

Choosing a Discernibility Measure for Reject-Option of Individual and Multiple Classifiers

Zacharias Voulgaris

Dept. of Electrical and Computing Engineering, Georgia Institute of Technology, Atlanta, GA 30332, USA, zacknv@gmail.com, and

Boris Mirkin*

School of Computer Science and Information Systems, Birkbeck University of London, Malet Street, London, WC1E 7HX, UK, mirkin@dcs.bbk.ac.uk, and

Department of Data Analysis and Machine Intelligence, State University – Higher School of Economics, Pokrovski Boulevard, Moscow, 101990, RF, bmirkin@hse.ru

Abstract

A novel method for evaluating the reliability of a classifier on a test pattern is proposed based on the discernibility of a pattern's class against other classes from the pattern's location. Use of three measures of discernibility is experimentally compared with more conventional techniques based on the classification scores for class labels. The discernibility measures can drastically improve the accuracy of a classifier on the most reliable – “elite” – patterns, which can be further boosted by forming an amalgamation of the elites of different classifiers. The improved performance is achieved with a price of rejecting many patterns. There are situations at which this price is worth paying – when the non-reliable accuracy rates lead to the need in manually testing of very complex technical devices or in diagnostics of human diseases. In contrast to conventional techniques for estimating reliability, the measures are applicable on small data sets. They can also work on data sets with complex class structures at which conventional classifiers would show low accuracy rates.

Keywords: pattern recognition, rejection, reliability elite, discernibility, combining classifiers

1. Introduction

The process of classification, or pattern recognition, involves mapping a group of testing patterns T onto specific classes S , using a training set of labelled patterns P . Since no classifier is perfect, errors

* Partial financial support of the Laboratory of Decision Choice and Analysis at SU-HSE is acknowledged.

may and do occur in the classification process, when some patterns are classified into wrong classes (misclassification). In many cases misclassification is costly. Moreover, there are situations in which classification with a low class label score can be costly too because of the need in testing the unreliable, even if correct, predictions by other, more expensive, means such as manual classification (Fumera et al., 2000) or developing a different feature set (Giusti et al., 2000) (e.g. in the cases of pre-screening cancer detection using medical images or complex technical devices being parts of important technical systems). In such a case, the option of rejection to classify a pattern is applied (Arlandis et al., 2002; Baram, 1998; Cordella et al., 1995; Duda et al., 2001; Fumera et al., 2000; Sansone et al., 2001; Santos-Pereira and Pires, 2005; Thien, 1996a). The principal approach in analyzing the trade-off between a potential error and rejection is formulated in terms of posterior probabilities of different classes: if the posterior probability of the best class label is high, then the label is attached to the pattern and reported; if not, the classification is not performed (reject-option); see, for example, Battiti and Colla (1994). The optimal value of the rejection threshold is provided by the so-called Chow's rule (Baram, 1998; Sansone et al., 2001; Santos-Pereira and Pires, 2005; Thien, 1996a, 1996b). Since in most applications the posterior probabilities are unknown, some other scoring indices can be used instead of the posterior probabilities. This approach was utilized several times by different authors, specifically, for classifiers based on neural networks in (Battiti and Colla, 1994; Cordella et al., 1995; Fumera et al., 2000; Sansone et al., 2001; Santos-Pereira and Pires, 2005) and K nearest neighbours in (Arlandis et al., 2002; Denoeux, 1995; Fumera et al., 2000; Giusti et al., 2002). Yet the major premise remains: the reject-option is defined in terms of the class label scoring function, not independently. This holds even in cases when the authors recognize limitations of the approach involving just one rejection threshold. Even when ROC curves, that are universal regarding classifiers, are used for rejection of certain patterns (Sansone et al., 2001), there are issues in the situations of more than two classes and of small training sets that may lead to unreliable ROC.

We propose an independent measure of classifier's reliability over an entity, which is applicable to any classifier on any data set. This measure relates to the "typicality" of an entity as a representative of its class, which is evaluated by the level of Discernibility of the class from the entity's location. We analyze three different scoring functions for the Discernibility measurement. One of them utilizes the proportion of the entity's class within the entity's neighbourhood. Two others are based on comparison of the average distances from the entity to entities of its class versus distances to entities of other classes. One of these other measures is the Silhouette Width coefficient (Kaufman and Rousseeuw, 1990), which is very popular in clustering but, to our knowledge, has been never applied in classification. The other measure relates the harmonic means of the within-class and out-of-class distances. Each of the Discernibility measures defines a reliability elite, the set of patterns with the highest Discernibility scores. We experimentally test our Discernibility scoring functions over their

elites. Then we use the elites for combining classifiers. Combining classifiers to make a better prediction is a common idea (see L. Kuncheva, 2005, and R. Polikar, 2006, for reviews). In our case, however, one can use the Discernibility scores as independent evaluations of classifiers' performances and combine classifiers in such a way that, at each entity, only classifiers with the highest Discernibility score are used.

Section 2 describes the set of classifiers used in this study along with their scoring functions. The measures of Discernibility are introduced in section 3. In section 4, the concept of reliability elite for a Discernibility measure is introduced and compared with more conventional reject-options. Then three rules for combining the individual classifier elites are introduced. Experimental comparison of the rules between each other and an averaging scheme by Battiti & Colla (1994) at five datasets from Irvine Machine Learning repository (UCI Repository) is presented in section 5. Conclusions are drawn in section 6.

3. Measuring Degree of Discernibility of Patterns

3.1. Degree of Certainty

Let us first consider an analogue to the Chow's index applied to any classifier's scoring function, for which Aidin and Guvenir coined the term Certainty Factor (CF) (Aydin and Guvenir, 2006). It is assumed that a classifier under consideration, for each pattern i and each class, produces a classification score, so that the class for which the score is maximal is predicted for i by the classifier. The relative proportion of this maximum classification score is referred to as the Degree of Certainty (DC):

$$DC_i = \frac{\max(\textit{classification score})}{\sum_{c=1}^{\# \textit{ of classes}} \textit{classification score}(c)} \quad (1)$$

where i denotes the i -th pattern classified, c the class label and *classification score* is the score determining the classification output of the classifier. This index could be used as a measure of reliability of classification in spite of the fact that it may provide a rather low correlation with the true class labels. Consider, for example, Table 1. In this table we compare the accuracy of a classifier on the entire training set (first column) with its accuracy on one third of all the training patterns, referred to as elite in the table (second column) - those with the highest Degree of Certainty scores, defined later in section 4. The average Degree of Certainty scores of the elite patterns are shown in the third column.

