DIMACS Technical Report 2003-23
September 2003

Feature Selection and Training Set Sampling for Ensemble Learning on Heterogeneous Data

by

Iryna Skrypnyk[1]
Dept. of Computer Science
and Information Systems
University of Jyväskylä
P.O. Box 35
40351 Jyväskylä, Finland
E-mail: iryna@cs.jyu.fi

Tin Kam Ho[2]
Computing Sciences Research Center
Bell Laboratories
Lucent Technologies
700 Mountain Ave., 2C-425
Murray Hill, New Jersey 07974
E-mail: tkh@research.bell-labs.com

[1]This research has been conducted during the visit of Iryna Skrypnyk to DIMACS and Bell Laboratories in March - April 2003. The visit has been hosted by Dr. Tin Kam Ho. A sponsorship of DIMACS is gratefully acknowledged.
[2]Permanent Member

ABSTRACT

Heterogeneity in the data is exhibited in the variability of feature importance across the set of instances and in the difference between the decision rules that distinguish groups of classes from the rules distinguishing classes within each group. Such situations occur frequently in the contemporary large databases. It is a critical challenge for most data mining techniques that requires a specific treatment. In this work we formalized the notion of data heterogeneity, considered its various types, and studied a feature space heterogeneity case with ensemble techniques. Ensembles, or multiple classifier systems, are often capable of producing more accurate classifications than a single classifier. However, the mechanisms of accuracy improvement are different in different ensemble techniques - bootstrapping, error correcting coding, etc. For instance, training set sampling techniques are superior in reducing error bias and variance while feature set sampling (feature subspace projection) techniques are often good in increasing diversity of component classifiers' predictions. Applying ensembles to the heterogeneous data turns those mechanisms to work on promoting the coverage of local homogeneous regions by different component classifiers. The reason for it is that those local homogeneous regions are very hard to elicit without prior domain knowledge, and known techniques to find homogeneous regions are designed for specific cases of heterogeneity, for example, contextual features. We experimented with two representative ensemble techniques, Bagging for training set sampling and the Random Subspace Method for feature subspace projection. Their contribution to accuracy growth has been studied on several synthetic data sets modeling different cases of feature space heterogeneity. The potential benefits of combining feature subspace projection and sampling are considered.

1 INTRODUCTION

Nowadays data mining techniques are applied in knowledge discovery processes to analyze huge amounts of heterogeneous data accumulated in electronic data repositories. Sometimes the data are collected over multiple periods of time and integrated from different sources, where the same-purpose measurements have been done under different schemes, and consequently, described by different features. Often data is not gathered by a purposefully designed experimental procedure, but rather obtained by some process whose goal was not necessarily knowledge discovery. Such data is typically of high dimension, often including redundant information, and prior knowledge about the problem domain is fairly limited. For example, in large enterprises, data may be scattered in a number of departments, where the goal, depth, and standard of data collection may vary. When data analysis requires data from different departments to be integrated in electronic form the collection could become complex, multidimensional, spatially clustered and redundant.

The fact that data is heterogeneous may not necessarily be known in advance. The problems with such data significantly embarrass learning with traditional methods, even when some exploratory analysis has been performed. Heterogeneity is the name used here for the situation when some features are unstable in their importance across the set of instances (feature space heterogeneity) (Apte *et al.*, 1997; Lazarevic *et al.*, 2000).

A particular case of feature space heterogeneity is class heterogeneity when the decision rules that distinguish one group of classes from another are very different from those that separate the classes within each group. In general, in different regions of the feature space the relevant features separating the classes may be different, i.e. feature subsets on which the decision rules are constructed may be different. Class heterogeneity, however, can be in many cases successfully solved by class-decomposition ensemble techniques, for example, when the decision rules to distinguish each pair of classes are constructed specifically for the corresponding component classifier. Feature space heterogeneity is critical for majority of learning models since the regions of feature relevance, or factors responsible for variability of feature relevance are unknown. Often mutual relationships between features may be the reason of relevance instability. This work is focused on studying a general case of feature space heterogeneity.

As an example of feature space heterogeneity we consider a data set containing all legal 8-ply positions in the Connect-4 6x7 board game (Allen, 1990) in which neither player has won yet and the next move is not forced. From these positions it is necessary to predict either win/loss for the first player, or draw. In this data set each cell is represented by a feature taking three possible values: x - the first player has taken, o - the second player has taken, and b - blank. For each case (instance) we do not need to consider all features (values of each cell) in order to make a prediction.

The features important for prediction may change from case to case. Obviously, the subset of features taking x or o values in the particular instance is always relevant for this instance. This subset specifies an additional subset of features taking b value to be considered. So, the relevance of the particular subset of b-valued features depends on the contextual features whose values are either x or o. This example illustrates an important issue of feature space heterogeneity: relevance of different feature subsets may rely on the values of some contextual feature, or a set of contextual features.

Sometimes contextual features are not available. Then we may consider that the relevance of a feature subset depends on the location of instances in the feature space. Relation between contextual features and the rest of features (primary features), so as relation between features and class variable, can be modeled as correlation between variables.

In the case of heterogeneous data some dependencies may not hold true for the whole data set. The problem can be solved by identifying local spatial regions where dependencies exist or by developing technique robust to heterogeneity in the data. Methods that follow these two approaches are considered in Section 2.

In this work we examine ensemble learning on data with globally and locally relevant features as a potentially useful approach to treat local dependencies. We consider the mechanisms of accuracy improvement used in different ensemble techniques on synthetic heterogeneous data in order to understand advantages and disadvantages of ensemble learning when heterogeneity is present. Two ensemble generation techniques, the Random Subspace Method (RSM) and Bagging, represent manipulation of the input features and training instances for component classifiers generation. These techniques are discussed in Section 3. The experimental study on synthetic data is described in Section 4.

We suggest that combination of different ensemble generation techniques might be useful in the case of data heterogeneity, in particular, feature space heterogeneity. Discussion about combining ensemble techniques is presented in Section 5. We summarize with conclusions in Section 6.

