination power is dispersed across many features, the subspace method yields better improvement over single classifiers of the same type; and
- when the sample is very sparse so that subsamples yield more variable characteristics, the subsampling method yields better improvements.

We used the number of training samples to normalize many measures. 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.

While we have not tried to exclude problems with any special characteristics, this collection is nevertheless small and may not be representative. It will be interesting if future studies turn up exceptions to the rules observed from this collection. Future work will also need to answer methodological questions such as what is a reasonable collection of problems to study along these lines, and to what extent can we generalize these conclusions.

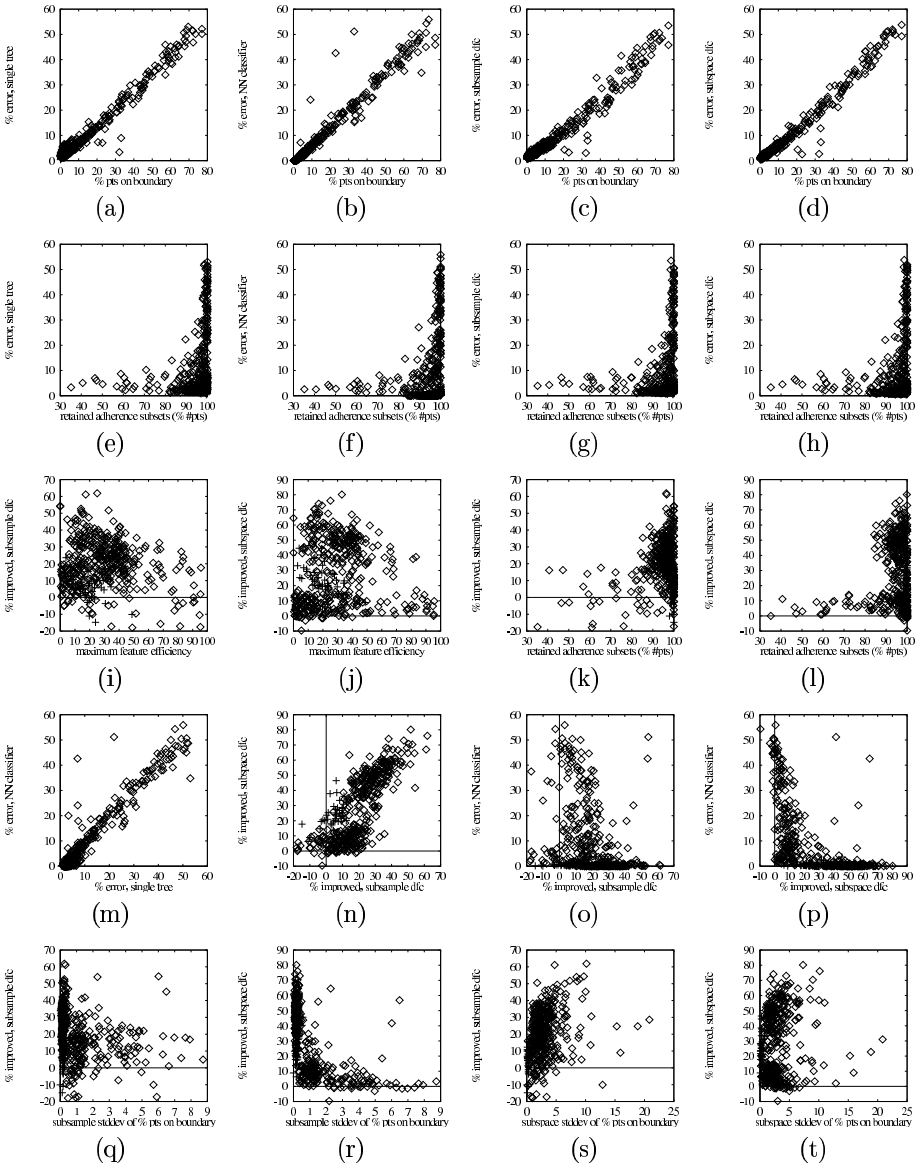


Fig. 2. Pairwise scatter plots of selected complexity and accuracy measures. In each plot, diamonds mark the 392 UCI problems and crosses mark the 45 NIST problems.