Classifier	Accuracy Rate	A. R. of Elite	Degree of Certainty
kNN	0.6632	0.8830	0.7676

LDA	0.6002	0.6391	0.1701
MSTC	0.6922	0.9270	0.4146
RCE	0.6255	0.7515	0.5605

Table 1. Results of experiments with Degree of Certainty elites.

These results are obtained as the averages in a series of 50 10-fold cross-validation experiments on the Glass data set from the Irvine repository of datasets (UCI Repository). In each of these experiments, the total training set is divided in ten non-overlapping subsets, of which each one in turn is considered as the test set whereas the rest 90% forms the training set to generate an instance of the algorithm which is applied then for predicting classes for the test patterns and, in this way, for estimating the algorithm's Degree of Certainty. While it is true that the algorithm's accuracy is improved on the entities with the highest Degrees of Certainty, and sometime quite well, as in the case of MSTC in which the accuracy has risen from 69% to almost 93%, the rise can be quite modest sometimes (from 60% to less than 64% for the LDA). There is no stability in both Degree of Certainty and Degree-of-Certainty based level of accuracy at different classifiers. Moreover, the Degree of Certainty levels themselves appear rather unstable regarding different sub-samples emerging in the cross-validation experiments. These point us in the direction for defining a measure of reliability of the prediction that should be less dependent on classifiers and more on the data structure.

3.2. Index of Discernibility and its Use in Classification

In the following three subsections, we introduce different versions of the Index of Discernibility (ID) of a pattern. The generic definition assumes that one operates over a training dataset P at which all patterns i are assigned with the labels $y(i)$ of classes they belong to. To capture the degree of reliability of a classifier over a pattern i , we rely on the idea that if a pattern is surrounded by patterns belonging to the same class, then the prediction of this class at the pattern should be more reliable than when the pattern is surrounded by patterns belonging to different classes. The ID value at a pattern estimates the proportion of elements belonging to the same class $y(i)$ within a neighbourhood of i . The definitions below differ because they use different concepts of the neighbourhood.

To estimate ID on a test element $i \notin P$ whose class is unknown, we use the following procedure for adoption i into P . Train the classifier under consideration over training set P , and apply it to the test element i to produce a predicted target value $\hat{y}(i)$, which is not necessarily correct. By using $\hat{y}(i)$ as the class label, the Discernibility of i , $D(i)$, can be determined. The intuition behind the adoption approach is that, because conventional classifiers are largely continuous, their action will correlate with the degree of discernibility of patterns over the hidden structure of classes. The fact that $\hat{y}(i)$ may be wrong is not that important here because we are interested in scoring the coherence between the prediction

and data rather than the prediction's correctness. What important though is that the estimated ID value depends now not only on the data structure and the definition of the neighbourhood, but also on the classifier involved.

3.3. Spherical Index of Discernibility

One way to define a measure for assessing the degree of discernibility of the class for a pattern i is to make use of a hyper-sphere around i . Assume a fixed radius around each element of the training set P , with the radius being the average distance between this element and all the other elements of its class. Note that the radius depends on the class so that elements belonging to different classes may have different radii. Once the radius r of an element i is established, we can count both the total number of elements of P and the number of elements of P belonging to the same class as i , within the radius distance r from i .

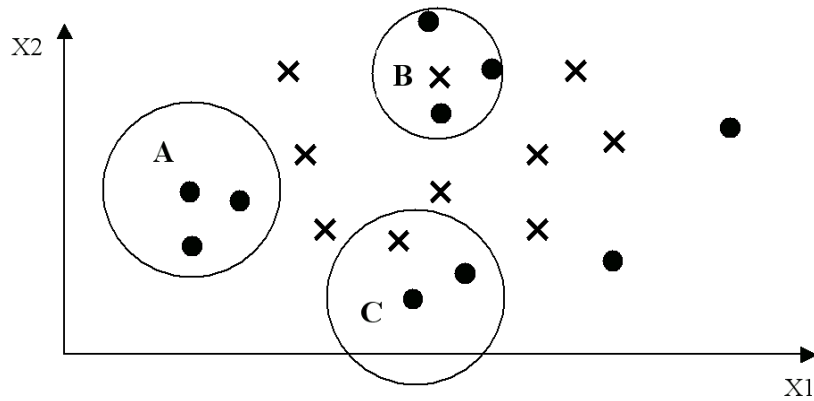


Figure 2. An illustration of the discernibility of elements in a two-class data set. In this example, the discernibility of the element in the centre of circle A is $D1 = 2 / 2 = 1$; that of the central element in the circle B is $D2 = 0 / 3 = 0$, and the discernibility of the central element in circle C is $D3 = 1 / 2 = 0.5$.

The Spherical Index of Discernibility of an element $SID(i)$ is defined as the ratio of the latter and the former, that is, proportion of i 's class elements among all the training dataset patterns in the hyper-sphere of radius r centred at i (see Figure 2). The greater the degree of discernibility, the better is the chance that the classifier's prediction is correct. This index has been used recently for conditioning the kNN classifier, which resulted in improved performances of the classifier (Voulgaris and Magoulas, 2008). To use SID on test patterns, the adoption procedure described in 3.2 is utilized.

3.4. Silhouette Width

The Silhouette Width index by Kaufman and Rousseeuw (1990) evaluates the relative closeness of an individual element to the cluster it belongs to, and we extend it to any class as well. The concept of Silhouette Width involves the difference between the within-class tightness and separation from the rest. Specifically, the Silhouette Width $SW(i)$ for entity $i \in I$ is defined as:

$$SW(i) = \frac{b(i) - a(i)}{\max(a(i), b(i))} \quad (2)$$

where $a(i)$ is the average distance between i and all other entities of the class to which i belongs, and $b(i)$ is the minimum of the average distances between i and all the entities in each of the other classes. This measure takes values in the range from -1 to 1 . If the Silhouette Width value is close to 1 , it means that the i 's class is well clustered. If the silhouette width value is close to -1 , it means that the entity is far away from its class.

In clustering, things that matter are the average Silhouette Width for each cluster and overall average Silhouette Width for the total clustering (Kaufman and Rousseeuw, 1990). We apply the measure as an index of discernibility of individual testing patterns on classes rather than clusters by using the adoption procedure of section 3.2.