2 LOCAL FEATURE RELEVANCE AND INSTABILITY

During the last decade the problem of heterogeneity in data have been addressed in several machine learning studies under different names, such as attribute instability (Lazarevic *et al.*, 2000), relevance in context (Domingos, 1997), feature space and class heterogeneity (Apte *et al.*, 1997), local feature relevance (Howe & Cardie, 1997; Friedman, 1994).

The fact that feature relevance may vary across the set of instances was elaborated in early works on feature weighting for lazy learning algorithms, usually for the k-Nearest Neighbor classifier. Two representative local weighting schemes for k-Nearest Neighbor (Hastie & Tibshirani, 1994; Friedman, 1994) are instance-specific. In (Hastie & Tibshirani, 1994) a separate distance metric for each target instance is computed through an iterative process. In (Friedman, 1994) the most relevant feature is scaled at each step such that a fixed fraction of the given training instances fall outside of a predetermined range around the target instance. The training instances outside of that range are then discarded, the new most relevant feature is determined, and the process is repeated until only k training instances remain. The local relevance of each feature is measured by the estimated reduction in classification error. Both works report favorable results on their local approaches comparatively to global ones, but both of them are computationally expensive.

Howe and Cardie in (Howe & Cardie, 1997) proposed a coarsely local feature weighting scheme, class distribution weighting, where feature weights are allowed to vary, and remain identical for certain clusters of instances. This method is a precursor for several class-decomposition ensemble and feature selection techniques like (Oza & Tumer, 1999; Hall, 1999), which assume that there are features that are useful at distinguishing whether an instance is of one particular class, but are not useful in distinguishing among the remaining classes. The method presented in (Howe & Cardie, 1997) is based on the assumption that although classes are not always homogeneous, it is plausible that for many domains features informative for a particular class are the same for most or all instances belonging to that class.

Domingos in (Domingos, 1997) extended the concept of local feature relevance. He motivated that some features may be highly relevant in certain regions of the instance set and irrelevant everywhere else by their sensitivity to a context, i.e. to the values of the other features. The Relevance-in-Context method proposed is distance-based and instance-specific that makes it computationally expensive. In this work a *feature difference* measure is also considered to evaluate the context dependency effect exhibited by Relevance-in-Context in real data sets.

Alternatively, the idea of contextual features has been explored in (Turney, 1996; Turney, 1993). Turney distinguishes three different types of features: *primary*, *contextual*, and *irrelevant* features. By his definition, primary features are useful for classification when considered in isolation, without regard for the other features. Contextual features are not useful in isolation, but can be useful when combined with other features. Irrelevant features are not useful, either when considered alone or when combined with other features. In those works different strategies for exploiting context have been studied, assuming that contextual features are available for learning.

Harries and Horn in (Harries & Horn, 1996) recognize hidden changes in context for concept learning and propose a batch learning approach. They identify *environmental* features, which reflect hidden context. The examples of environmental features are time or spatial location. The SPLICE method proposed is a meta-level algorithm that uses a learning algorithm capable of performing context-sensitive feature selection, like Relevance-in-Context, or decision tree. Such learning algorithms select features at each node, rule, or clause in the context of locally relevant prior selections. Partitions made by them over some environmental feature selected (for example, time feature) identify possible changes in context. Then contextual clustering is performed over the intervals according to apparent similarity of context and local context-specific concepts are learned. SPLICE does not extract the contextual features from heterogeneous data, assuming that context may be contiguous over some ordinal features (environmental).

In (Apte *et al.*, 1997) a method to search for evidence of heterogeneity and a method to decompose the problem into constituent subproblems are proposed. As a method to detect heterogeneity they have proposed a measure that reflects the degree of dissimilarity between the decision boundaries for a couple of subproblems. This measure is based on a feature merit measure (Contextual Merit measure, ReliefF, Information Gain measure, etc.) If any heterogeneity is found, a tree strategy is proposed to identify homogeneous regions of feature space.

The hypothesis used in (Cardie & Howe, 1997; Hall, 1999; Oza & Tumer, 1999) that one may coarsely group features informative to predict instances of particular classes has been reconsidered in (Apte *et al.*, 1997). In this work has been argued that differences in class probability distribution may give an indirect indication for decomposition and propose to compare the profiles of importance of the features for the subproblems.

Analyzing the existing literature on local relevance and heterogeneity we observe two ways to categorize the strategies: (1) data structure oriented strategy versus class separability oriented strategy, and (2) identification of contextual features versus increasing sensitivity to context changes.

As we argued in Section 1, correlation between feature and class variable may be used to model heterogeneity. In the case of continuous variables, this can be measured by Pearson's or Spearman's correlation coefficient or a correlation ratio. However, for classification problem we usually have a mixture of categorical (nominal) and continuous (numeric) variables. Then correlation between two features or a feature and class variable can be evaluated through other measures. For estimating a feature's relevance to describe a particular class various feature merit measures have been developed, for example, Information Gain (Quinlan, 1986), Gini Index (Brieman *et al.*, 1984), Relief (Kira & Rendell, 1992), Contextual Merit (Apte *et al.*, 1997). In order to examine a relationship between two features a modification of those measures may be used. One of them, the Symmetrical Uncertainty measure (SU), has been considered in (Hall, 1999).

Contextual features as a particular case of feature dependencies have been considered in (Robnik-Sikonja & Kononenko, 1999; Robnik-Sikonja & Kononenko, 1996). Robnik-Sikonja and Kononenko in (Robnik-Sikonja & Kononenko, 1996) argue that Relief and its extension ReliefF are both capable of estimating correctly the quality of features in classification problems with strong feature dependencies. By exploiting the local information provided by different contexts they provide a global view and recognize contextual features.

In the original Relief (Kira & Rendell, 1992) the quality of features is estimated according to how well their values distinguish between the instances that are near to each other, evaluating two nearest neighbors of the target instance - from the same and different class. When calculating a merit measure for a feature Relief, similar to Contextual Merit, takes into account correlations between features contrary to the myopic feature merit measures that assume feature independence, such as Information Gain and Gini Index.