References

1. Basu, M., Ho, T.K., The learning behavior of single neuron classifiers on linearly separable or nonseparable input, *Proc. of the 1999 Int'l Joint Conf. on Neural Networks*, Washington, DC, July 1999.
2. Berlind, R., *An Alternative Method of Stochastic Discrimination with Applications to Pattern Recognition*, Doctoral Dissertation, Dept. of Mathematics, SUNY at Buffalo, 1994.
3. Breiman, L., Bagging predictors, *Machine Learning*, **24**, 1996, 123-140.
4. Devroye, L., Any discrimination rule can have an arbitrarily bad probability of error for finite sample size, *IEEE Trans. on PAMI*, **4**, 2, March 1982, 154-157.
5. Friedman, J.H., Rafsky, L.C., Multivariate generalizations of the Wald-Wolfowitz and Smirnov two-sample tests, *Annals of Statistics*, **7**, 4, 1979, 697-717.
6. Fukunaga, K., Kessell, D.L., Estimation of classification error, *IEEE Trans. on Computers*, **20**, 12, December 1971, 1521-1527.
7. Ho, T.K., Random decision forests, *Proc. of the 3rd Int'l Conf. on Document Analysis and Recognition*, Montreal, August 14-18, 1995, 278-282.
8. Ho, T.K., The random subspace method for constructing decision forests, *IEEE Trans. on PAMI*, **20**, 8, August 1998, 832-844.
9. Ho, T.K., Baird, H.S., Large-scale simulation studies in image pattern recognition, *IEEE Trans. on PAMI*, **19**, 10, October 1997, 1067-1079.
10. Ho, T.K., Baird, H.S., Pattern classification with compact distribution maps, *Computer Vision and Image Understanding*, **70**, 1, April 1998, 101-110.
11. Kittler, J., Devijver, P.A., Statistical properties of error estimators in performance assessment of recognition systems, *IEEE Trans. on PAMI*, **4**, 2, March 1982, 215-220.
12. Kleinberg, E.M., An overtraining-resistant stochastic modeling method for pattern recognition, *Annals of Statistics*, **4**, 6, December 1996, 2319-2349.
13. Lebourgeois, F., Emptoz, H., Pretopological approach for supervised learning, *Proc. of the 13th Int'l Conf. on Pattern Recognition*, Vienna, 1996, 256–260.
14. Li, M., Vitanyi, P., *An Introduction to Kolmogorov Complexity and Its Applications*, Springer-Verlag, 1993.
15. Maciejowski, J.M., Model discrimination using an algorithmic information criterion, *Automatica*, **15**, 1979, 579-593.
16. Murthy, S., Kasif, S., Salzberg, S., A System for Induction of Oblique Decision Trees, *J. of Artificial Intelligence Research*, **2**, 1, 1994, 1-32.
17. Raudys, S., On dimensionality, sample size, and classification error of nonparametric linear classification algorithms, *IEEE Trans. on PAMI*, **19**, 6, June 1997, 667-671.
18. Raudys, S., Jain, A.K., Small sample size effects in statistical pattern recognition: Recommendations for practitioners, *IEEE Trans. on PAMI*, **13**, 3, 1991, 252-264.
19. Raudys, S., Pikelis, V., On dimensionality, sample size, classification error, and complexity of classification algorithm in pattern recognition, *IEEE Trans. on PAMI*, **2**, 3, May 1980, 242-252.
20. Sohn, S.Y., Meta analysis of classification algorithms for pattern recognition, *IEEE Trans. on PAMI*, **21**, 11, 1999, 1137-1144.
21. Toussaint, G.T., Bibliography on estimation of misclassification, *IEEE Trans. on Information Theory*, **20**, 4, July 1974, 472-479.
22. Vapnik, V., *Statistical Learning Theory*, John Wiley & Sons, 1998.