3.5. Harmonic Index of Discernibility

The harmonic mean of a set of numbers x_1, x_2, \dots, x_N is defined as $N/(\sum_k 1/x_k)$. This tends to be much nearer to smaller values among the set than the arithmetic average. Thus the Harmonic Index of Discernibility HID of a pattern i is defined as

$$HID(i) = \frac{z_2(i) - z_1(i)}{z_2(i) + z_1(i)} \quad (3)$$

where $z_1(i)$ is the harmonic mean of the distances from i to its class' entities, and $z_2(i)$ the harmonic mean of the distances from i to all other entities in the set. As usual, we add a small positive number to the denominator, to avoid division by 0. If $HID(i)$ in (3) is negative, it is adjusted to be 0.

To apply HID to a test element with no class label assigned to it, the adoption procedure of section 3.2 is utilized.

4. Reliable Elites for a Classifier and a Set of Classifiers

4.1 Reject Option and Reliable Elite for a Classifier

Given a classifier and a set of known patterns (Training Set P), we aim to find the class labels and discernibility levels of a given set of unknown patterns (Testing Set) by applying the mechanism of adoption described in section 3.2 to each test pattern independently.

Given a classifier, the conventional reject option utilizes a measure of classifier reliability, such as the Degree of certainty in section 3.1. By specifying a threshold value, $T > 0$, one accepts the following rejection policy: if a pattern has Degree of certainty less than or equal to T , the classifier uses the option of rejection of classification as too uncertain. Otherwise, the classifier applies to the pattern and the label with the maximum classification score is predicted.

Sometimes the same rule applies to the classification scores output by the classifier so that the

rejection option is taken for all patterns whose maximum classification score is less than a threshold value. Somewhat more intricate rejection scheme involving the classification score is proposed by Battiti & Colla (1994) to involve two thresholds: T_1 and T_2 . If all the individual class scores of a pattern are less than T_1 , there is no decision, as usual – the pattern is rejected. If, however, the maximum score is greater than T_1 , the decision is not necessarily taken either. It depends on the difference between the maximum and next maximum individual class scores. If the difference is less than T_2 , the situation is considered uncertain and the rejection option applies. This rejection rule is referred to as T_1 & T_2 by Battiti & Colla (1994) who also recommend the value $T_2=0.2$ when classification scores range between 0 and 1, which is accepted in this paper.

Complimentary to rejecting certain patterns is pinpointing the ones that should remain, which should naturally be the ones less likely to lead to wrong classifications. One way to accomplish this is to concentrate on the most reliable, according to a measure of discernibility, test examples. Let us specify a proportion value α between 0 and 1. Given a classifier, we refer to a set of patterns as its reliability α -elite if they constitute the first αN entities in the list of test patterns sorted in the order of descent of the degree of reliability as measured by either of the three Discernibility indexes above. For example, at $\alpha=1/2$, the reliability elite will be comprised of approximately half of the test data set. An advantage of this parameter is that it allows the user to have control over the rejection process, as the number of rejected patterns is strongly correlated with α , which is not the case when using other methods.

4.2. Combining Classifiers with Reject Option

Given a set of classifiers, one can combine them in various ways. Conventionally, classifiers have been combined by using weighted voting or scoring systems (see, for reviews, Kuncheva, 2005, Polikar, 2006). In (Sansone et al., 2001) five different classifiers are used in parallel, and in (Giusti et al., 2002) two classifiers work together in a serial fashion. Battiti & Colla (1994) proposed a number of schemes, based on the voting strategy for combining classifiers with a reject option. They recommended the so-called TP1&TP2 scheme which is built on top of their T_1 & T_2 procedure described in the previous section. Specifically, they average the classification scores of the ensemble members over each individual class and then apply the T_1 & T_2 procedure to the averaged score; thus modified, the procedure is referred to as TP1&TP2 by Battiti & Colla (1994).

In our case, however, we can combine classifiers according to their reliability. Specifically, for the patterns belonging to more than one of the elites, the classification with the highest degree of reliability should prevail. This goes along the idea that is articulated by Polikar (2006) as follows: “Had we known which classifiers would work better, we would give the highest weights to those classifiers, or perhaps, use only those classifiers (Polikar 2006, p. 34).” Indeed we utilize the level of

discernibility as an ad-hoc evaluation of the classification quality.

This would lead to a simple rule for combining classifiers by combining their elites in the set-theoretic union E and using the most reliable classifier at each of the patterns in E . In other words, we combine all the different entities of the elite sets (defined by the top αN entities for each classifier). However, the overall accuracy rate on the union E can be compromised by combining “unworthy” cases for different classifiers so that the average accuracy rate on E can be lower than the accuracy rates of the individual classifiers over their elites. To mend this, we introduce two stricter rules for combining the elites: the one-third-out rule and the loners-out rule. These rules aim to filter out those elite patterns that might be riskier than the others by restricting the size of the E set to the most essential entities.

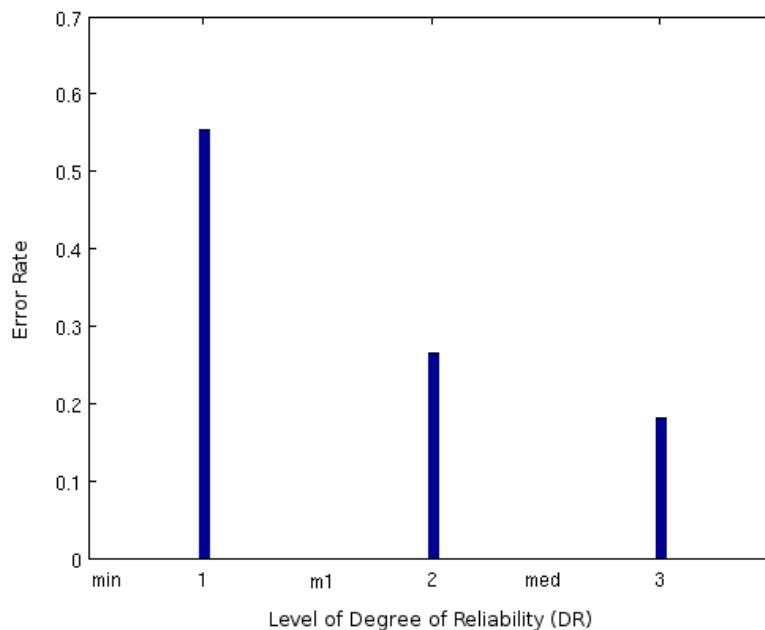


Figure 3. The average levels of misclassification for three SID categories for the *Glass* dataset.