Late works dealing with heterogeneous data and the local relevance phenomenon employ ensemble learning. Class-decomposition ensemble techniques considered in the next section originally have been designed to solve multi-class classification problems. A new use of class-decomposition has been proposed in (Oza & Tumer, 1999). The Input Decimation method described in this work generates one component classifier for each class over the features correlated with this class.

In (Lazarevic *et al.*, 2000) the Adaptive Boosting technique for heterogeneous spatial databases with instable features changing their relevance across the instance set has been developed. In each boosting round the local information for the drawn sample is maximized by feature selection, extraction or weighting. At the same time instead of sampling the training instances spatial data blocks are drawn.

From this overview we may see that the question of the context of a feature is related to the question of its relevance. In (Aha & Goldstone, 1990) it is pointed out that feature relevance is a context-specific property, i.e. depends on the nearest instances or values of other features. Context in the works considered above has slightly different meaning, thus we will refer to contextual features as to *control* features; it reflects that relevance of the other features depends on the values of control features.

We distinguish *relevant*, *irrelevant*, *redundant* and *interacting* features similar to (Hall, 1999). All these kinds of features influence learning differently.

Features that provide information about the class are called *relevant* features. Features that do not provide any information about the class are *irrelevant*. *Interacting* features are those whose values are dependent both on the values of the other features and on the class. One of two interactive features may be discarded if it does not imply any loss of the information about the class. *Redundant* features are those whose values are dependent on the values of other features regardless the class, and as such provide no further information about the class. The definition of redundant features commonly used in machine learning literature[1] is similar to the definition of interacting features used here.

Below we summarized the most common situations of relevant, irrelevant, redundant, and interacting features combinations in the classification problems, including heterogeneity case.

1. Features are relevant and independent.

2. Some of the features are globally irrelevant. All features are independent.

3. Features are relevant; some of them interact.

4. Some features are redundant, depending only on the values of relevant features.

5. Some features are relevant locally. All features are independent.

6. There are control and primary features. Primary features are relevant locally.

Different situations can be modeled as a combination of the cases above. Now we extend local feature relevance cases 5 and 6 considering the class distribution.

5.1 Features are independent; some of them are relevant locally for a particular class.

[1] Often feature considered as redundant is a relevant feature whose values are dependent on the values of another feature and can be expressed via that other feature values without loss of information provided about the class.

5.2 Some features are relevant locally in the instance set representing instances from different classes.

6.1 There are control and primary features. Different subsets of primary features are relevant for different classes. All features are independent.

6.2 There are control and primary features. Primary features are relevant locally in the instance set representing instances from different classes.

In order to study the behavior of feature subspace projection and sampling ensemble techniques we have chosen three cases from the considered above: 1-3, 5.2 and 6.2. Cases 5.1 and 6.1 are simpler than 5.2 and 6.2, respectively, because groups of features informative to predict instances of a particular class specify the local relevance - the assumption used in (Howe & Cardie, 1997; Oza & Tumer, 1999; Hall, 1999).

3 ENSEMBLE CLASSIFICATION ON HETEROGENEOUS DATA

Nowadays multiple classifier systems (ensembles) became firmly established as a practical and effective solution for difficult prediction tasks. The rationale for combining several learning algorithms, i.e. classifiers, for the classification task is the following. Different component classifiers make errors in different situations due to their different design. When they are combined, they can produce more accurate prediction comparatively to a single classifier trained to reach the highest accuracy for all cases in the data set at once. There are different theories explaining ensemble efficiency (Dietterich, 1997; Kleinberg, 1990; Schapire *et al.*, 1998).

Usually ensembles are used to improve the prediction accuracy by mechanisms motivated by learning theory rather than by particular data characteristics. In (Dietterich, 1997; Maclin & Opitz, 1997; Opitz & Maclin, 1999; Tumer & Ghosh, 1996) it is explained why the component classifiers having accuracy of more than 50% should make uncorrelated errors to be effective in the ensemble. This important property of ensemble is called diversity of predictions.

Kleinberg's theory of stochastic discrimination (Kleinberg, 1990) argues that a set of classifiers in the ensemble should cover all data points uniformly, and with equal chance of success. Then discrimination between classes is achieved when the number of component classifiers is large and all classifiers satisfy a requirement of minimal difference in inclusion of instances from different classes.

These approaches emphasize the properties of component classifiers that will be integrated in the ensemble. To generate those component classifiers some sophisticated search strategy is usually applied. The simplest approach to generate component classifiers is to produce alternative perturbations of the training process created by random sampling.

Research progress in the direction of ensemble learning resulted in a number of techniques. Different techniques can be categorized as using homogenous classifiers, using a single learning model, and using different learning models for generating component classifiers. Some ensemble techniques are developed for a set of particular homogenous classifiers, as for example, neural networks. Three major categories of ensemble techniques can be recognized by the principle of ensemble generation (Dietterich, 1997): (1) *sampling* the training data to construct the component classifiers, (2) manipulation by the features (attributes) describing the training instances, called *feature subspace projection*, and (3) encoding of the output target (predictive feature or class variable) for different classes, called *class-decomposition*.

The situation with the voluminous data stored in the contemporary data repositories can be much more complicated than in the examples provided in Section 1. Ensembles techniques and their combination may be advantageous in the case of heterogeneous data since the learning process is altered in different ways.

For the heterogeneity problem ensemble components are supposed to cover local regions of the feature space with the best suited models or model parameters. In the case of feature space heterogeneity constructing the component classifiers on different feature subsets and different samples might capture the best feature subset for the particular set of instances while a number of weak component classifiers will be generated.

In order to understand how sampling and feature subspace projection techniques work in the case of feature space heterogeneity and with presence of globally irrelevant features we experimented with two representative ensemble generation techniques, Bagging and the Random Subspace Method (RSM).

RSM has been selected for study as a simplest method to manipulate the input features. It was originally designed for decision trees (Ho, 1998) and tried on k-Nearest Neighbors (Ho, 1998a). A variant of its usage with Linear Discriminant Analysis is considered in (Skurichina & Duin, 2001).