4.2.1 The One-Third-Out Rule

This rule jettisons one third of the members from the set-theoretic union E of the individual elites. Each pattern in E is assigned with a reliability score, which is the maximum of the Discernibility scores of the classifiers whose elite the pattern belongs to. Then all elements of E are sorted according to their reliability scores, after which the bottom third of them is eliminated. The choice of this threshold is made after careful examination of the levels of misses for the members of E having their discernibility scores within different ranges. In Figure 3, one can see that indeed most of the misclassifications occur for the discernibility scores between the minimum and $m1$, which is the trisection point of the set. The graph is based on the figures obtained for the *Glass* dataset with the Spherical Index of Discernibility, yet they are typical for other sets of data considered. The main

advantage of the One-Third-Out rule is that it guarantees high accuracy rates for the amalgamation of classifiers for a variety of datasets. On the other hand, it drastically reduces the size of E.

4.2.2 The Loners-Out Rule

In this case, we focus on the number of classifiers supporting patterns in the amalgamation elite E. In our experiments, we found that in most cases a misclassification occurs on such a member of E which is voted for by only one classifier, i.e., this pattern belongs to the elite of only one individual classifier, as can be seen on Fig. 4. Like Figure 3, this graph is based on the *Glass* dataset, but it is typical for other data sets as well.

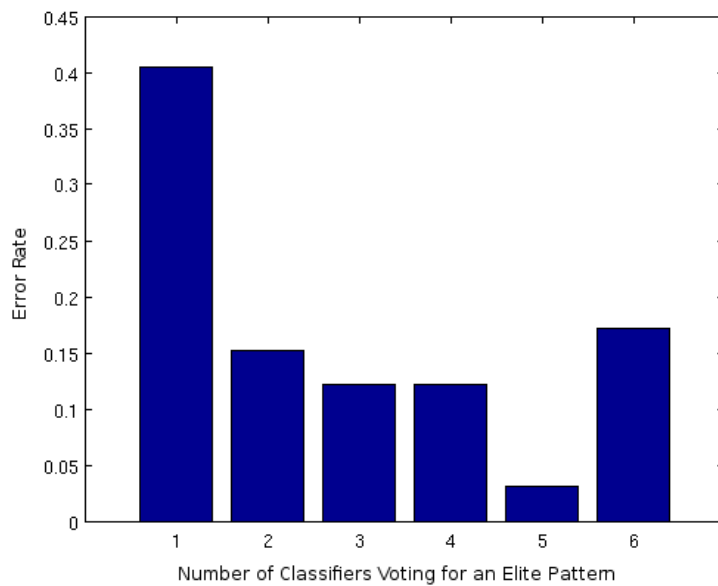


Figure 4. Average levels of misclassification for different numbers of classifiers voting for a pattern on the *Glass* dataset.

Therefore, by eliminating those members of E that come from a sole classifier, we may increase the accuracy rate at the amalgamation. Since, on average, the proportion of such patterns is rather small, the size of E is not that greatly reduced by using this rule as by the One-Third-Out rule. At the same time, since most of the patterns eliminated are bound to be misclassified anyway, the overall accuracy rate remains relatively high.

Since the two rules for removing “risky” elite members are independent from each other, they can be combined. Obviously, applying the Loners-Out rule after the One-Third-Out rule would produce larger amalgamated elites, so that we always keep this order for the combined rule. By combining the two rules, the accuracy rate in our experiments greatly improves without much decrease of the elite size of the amalgamation of classifiers.

It should be pointed out that our concern is not the conventional considerations of cost of error versus cost of rejection but rather the idea that the low accuracy rates, say those less than 93-95%, are

not acceptable at all.

5. Experimental Results

5.1 Setting Algorithms and Their Classification Scores

In the remainder, we consider the following six classifiers: k Nearest Neighbours (kNN), Linear Discriminant Analysis (LDA), Classification Decision Tree C4.5, Reduced Coulomb Energy (RCE), Gravity Model Classifier (GMC), and a Minimum Spanning Tree classifier (MSTC). We describe them in turn, paying attention to the classification score which is used further on in various algorithms:

1. kNN algorithm classifies a pattern according the plurality vote among its k nearest neighbours from the training set: the pattern is assigned with the winning class; if there is a tie, the minimum index wins. The classifier scoring function, for a class, is the number of those of the k neighbours that belong to the class. In our experiments, we accept $k=5$.
2. LDA algorithm implements Fisher's linear decision rule by deriving a separating hyperplane for each class to minimize the ratio of the "within-class" average error over the "out-of-class" average error. Its scoring function, for each class, is the value of the separating linear rule derived for the class, if it is positive, or zero, if it is negative.
3. C4.5 is a popular Quinlan's program that draws a decision tree over the training set to maximise the entropy between the tree leaf partition and class-partition. Its default parameter is the split stopping threshold ($th = 10\%$ of alien entities in a node). We take the scoring function of this decision rule to be such that the winning class has a classification score of 0.9 (1 minus 10%).
4. RCE algorithm surrounds each of the training patterns by a sphere of the maximum radius satisfying the property that no training patterns belonging to different classes belong to the sphere. (The radius is defined, thus, as the minimum distance between the current pattern and a pattern from a different class minus a small positive real, typically about 10^{-4} .) Given a test pattern, the classification score of a class is the number of training patterns from the class, whose spheres contain the test pattern.

Algorithms 1- 4 are described by Duda et al. (2001). We implemented them based on the companion manual (Stork and Yom-Tov, 2004). The following two algorithms are coded by the authors:

5. GMC algorithm closely follows the method in (Ruta and Gabrys, 2003). Given a test pattern, its squared Euclidean distances to all training instances are computed, inversed (with an added very small positive to avoid divisions by zero) and summed up within the classes. A within-class sum represents the class' gravity force. The class whose force is maximal wins and is assigned to the pattern. The forces form the classification scores.

6. MSTC method is an extension of 2NN rule utilizing a Minimum Spanning Tree (MST) representation of the training instances within each class (see Figure 1). Each edge in an MST is represented by the straight line between the corresponding patterns. The distance from a test pattern to an MST is defined as the Euclidean distance to the MST nearest edge. The classification score is the inverse of the distance to the class' MST (plus a small positive to avoid the division by zero) (Voulgaris, 2008).

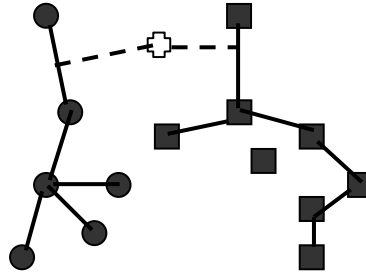


Figure 1. Illustration of MSTC: classes of circles and squares are represented by their MSTs. The dashed lines represent the distances from the white cross pattern to its nearest MST edges.

In the classifiers utilizing the between-pattern distances (those numbered by 1, 4, 5, and 6), the data are pre-normalized in such a way that, for each feature, its minimum is zero and the maximum is unity.

5.2. Accuracy Rates of the Individual Classifiers

For our experiments, five datasets from the UCI repository were used: *Iris*, *Wine*, *Heart*, *E.coli* and *Glass*. Characteristics of these datasets are shown in Table 2 below. These datasets have been chosen because of their:

- (a) diversity regarding the numbers of attributes and classes,
- (b) diversity with respect to the proportions of classes, both balanced and unbalanced, and
- (c) relatively small sizes.

The last item is important because the conventional probability-based methodologies for estimating the reliability are more problematic to implement on small datasets, so that our method can fill in the gap.

Dataset	Number of patterns	Number of attributes	Number of classes
Iris	150	4	3
Wine	178	13	3
Heart (disease)	270	13	2

E.coli	336	8	8
Glass (identification)	214	10	6

Table 2. Characteristics of the datasets used in our experiments.

All of the experiments were conducted as 50 rounds of the 10-fold cross-validation testing described in section 2.1. Therefore there were 500 classification testing exercises for each of the classifiers, rendering the results rather stable and, thus, reliable.

Table 3 presents the average accuracy rates of the individual classifiers on the whole datasets, as well as the standard deviations of the rates.

Dataset Classifier	Iris		Wine		Heart		E.coli		Glass	
	Acc.	St. Dev.	Acc.	St. Dev.	Acc.	St. Dev.	Acc.	St. Dev.	Acc.	St. Dev.
kNN	0.957	0.004	0.953	0.008	0.801	0.007	0.865	0.007	0.669	0.013
LDA	0.839	0.009	0.984	0.006	0.841	0.006	0.855	0.005	0.606	0.013
MSTC	0.953	0.003	0.955	0.004	0.765	0.014	0.794	0.008	0.696	0.015
GMC	0.955	0.003	0.979	0.004	0.800	0.007	0.791	0.007	0.682	0.014
C4.5	0.944	0.011	0.930	0.011	0.702	0.017	0.785	0.015	0.572	0.026
RCE	0.881	0.021	0.970	0.007	0.739	0.015	0.706	0.012	0.600	0.027

Table 3. Accuracy rates and their standard deviations, for the classifiers under consideration.

One can see that, with respect to the classifiers, the datasets fall in three categories:

- (a) relatively high accuracy rate of 90-97%, on Iris and Wine;
- (b) medium accuracy rate of about 80%, on Heart and E.coli, and
- (c) relatively low accuracy rate of 60-70%, on Glass.

Also, one should point out how low standard deviations of the accuracy rates are for all the classifiers. We can also see that performances of different classifiers peak at different datasets: kNN is the best on Iris and E.coli, LDA the best on Wine and Heart (and the worst on Iris), and MSTC is the best on Glass. The other three algorithms trail behind regarding the overall performances, yet they should not be discarded altogether – each may perform well on elites as reflected in further tables. Since the standard deviations are very low in all the cases, we do not report them further on, for the sake of space.

5.3. Choosing the Discernibility Measure

Classifier	D. Meas, Elite	Iris	Wine	Heart	E.coli	Glass	
kNN	SID	50%	0.9997	0.9997	0.9139	0.9658	0.6941
		33%	1.0000	1.0000	0.9476	0.9788	0.7328
	SW	50%	0.9997	0.9965	0.9239	0.9366	0.6898

		33%	1.0000	1.0000	0.9511	0.9788	0.7328
	HID	50%	1.0000	0.9990	0.9141	0.9268	0.8596
		33%	1.0000	1.0000	0.9478	0.9305	0.9230
LDA	SID	50%	0.9990	1.0000	0.9146	0.9705	0.6929
		33%	1.0000	1.0000	0.9476	0.9790	0.7219
	SW	50%	0.9995	1.0000	0.9241	0.9319	0.6453
		33%	1.0000	1.0000	0.9511	0.9502	0.6970
	HID	50%	1.0000	1.0000	0.9174	0.9267	0.8124
		33%	1.0000	1.0000	0.9482	0.9302	0.8887
C4.5	SID	50%	0.9997	0.9997	0.9175	0.9605	0.6913
		33%	1.0000	1.0000	0.9416	0.9779	0.7289
	SW	50%	0.9997	0.9972	0.9186	0.9545	0.6413
		33%	1.0000	1.0000	0.9509	0.9579	0.7041
	HID	50%	1.0000	0.9990	0.9062	0.9470	0.8284
		33%	1.0000	1.0000	0.9524	0.9449	0.9157
RCE	SID	50%	0.9997	1.0000	0.9071	0.9581	0.6401
		33%	1.0000	1.0000	0.9453	0.9762	0.7002
	SW	50%	0.9995	0.9967	0.9199	0.9548	0.6669
		33%	1.0000	1.0000	0.9491	0.9745	0.7259
	HID	50%	1.0000	0.9988	0.9116	0.9492	0.8388
		33%	1.0000	1.0000	0.9476	0.9631	0.9300
GMC	SID	50%	0.9997	0.9997	0.9123	0.9621	0.7055
		33%	1.0000	1.0000	0.9476	0.9776	0.7239
	SW	50%	0.9997	0.9967	0.9237	0.9382	0.7080
		33%	1.0000	1.0000	0.9511	0.9501	0.7554
	HID	50%	1.0000	0.9990	0.9140	0.9266	0.8740
		33%	1.0000	1.0000	0.9478	0.9286	0.9408
MSTC	SID	50%	0.9997	0.9997	0.9144	0.9617	0.7440
		33%	1.0000	1.0000	0.9560	0.9777	0.7739
	SW	50%	0.9997	0.9967	0.9224	0.9310	0.7026
		33%	1.0000	1.0000	0.9533	0.9449	0.7540
	HID	50%	1.0000	0.9990	0.9156	0.9254	0.8463
		33%	1.0000	1.0000	0.9520	0.9288	0.9074

Table 4. Accuracy rates of various classifiers at different elite levels and indexes of Discernibility.

To choose among the three different discernibility measures defined, we analysed performances of all six classifiers at two levels of elites: $\alpha=1/2$, and $\alpha=1/3$. The former using only those patterns whose degree of discernibility is better than the median discernibility, and the latter comprises the best third of discernible patterns. These two levels are maintained at each of the three discernibility indexes defined above, SID, SW and HID. Table 4 presents average accuracy rates of the individual classifiers on the five datasets at each of the six combinations of the elite level and discernibility index.