In RSM component classifiers are generated in multiple, independent passes. In each pass a subset of p features from m available features is selected randomly. Then a classifier is constructed on these p features by a learning algorithm utilized. Generation of new feature subsets with the subsequent construction of a new component classifier continues until the desired number of component classifies is reached.

Bagging (Brieman, 1996) is based on a repeated bootstrap sampling (Efron & Tibshirani, 1993) and aggregation procedure. In order to generate a component classifier a random sampling is performed taking 63.2% of the training instances from the original N training instances and drawing a sample of N instances from those 63.2% with replacement. Then a classifier is constructed on this sample by a learning algorithm. Generation of new samples continues until the desired number of component classifies is achieved. The ratio 63.2% is substantiated in (Brieman, 1996; Bauer & Kohavi, 1999). Bootstrapping helps to avoid or get less non-representative instances in the training set.

For both RSM and Bagging the simple majority voting has been applied to integrate the component classifiers. In this study the main focus is the performance of the generated component classifiers. Integration of the component classifiers may be different, and the most suitable way of integration for the particular generation technique is a subject for a separate study.

Behavior of individual classifiers is strongly data dependant. For example, when a classifier is designed to find a linear boundary between the classes it is likely to work well on the well-separated clouds-like data. However, when one class is surrounded by the the other class, this kind of classifier will not be effective. Thus, compatibility of data characteristics and decision boundary produced by classifier is very important for understanding and improving prediction performance. Here we experimented with two classifiers: the Nearest Mean classifier and the Nearest Neighbor classifier.

The Nearest Mean Classifier (Fukunaga, 1990, Skurichina *et al.*, 2002.) minimizes the distance between the target instance and a class mean. Nearest Mean produces a linear decision boundary.

The Nearest Neighbor classifier, or instance-based classifier, classifies an unknown object to the class of its nearest neighbor in the feature space using, most commonly, Euclidian metrics. The decision boundary of the nearest neighbor rule consists of the boundaries of the Voronoi regions[2] that separate the regions of different classes (Toussaint, 2002).

The question whether ensemble learning will improve accuracy for heterogeneous data deserves further investigation and leads to consideration of many subsidiary questions. The most important of them are (1) which ensemble technique suits better for heterogeneous data, (2) how this technique can be modified to achieve the best results, and (3) how these results can be explained regarding data characteristics.

Advances in understanding the nature of real-world data depend on the knowledge of data characteristics, which might explain the performance of an individual classifier and ensemble of classifiers. Traditionally ensemble is a collection of learning models large enough to correct biases of individual classifiers and/or reduce variability of ensemble predictions. Here we argue that ensemble may be viewed as a collection of locally good learning models for heterogeneous data. Selective application of those models to classify the target instances may promote coverage of local homogeneous regions resulting in accuracy improvement. Ensemble technique for heterogeneous data is considered here as an alternative to the approaches reviewed in Section 2.

4 THE EXPERIMENTAL STUDY

Bagging and RSM have been selected for the experimental study on different synthetic data sets. These data sets represent: (1) the case with all informative features, (2) the case of globally irrelevant features, and (3) the case of locally relevant features with control features (feature space heterogeneity). RSM is tested with different number of features per subset.

[2] A Voronoi diagram is a partition of space into regions, within which all points are closer to some particular training point than to any other training point. By definition of the Voronoi region, every point in a region is closer to the region's training point than to any other training point, and thus this training point must be the nearest neighbor of any new point inside the region.

Two Euclidian distance-based classifiers, Nearest Mean and Nearest Neighbor, are used as learning algorithms.

In this study the number of component classifiers in ensemble is 25, though the accuracy of Bagging can be further improved using more than 25 classifiers.

Training accuracy is a ratio of correctly classified instances to the total number of instances in the training set. This accuracy may assist in assessment of capabilities of an individual classifier at the particular classification problem, i.e. reveal the degree of discrepancy between the real class boundary and the decision boundary produced by a classifier.

Generalization accuracy is a ratio of correctly classified instances to the total number of instances in the test set.

We use the diversity measure calculated as an average difference in predictions between all pairs of classifiers. The diversity is calculated similarly to (Prodromidis & Stolfo, 1998) where it was used as a measure of independence of the component classifiers. The modified formula for calculating the approximated diversity Div_i of a classifier h_i is:

$$Div_i = \frac{\sum_{j=1}^{M} \sum_{k=1,k\neq i}^{S} Dif(h_i(\mathbf{x_j}), h_k(\mathbf{x_j}))}{M \cdot (S-1)} \tag{1}$$

where S denotes the number of the component classifiers, $h_i(\mathbf{x_j})$ denotes the classification of the vector $\mathbf{x_j}$ by the classifier h_i, and *Dif(a,b)* is zero if the classifications a and b are the same and one if they are different, and M is the number of instances in the test set. The diversity of an ensemble is calculated as the average diversity of all the component classifiers (2).

$$Div = \frac{\sum_{i=1}^{S} Div_i}{S} \tag{2}$$

4.1 Study of feature subspace projection and sampling on data without irrelevant features

In order to explore a behavior of the selected feature subspace projection and sampling techniques, first, we run them on synthetic Gaussian data introduced in the ELENA project (Avilcs-Cruz *et al.*, 1995).

Gaussian data sets G_2D - G_8D represent different dimensionalities of the input vectors (2-8 features), heavily overlapped distributions and non-linear separability. The class 0 is a multivariate normal distribution with zero mean and standard deviation equal to 1 in all dimensions, and the class 1 is a normal distribution with zero mean and standard deviation equal to 2 in all dimensions. For these data sets, same as for Clouds and Concentric data sets, 500 instances have been selected from the original 5000 instances for each data set, 250 for each class. 250 instances have been used as a training set, and another 250 instances as a test set.

The Clouds data set represents heavy intersection of the class distributions and high degree of non-linearity of the class boundaries. Distributions are 2-dimentional (2 features). There are two classes, w_0 and w_1.

Classifier	G_2D	G_3D	G_4D	G_5D	G_6D	G_7D	G_8D	Clouds	Concentric
	Training accuracy, %								
NMean	53.60	54.00	55.20	55.20	57.60	58.00	57.60	71.20	56.80
1-NNeigh	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.0
7-NNeigh	76.40	81.20	80.00	78.40	75.60	72.40	70.40	84.00	95.60
	Generalization accuracy, %								
NMean	56.80	54.80	55.20	52.80	54.80	54.00	53.60	68.40	48.00
1-NNeigh	67.60	66.80	67.60	69.60	75.20	70.40	71.60	82.00	97.20
7-NNeigh	70.00	67.60	74.00	79.20	74.40	70.40	68.40	82.80	96.40

Table 1. Training and generalization accuracies of the Nearest Mean, 1- and 7- Nearest Neighbor individual classifiers on the Gaussian, Clouds and Concentric data from the ELENA project.

Class 0 is a sum of three different Gaussian distributions:

$$p(x|\omega_0) = \{p_1(x|\omega_0) + p_2(x|\omega_0) + p_3(x|\omega_0)\}/4 \tag{3}$$

with

$$p_j(x|\omega_0) = (1/2p_i)(1/s_j^x)(1/s_j^y) \cdot \exp\left\{-(x - m_j^x)^2/2(s_j^x)^2 - (y - m_j^y)^2/2(s_j^y)^2\right\} \tag{4}$$

where m_j^x and m_j^y are the x and y means of the j's distribution while s_j^x, s_j^y are their x and y standard deviations:

	s_j^x	s_j^y	m_j^x	m_j^y
$p_1(x\|\omega_0)$	0.2	0.2	0.0	0.0
$p_2(x\|\omega_0)$	0.2	0.2	0.0	0.0
$p_3(x\|\omega_0)$	0.2	0.1	0.2	0.1

Class 1 is a single normal distribution:

$$p(x|\omega_1) = (1/4p_i) \cdot \exp\left\{-x^2/2\right\} \tag{5}$$

In the Concentric data set one class is nested in the other without overlapping. This data set consists of two 2-dimentional uniform concentric circular distributions each representing a class. The data is entirely contained in the square (0,0), (1,1). The points of class 0 are uniformly distributed in a circle of radius 0.3 centered on (0.5,0.5). The points of class 1 are uniformly distributed in a ring centered on (0.5,0.5) with inner and outer radius equal to 0.3 and 0.5, respectively.

Training and generalization accuracies of Nearest Mean, 1- and 7-Nearest Neighbor on these data sets are shown in Table 1.

Classifier	G_2D	G_3D	G_4D	G_5D	G_6D	Clouds	Concentric
	Bagging, NMean, Generalization accuracy, %						
Ensemble	56.00	54.80	55.60	54.00	52.00	71.20	46.40
Min acc.	46.00	46.40	47.20	48.80	41.20	66.40	45.60
Max acc.	58.00	57.60	56.80	56.40	60.00	72.40	54.40
	Bagging, 1-NNeigh, Generalization accuracy, %						
Ensemble	68.40	66.80	70.00	72.80	75.60	81.20	97.60
Min acc.	63.20	61.20	66.00	69.60	69.60	78.00	93.60
Max acc.	68.80	69.60	72.00	74.40	76.40	82.40	98.40
	Bagging, 7-NNeigh, Generalization accuracy, %						
Ensemble	67.20	74.80	73.20	80.00	76.40	81.60	95.60
Min acc.	62.40	65.20	67.60	70.40	65.60	76.80	91.20
Max acc.	71.20	78.00	76.80	78.80	76.80	83.20	96.80

Table 2. Generalization accuracy of Bagging (25 component classifiers) on the Gaussian, Clouds and Concentric data from the ELENA project.

Since both Gaussian and Concentric data sets have non-linear class boundaries, Nearest Mean shows a low accuracy on these data sets while 1- and 7-Nearest Neighbor were more accurate. On the Clouds data set Nearest Mean performance was better due to the class structure. Tables 2 and 3 show performance of Bagging and RSM.

All the data sets considered have *interacting* features, i.e. features whose values are dependant on the values of other features and the class, and as such, provide further information about the class. Discarding some of those features will lower the accuracy. That is why the results of 2-feature RSM for G_ 3D-G_ 8D Gaussian data sets demonstrate accuracy reduction. Accuracy decreased more for Nearest Neighbor with RSM then for Nearest Mean. Bagging has raised accuracy for all data sets only slightly. Minimum and maximum accuracy of component classifiers in ensemble does not vary significantly for both RSM and Bagging.

4.2 Study of feature subspace projection and sampling on data with globally irrelevant features

In order to evaluate the selected ensemble techniques on data with globally irrelevant (noisy) features the following synthetic data set has been created. The Global relevance data set has 8 binary features from which only 3 are relevant. There are 3 classes and 256 instances. Training and test sets contain 128 instances each. In the training set there are 43 instances in classes 1 and 2, 42 instances in class 3; in the test set there are 53 instances in class 1, 21 instances in class 2, and 54 instances in class 3. In Table 4 the rules of class presentation are given.

Table 5 shows training and generalization accuracies of Nearest Mean and 1-Nearest

Classifier	G_3D	G_4D	G_5D	G_6D	G_7D	G_8D
	RSM, 2 features, NMean, Generalization accuracy, %					
Ensemble	51.20	49.20	52.00	49.20	52.00	51.60
Min acc.	45.20	44.40	45.20	44.40	44.40	43.60
Max acc.	59.60	59.60	59.60	56.80	57.60	56.80
	RSM, 2 features, 1-NNeigh, Generalization accuracy, %					
Ensemble	58.40	57.20	44.40	47.60	48.40	56.80
Min acc.	52.80	51.60	41.60	46.00	42.40	48.00
Max acc.	57.20	57.20	53.60	52.80	55.20	56.80
	RSM, 2 features, 7-NNeigh, Generalization accuracy, %					
Ensemble	55.20	58.00	48.80	48.40	55.60	54.80
Min acc.	50.80	54.00	44.00	46.80	48.80	45.20
Max acc.	56.00	56.80	55.60	52.40	54.80	56.00

Table 3. Generalization accuracy of the Random Subspace Method (2 features per subset, 25 component classifiers) on the Gaussian, Clouds and Concentric data sets from the ELENA project.