The results from Tables 4 show:

- (1) All three discernibility indexes lead to drastically raising accuracy rates for all the classifiers, reaching 100% accuracy for Iris and Wine datasets and about 95-97% accuracy on E.coli dataset on the 50%-elites. The only diehard is Glass dataset that does not much change the accuracies at the 50%-elites over SID and SW indexes. Still, HID index leads to a much improved, 85%, accuracy over 50%-reliability elites, and more than 90% accuracy over the 33%-reliability elites.
- (2) There is no overwhelming winner among the three Discernibility indexes, though each of the indexes shows consistent results over all the classifiers. Specifically, SID always wins on Wine and E.coli datasets, HID always wins on Iris and Glass datasets, and SW is the winner on Heart dataset.
- (3) The RCE and GMC classifiers, unimpressive over the total data set, become most effective over the elites.

One can see that the best performance overall is achieved at the HID index, which is used in all the further experiments.

The next series of experiments has been performed on the two data sets, Glass and Heart, which are more complex than the others as is easily seen from Tables 3 and 4. We consider here only two, best performing classifiers, the kNN and MSTC. The elites were found using the proposed method DD (HID variation), the Degree of certainty (DC) approach and the T1&T2 method of Battiti & Colla (1994) described in section 4.1. Different rejection rates were tried so as to examine how the accuracy rate varies for the three methods. The results can be viewed in Figures 5 and 6.

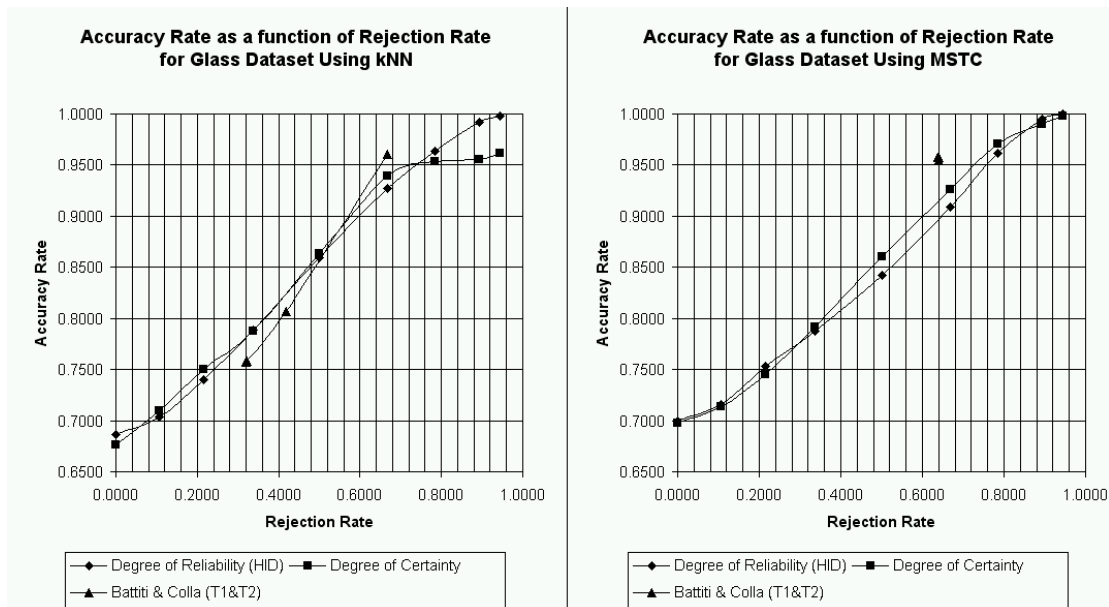


Figure 5. Comparison among the proposed method, the Degree of certainty method and a method proposed by Battiti & Colla (T1&T2), for the *Glass* dataset, using two classifiers individually.

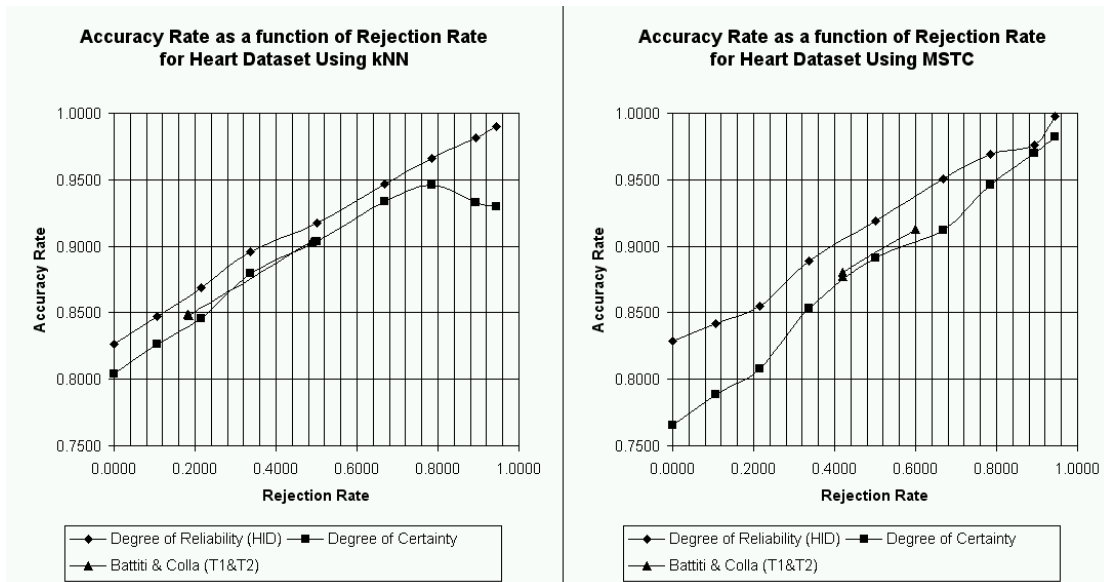


Figure 6. Comparison of the proposed method, the Degree of Certainty method and a method proposed by Battiti & Colla (T1&T2), for the *Heart* dataset, using two classifiers individually.

From these graphs one can see that in many cases – but not always! - the proposed method performs better than the others. Yet, this method, as well as the one based on Degree of certainty, has a broader range of rejection rates compared to the T1&T2, making them more versatile.