Neighbor on this data.

Tables 6-9 present the results of RSM application with different amount of features per subset (3-6). Relevant features are marked in **bold**. As we can see from Table 6, the first component classifier picked up all 3 relevant features, thus its accuracy should be the best one. However, many other Nearest Mean component classifiers containing only one relevant feature sometimes are more accurate than the first classifier. When irrelevant features are selected Nearest Neighbor does not produce randomly as high accuracies as Nearest Mean. Since the integration method is majority voting, the ensemble decision is rather random. In this case some integration method based on classifier selection would be appropriate. Diversity of the RSM ensemble is high in the majority of cases.

In Table 7 accuracy of 4 features RSM is shown. Classifier 11 contains all 3 relevant features plus one redundant one injecting some noise. However, comparatively to the previous 3-RSM ensembles more component classifiers contain 2 relevant features, thus Nearest Neighbor ensemble accuracy is higher. Nearest Mean accuracy remains on the same level since it is still affected by random decisions. We may say that change in Nearest Neighbor accuracy is less liable to the noise provided by inclusion of irrelevant features and resulting majority voting accuracy of ensemble is higher.

As we can see from Tables 8 and 9 the observation made before holds true. In the case of 6 features per subset (Table 9) along with the relevant features the same number of irrelevant ones is included for the best classifiers, thus the accuracy growth is not that significant for Nearest Neighbor while for Nearest Mean the accuracy level has been even lower. Table 10 demonstrates Bagging's performance when irrelevant features are present. Since all features

Feat 1	Feat 2	Feat 3	Feat 4	Feat 5	Feat 6	Feat 7	Feat 8	Class
irrelevant	irrelevant	irrelevant	irrelevant	irrelevant	0	0	0	1
irrelevant	irrelevant	irrelevant	irrelevant	irrelevant	0	0	1	
irrelevant	irrelevant	irrelevant	irrelevant	irrelevant	0	1	0	
irrelevant	irrelevant	irrelevant	irrelevant	irrelevant	0	1	1	2
irrelevant	irrelevant	irrelevant	irrelevant	irrelevant	1	0	0	
irrelevant	irrelevant	irrelevant	irrelevant	irrelevant	1	0	1	3
irrelevant	irrelevant	irrelevant	irrelevant	irrelevant	1	1	0	
irrelevant	irrelevant	irrelevant	irrelevant	irrelevant	1	1	1	

Table 4. Specification of the synthetic problem for the Global relevance data set.

Classifier	Training accuracy, %	Generalization accuracy, %
Nearest Mean	83.59	75.78
1-Nearest Neighbor	100.00	78.91

Table 5. Training and generalization accuracy of the Nearest Mean and 1-Nearest Neighbor individual classifiers on the synthetic data set Global relevance.

are used for classification, we do not see any significant variability in accuracy among the component classifiers. However, due to bootstrapping ensemble accuracy has been increased for both Nearest Mean and Nearest Neighbor. In this case integration by voting is justified.

We may conclude that combination of RSM and Bagging may improve accuracy for the data with globally irrelevant features. However, some prior feature selection procedure providing a subset of good features unique for the whole data as an input for Bagging would work even better in the case of global correlations in data. But in the situation when local correlations are present this global feature selection may fail contrary to the combination of feature subspace projection and sampling. We model such a situation with synthetic data in the following section.

4.3 Study of feature subspace projection and sampling on data with locally relevant features

As it was stated in Section 2, some correlations in data may be local, and therefore, some features may be relevant only in the particular region of the instance set. When control features are available they specify the local relevance rules identifying those regions.

Here we model the situation with local relevance specified by two control features, feature 1 and 2. By their joint values the relevance of a feature subset 3,4,5 or 6,7,8 is determined (Table 11). All features are binary, irrelevant features take random values. The Local relevance data set has 3 classes, 256 instances evenly distributed between classes, same as in

Classifier	Features	Random Subspace Method, 3 features per subset			
		Nearest Mean		1-Nearest Neighbor	
		Accuracy	Diversity	Accuracy	Diversity
1	**8**,**6**,**7**	71.09	0.42	100.00	0.58
2	**7**,4,2	41.41	0.54	28.12	0.53
3	**8**,2,5	42.19	0.56	28.12	0.60
4	5,4,3	26.56	0.70	28.91	0.54
5	2,5,**6**	71.88	0.39	56.25	0.53
6	**7**,3,5	52.34	0.47	25.00	0.53
7	**7**,5,4	52.34	0.47	32.03	0.51
8	**6**,2,5	71.88	0.39	56.25	0.53
9	**6**,**8**,4	76.56	0.43	75.00	0.53
10	1,5,3	25.78	0.67	31.25	0.61
11	5,3,**7**	52.34	0.47	25.00	0.53
12	2,4,5	21.88	0.74	27.34	0.53
13	5,4,**7**	52.34	0.47	32.03	0.51
14	**8**,5,4	56.25	0.54	23.44	0.56
15	5,4,**6**	83.59	0.37	52.34	0.43
16	3,**7**,**6**	71.09	0.42	68.75	0.56
17	**6**,**8**,3	76.56	0.43	72.66	0.55
18	**6**,5,4	83.59	0.37	52.34	0.43
19	4,3,1	26.56	0.71	29.69	0.57
20	**6**,5,4	83.59	0.37	52.34	0.43
21	5,**6**,3	83.59	0.37	51.56	0.51
22	5,**7**,3	52.34	0.47	25.00	0.53
23	2,**8**,5	42.19	0.56	28.12	0.60
24	4,**6**,5	83.59	0.37	52.34	0.43
25	4,**6**,3	83.59	0.37	53.12	0.48
Ensemble	—	82.03	0.48	50.00	0.52

Table 6. Generalization accuracy of the Random Subspace Method (3 features per subset, 25 component classifiers) with the Nearest Mean and 1-Nearest Neighbor base classifiers on the synthetic data set Global relevance.