5.4. Accuracy Rates for Combined Classifiers

The results for the amalgamation of reliability elites using rules 1/3-Out, Loners-Out and their combination are shown in Tables 5 and 6. The former relates to the accuracy rates whereas the latter to the amalgamation sizes.

These lead to the following conclusions. Rather expectedly, 1/3-Out rule consistently outperforms the Loners-Out rule; however, this is by just a small margin of the order of 1% or less - with the price of drastically reducing the size of the elite. Overall, the amalgamation does not boost performances of the algorithms that much. However, we can see that HID 33%-elites consistently lead to the accuracy rates of 90% and more. The combined rule raises the accuracy on the Glass dataset – the most difficult for predictions – to more than 94%. The price, in terms of reject-option applied to the non-elite, is rather high indeed: 60%, 70%, and 75% of all cases for Loners-Out, 1/3-Out and Combined rules, respectively. But this may be worth doing in the situations at which the reliability of classification is a must.

Also, Table 6 shows that the SID 33%-elite Loners-Out amalgamation leads to somewhat better coverage of the data – about 45%, rather than 33%, of the dataset are there.

Rule	DI Elite	Iris	Wine	Heart	E.coli	Glass
------	----------	------	------	-------	--------	-------

Loners Out	SID	50%	0.9991	0.9993	0.9054	0.9557	0.6630
		33%	1.0000	1.0000	0.9422	0.9753	0.6907
	SW	50%	0.9991	0.9994	0.9142	0.9332	0.6107
		33%	1.0000	1.0000	0.9476	0.9450	0.6843
	HID	50%	0.9998	0.9991	0.9120	0.9223	0.7714
		33%	1.0000	1.0000	0.9470	0.9317	0.9030
One Third Out	SID	50%	1.0000	1.0000	0.9396	0.9763	0.6754
		33%	1.0000	1.0000	0.9710	0.9823	0.7253
	SW	50%	1.0000	0.9997	0.9463	0.9420	0.6277
		33%	1.0000	1.0000	0.9751	0.9630	0.7123
	HID	50%	1.0000	1.0000	0.9407	0.9305	0.8645
		33%	1.0000	1.0000	0.9588	0.9371	0.9164
Combination of Both Rules	SID	50%	1.0000	1.0000	0.9425	0.9764	0.6824
		33%	1.0000	1.0000	0.9736	0.9817	0.7484
	SW	50%	1.0000	0.9997	0.9423	0.9478	0.6457
		33%	1.0000	1.0000	0.9760	0.9652	0.7191
	HID	50%	1.0000	1.0000	0.9405	0.9302	0.8677
		33%	1.0000	1.0000	0.9635	0.9417	0.9444

Table 5. Accuracy rates at the different methods of amalgamation of reliability elites.

Furthermore, we also performed another comparison with the Battiti & Colla method TP1&TP2 (1994), which is applied to the whole set of six classifiers. The results of this comparison, along with the results of a comparison with the DC measure can be viewed in Figure 7. Note that due to the unpredictability of the exact reject rates (which we determine indirectly via one parameter, in both methods), it is infeasible to have an exact correspondence of the data points of the two lines on the X axis.

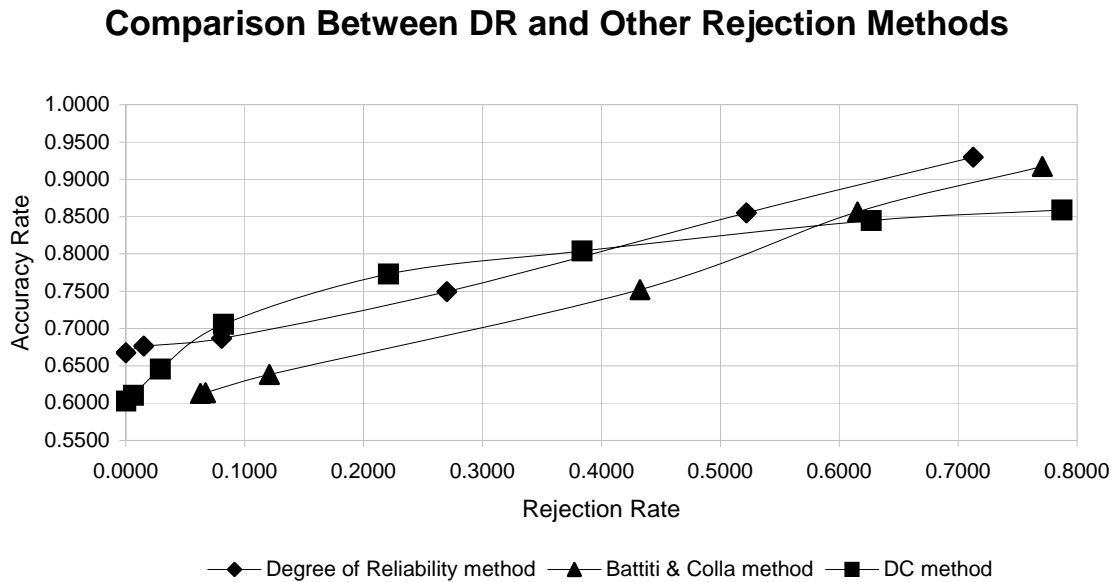


Figure 7. Comparison between the proposed method with the Degree of certainty (DC) method and a majority-based method from the literature (TP1&TP2).

From Figure 7 one can see that the proposed method appears to yield better results than the TP1&TP2. Especially important for the goal of reducing the need in manual testing of devices with low accuracy rates is the rejection rate at which the accuracy exceeds 90%: about 65 % for DR and 73% for TP1&TP2. The DC method never achieves the level of 90% accuracy, probably because of its flaws mentioned in section 3.1. A possible explanation for the less impressive performance of TP1&TP2 is that the differences between different classifiers make it difficult to efficiently average their classification scores. Note that the original research by Battiti & Colla (1994) dealt with only classifiers generated by changing parameters of a generic classifier.

6. Conclusion

We propose evaluation of the reliability of a classifier on a test pattern by using an independent Index of discernibility. This measure evaluates how well the pattern is positioned against its class members versus to its position against other classes. Three different discernibility measures are considered: (a) Spherical Index of discernibility (SID), the class' proportion in the pattern's neighbourhood, (b) Silhouette Width (SW), the relative difference between the average pattern's distances to its own class and the nearest other class, and (c) Harmonic Index of discernibility (HID), based on the harmonic

average distances from the pattern to its class members and the rest. We compare them with more conventional procedures and show that using the discernibility index leads to comparable results – sometimes slightly better, sometimes slightly worse – with those conventional ones, at small and moderate rejection rates.

Yet the focus of our study is in using our indexes as devices for pre-selection of instances to warrant the error rates within 5% or less. Such a pre-selection is required at domains related to testing very complex technical devices or diagnosing human diseases, because the low classification score or a lesser accuracy rate at a non-rejected pattern still may lead to the need in manual testing. It appears then that HID is the only rule to achieve higher accuracy rates on all of the data sets under consideration. Amalgamating the reliability elites of different classifiers can further boost the pattern discernibility and make it classifier-independent.

Rule	D. Meas., Elite	Iris	Wine	Heart	E.coli	Glass	
Loners Out	SID	50%	55.6	50.3	57.0	54.8	68.0
		33%	43.4	39.5	35.6	36.8	46.0
	SW	50%	54.1	50.1	56.3	57.8	70.7
		33%	33.3	37.1	35.0	37.4	47.6
	HID	50%	53.8	50.2	54.7	55.8	66.5
		33%	33.4	37.1	34.1	37.0	39.6
One Third Out	SID	50%	43.8	36.0	37.0	35.3	43.2
		33%	20.0	24.7	22.5	23.5	29.3
	SW	50%	34.0	31.7	36.6	37.6	45.3
		33%	20.0	24.7	22.5	24.2	29.9
	HID	50%	33.8	31.7	35.4	36.0	43.0
		33%	21.0	24.7	25.8	27.0	30.1
Combination of Both Rules	SID	50%	43.9	35.9	36.9	35.3	43.3
		33%	19.9	24.7	22.6	23.6	28.5
	SW	50%	34.1	31.7	36.5	37.7	45.6
		33%	19.9	24.7	22.5	24.2	30.0
	HID	50%	33.9	31.6	35.5	36.0	42.5
		33%	20.0	24.7	22.3	24.1	24.4

Table 6. Elite sizes, per cent, at different methods of amalgamation of reliability elites.

Our approach additionally has the following advantages. First, in contrast to conventional techniques for estimating reliability, the measure is applicable on small data sets with many classes to diagnose. Second, it can work on data sets with a complex class structure at which all classifiers under consideration would show low accuracy rates. The Glass dataset provides an example manifesting both of these.

We consider this work as of a proof-of-the-concept stage rather than of a final stage. Future work should include examination of how the reject option range can be reduced by combining the pattern discernibility measures with those of classifier's accuracy and certainty. Another possibility may emerge by developing better methods for combining classifiers; specifically, there may be some potential in using two-level ensembles of classifiers. Development of additional rules for filtering out "unreliable" elite members can be another avenue for future research.

References

1. Arlandis, J., Perez-Cortes, J.C., Cano, J., 2002. Rejection strategies and confidence measures for a k-NN classifier in an OCR task. In: Proceedings of 16th International Conference on Pattern Recognition ICPR-2002, Vol. 1, Québec (Canada), 576 – 579.
2. Aydin, T., Guvenir, H. A., 2006. Modeling interestingness of streaming classification rules as a classification problem. Lecture Notes in Computer Science, Springer, Berlin / Heidelberg, ISSN 1611-349 (Online), 3949 (last accessed: October 2009).
3. Baram, Y., 1998. Partial Classification: The benefit of deferred decision. IEEE Transactions on Pattern Analysis and Machine Intelligence, 20 (8), 769-776.
4. Battiti, R. and Colla, A.M., 1994. Democracy in neural networks: voting schemes for classification. Neural Networks, 7(4), 691-707.
5. Cordella, L. P., De Stefano, C., Tortorella F., Vento, M., 1995. A method for improving classification reliability of multilayer perceptrons. IEEE Transactions on Neural Networks, 6 (5), 1140-1147.
6. Denoeux, T., 1995. A k-nearest neighbor classification rule based on Dempster-Shafer. IEEE Transactions on Theory, Systems, Man and Cybernetics, 25 (5), 804 – 813.
7. Duda, R. O., Hart, P. E., Stork, D. G., 2001. Pattern Classification (2nd ed.), John Wiley and Sons, University of Michigan.
8. Fumera, G., Roli, F., Giacinto G., 2000. Reject option with multiple thresholds. Pattern Recognition, 33 (12), 2099-2101.
9. Giusti, N., Masulli, F., Sperduti, A., 2002. Theoretical and experimental analysis of a two-stage system for classification. IEEE Transactions on Pattern Analysis and Machine Intelligence, 24 (7), 893-904.
10. Kaufman, L., Rousseeuw, P. J. , 1990. Finding groups in data. Wiley Interscience Publications, New York, 83-85.
11. Kuncheva, L.I., 2005. Combining Pattern Classifiers, NY, Wiley Interscience.
12. Polikar, R., 2006. Ensemble based systems in decision making, IEEE Circuits and Systems Magazine, Third Quarter, 21-44.

13. Ruta, D., Gabrys, B., 2003. Physical field models for pattern classification. *Soft Computing Journal*, 8 (2), 126-141.
14. Sansone, C., Tortorella, F., Vento, M., 2001. A classification reliability driven reject rule for multi-expert systems. *International Journal of Pattern Recognition and Artificial Intelligence*, 15 (6), 1-19.
15. Santos-Pereira, C. M., Pires, A. M., 2005. On optimal reject rules and ROC curves. *Pattern Recognition Letters*, 26 (7), 943-952.
16. Stork, D. G., Yom-Tov, E., 2004. *Computer manual in Matlab to accompany pattern classification* 2nd ed., Wiley-Interscience.
17. Thien, M. Ha, 1996. An experimental study of the optimal class-selective rejection rule. Available online at: <http://citeseer.ist.psu.edu/103812.html> (last accessed: December 2007).
18. Thien, M. Ha, 1996. Application of the optical class-based rejection rule to the detection of abnormalities in OCR databases. Available online at: <http://citeseer.ist.psu.edu/135692.html> (last accessed: December 2007).
19. Tsymbal, A., Puuronen, S., 2000. Bagging and boosting with dynamic integration of classifiers. *Principles of Data Mining and Knowledge Discovery*, 116-125.
20. UCI repository: <http://archive.ics.uci.edu/ml/datasets.html> (last accessed: February 2008).
21. Voulgaris, Z., 2008. An extension of the nearest neighbour classifier using minimum spanning tree, Unpublished manuscript.
22. Voulgaris, Z., Magoulas, G., 2008. Extensions of the k nearest neighbour methods for classification problems. In: *Proceedings of the 26th IASTED International Conference on Artificial Intelligence and Applications*, 23-28.