Classifier	Features	Random Subspace Method, 4 features per subset			
		Nearest Mean		1-Nearest Neighbor	
		Accuracy	Diversity	Accuracy	Diversity
1	**7**,**8**,5,3	71.09	0.32	100.00	0.40
2	4,2,**8**,5	42.19	0.58	32.81	0.61
3	5,4,3,2	24.22	0.73	32.03	0.65
4	5,**6**,**7**,3	71.09	0.32	76.56	0.42
5	**7**,5,4,**6**	71.09	0.32	76.56	0.39
6	2,5,**6**,**8**	77.34	0.36	78.12	0.44
7	4,1,5,3	25.00	0.72	30.47	0.60
8	5,3,**7**,2	43.75	0.52	35.94	0.61
9	4,5,**7**,**8**	34.38	0.54	50.00	0.56
10	5,4,**6**,3	83.59	0.31	64.84	0.39
11	3,**7**,**6**,**8**	71.09	0.32	100.00	0.40
12	3,**6**,5,4	83.59	0.31	64.84	0.39
13	4,3,1,**6**	83.59	0.31	63.28	0.43
14	5,4,**6**,3	83.59	0.31	64.84	0.39
15	5,**7**,3,2	43.75	0.52	35.94	0.61
16	**8**,5,4,**6**	78.91	0.34	79.69	0.40
17	**6**,4,5,3	83.59	0.31	64.84	0.39
18	5,**6**,4,**7**	71.09	0.32	76.56	0.39
19	**7**,4,**8**,5	34.38	0.54	50.00	0.56
20	5,2,**8**,**6**	77.34	0.36	78.12	0.44
21	3,4,1,**6**	83.59	0.31	63.28	0.43
22	**7**,4,1,3	46.09	0.53	35.16	0.59
23	5,4,**8**,3	56.25	0.54	32.03	0.60
24	5,**6**,**7**,3	71.09	0.32	76.56	0.42
25	**8**,**6**,3,2	79.69	0.34	78.12	0.46
Ensemble	—	83.59	0.42	81.25	0.48

Table 7. Generalization accuracy of the Random Subspace Method (4 features per subset, 25 component classifiers) with the Nearest Mean and 1-Nearest Neighbor base classifiers on the synthetic data set Global relevance.

Classifier	Features	Random Subspace Method, 5 features per subset			
		Nearest Mean		1-Nearest Neighbor	
		Accuracy	Diversity	Accuracy	Diversity
1	**7**,**8**,**6**,4,2	78.12	0.23	100.00	0.32
2	**8**,2,5,4,3	47.66	0.54	32.81	0.61
3	2,5,**6**,**7**,3	71.09	0.25	73.44	0.42
4	**7**,5,4,**6**,2	71.09	0.25	72.66	0.43
5	5,**6**,**8**,4,1	76.56	0.25	75.78	0.37
6	5,3,**7**,2,4	43.75	0.56	32.81	0.64
7	5,4,**7**,**8**,**6**	75.00	0.24	100.00	0.32
8	3,**7**,**6**,**8**,5	78.91	0.23	100.00	0.32
9	5,4,3,1,**6**	78.91	0.27	54.69	0.49
10	5,4,**6**,3,**7**	71.09	0.25	71.88	0.42
11	3,2,**8**,5,4	47.66	0.54	32.81	0.61
12	**6**,4,5,3,**7**	71.09	0.25	71.88	0.42
13	**7**,4,**8**,5,2	35.16	0.57	46.88	0.60
14	**8**,6,3,4,1	77.34	0.26	78.12	0.35
15	1,4,**6**,**7**,3	71.09	0.25	69.53	0.44
16	5,4,**8**,3,**6**	77.34	0.25	71.88	0.36
17	**6**,5,**7**,3,**8**	78.91	0.23	100.00	0.32
18	**6**,**8**,3,2,5	77.34	0.25	68.75	0.38
19	2,**6**,5,**8**,3	77.34	0.25	68.75	0.38
20	5,**8**,3,**7**,4	36.72	0.58	41.41	0.61
21	**6**,3,5,**8**,4	77.34	0.25	71.88	0.36
22	**6**,5,4,**8**,2	78.91	0.25	78.91	0.35
23	**8**,**6**,4,3,1	77.34	0.26	78.12	0.35
24	**6**,**8**,3,5,4	77.34	0.25	71.88	0.36
25	3,**6**,5,4,**8**	77.34	0.25	71.88	0.36
Ensemble	—	83.59	0.31	90.62	0.42

Table 8. Generalization accuracy of the Random Subspace Method (5 features per subset, 25 component classifiers) with the Nearest Mean and 1-Nearest Neighbor base classifiers on the synthetic data set Global relevance.

Classifier	Features	Random Subspace Method, 6 features per subset			
		Nearest Mean		1-Nearest Neighbor	
		Accuracy	Diversity	Accuracy	Diversity
1	**7**,**8**,**6**,4,2,5	73.44	0.11	100.00	0.25
2	5,4,3,2,**6**,**7**	71.09	0.11	67.97	0.35
3	3,**7**,5,4,**6**,2	71.09	0.11	67.97	0.35
4	5,**6**,**8**,4,1,3	77.34	0.23	67.97	0.38
5	5,3,**7**,2,4,**8**	35.16	0.55	37.50	0.63
6	5,4,**6**,3,**7**,**8**	75.00	0.10	90.62	0.25
7	3,**6**,5,4,1,**7**	71.09	0.11	67.97	0.36
8	3,2,**8**,5,4,**6**	78.91	0.21	69.53	0.36
9	**6**,4,5,3,**7**,**8**	75.00	0.10	90.62	0.25
10	4,**5**,2,**8**,**6**,3	78.91	0.21	69.53	0.36
11	4,1,**6**,**7**,3,5	71.09	0.11	67.97	0.36
12	5,4,**8**,3,**6**,**7**	75.00	0.10	90.62	0.25
13	3,**8**,**6**,2,5,**7**	74.22	0.10	87.50	0.29
14	4,**6**,3,5,**8**,2	78.91	0.21	69.53	0.36
15	**8**,**6**,4,3,1,5	77.34	0.23	67.97	0.38
16	4,3,**6**,5,**8**,**7**	75.00	0.10	90.62	0.25
17	**6**,5,3,2,4,**8**	78.91	0.21	69.53	0.36
18	**8**,**7**,3,5,**6**,4	75.00	0.10	90.62	0.25
19	**7**,**6**,3,5,**8**,4	75.00	0.10	90.62	0.25
20	3,4,**7**,**6**,5,**8**	75.00	0.10	90.62	0.25
21	5,**6**,2,1,**8**,**7**	75.78	0.13	90.62	0.30
22	**6**,3,**7**,5,2,4	71.09	0.11	67.97	0.35
23	5,2,4,**7**,**6**,3	71.09	0.11	67.97	0.35
24	3,5,4,**7**,2,**6**	71.09	0.11	67.97	0.35
25	1,5,**7**,4,**6**,2	71.88	0.10	64.06	0.43
Ensemble	—	75.00	0.15	92.19	0.33

Table 9. Generalization accuracy of the Random Subspace Method (6 features per subset, 25 component classifiers) with the Nearest Mean and 1-Nearest Neighbor base classifiers on the synthetic data set Global relevance.

Classifier	Bagging			
	Nearest Mean		1-Nearest Neighbor	
	Accuracy	Diversity	Accuracy	Diversity
1	71.88	0.32	75.78	0.21
2	70.31	0.32	71.88	0.27
3	73.44	0.30	77.34	0.22
4	76.56	0.30	79.69	0.24
5	77.34	0.31	76.56	0.24
6	69.53	0.29	78.91	0.32
7	71.09	0.30	74.22	0.24
8	67.97	0.31	78.91	0.27
9	70.31	0.30	75.78	0.30
10	74.22	0.33	73.44	0.28
11	76.56	0.31	78.12	0.23
12	71.88	0.29	79.69	0.31
13	75.00	0.31	73.44	0.22
14	72.66	0.32	71.88	0.26
15	67.97	0.28	79.69	0.26
16	71.09	0.32	75.00	0.24
17	70.31	0.32	74.22	0.24
18	90.62	0.29	78.91	0.28
19	78.91	0.28	77.34	0.24
20	67.97	0.30	75.78	0.27
21	68.75	0.35	64.84	0.24
22	69.53	0.31	73.44	0.28
23	75.00	0.32	77.34	0.25
24	72.66	0.30	75.78	0.26
25	72.66	0.29	77.34	0.21
Ensemble	81.25	0.31	90.62	0.25

Table 10. Generalization accuracy of Bagging (25 component classifiers) with the Nearest Mean and 1-Nearest Neighbor base classifiers on the synthetic data set Global relevance.

the Global relevance data set. Training and test sets contain 128 instances each. This data set has been created by relabeling classes of the Global relevance data set according to the rules presented in Table 11.

Training set includes 58 instances from the subproblem 1 (relevant features 3,4,5), and 70 instances of the subproblem 2 (relevant features 6,7,8). Test set includes 65 instances of subproblem 1 and 63 instances of subproblem 2. Accuracies of individual classifiers on this data set are presented in Table 12.

Tables 13-17 depict the results of RSM application. Features relevant for the subproblem 1 are marked in *italic*, and features relevant for the subproblem 2 are marked in **bold**.

As we can see, performance of classifiers 1 and 4 are different for both Nearest Mean and Nearest Neighbor, despite the fact that both component classifiers were built only on features from the same subproblem. This can be explained by non-equal proportion of instances from both subproblems in the training set. For example, when classifier 4 was trained, the real number of nearest neighbors representing subproblem 1 has been lowered, thus for each target instance representing subproblem 1 decision has been biased. That explains low accuracy of this classifier. The opposite situation has happened to classifier 1 resulting in high accuracy.

All classifiers have been trained on instances of both subproblems, i.e. some of the instances are noisy. Due to the random values of the irrelevant features decisions of both classifiers Nearest Mean and Nearest Neighbor can be considered as random in many cases. For example, first classifier is built on features relevant for only half of the instances, and then another half will be noisy. Classifier 12 is built on one control (globally relevant) feature, and two features of the same local region. Considering the accuracies obtained one may notice that the degree of relevance of control and primary features is different.

Table 17 demonstrates performance of Bagging in the case of locally relevant features. Each bootstrapping sample has included an even number of instances representing two subproblems. Due to bootstrapping, accuracy of ensemble has been increased for both Nearest Mean and Nearest Neighbor, same as for the Global relevance data set.

A summary of our experimental results is the following. Some classifiers are intrinsically weak for certain problem domain and fail to produce correct decision boundary. Nearest Mean is not strong for our synthetic domains, while 1-Nearest Neighbor demonstrates better performance. Thus, geometry of the problem domain and the decision boundary produced by a classifier are important in understanding and improving classification accuracy.

Bagging improves accuracy because of the bootstrap sampling that is a general technique for error variance reduction. Each classifier is constructed on the instances that are distributed evenly for the subproblems of the Local relevance data set. In order to have a sampling technique that will more or less cover the local regions in the dataset, it should be biased towards those regions.

RSM does not guarantee to produce the best classifier with all relevant features for a selected number of trials. However, the more sophisticated strategy for searching feature subsets among all possible variants may perform better.

When the classifier with all relevant features is selected, it represents the best possible accuracy we could get for the data set, but the rest of the classifiers bring their random

Feat 1	Feat 2	Feat 3	Feat 4	Feat 5	Feat 6	Feat 7	Feat 8	Class
0	0	irrelevant	irrelevant	irrelevant	0	0	0	1
0	0	irrelevant	irrelevant	irrelevant	0	0	1	
0	0	irrelevant	irrelevant	irrelevant	0	1	0	
0	0	irrelevant	irrelevant	irrelevant	0	1	1	2
0	0	irrelevant	irrelevant	irrelevant	1	0	0	
0	0	irrelevant	irrelevant	irrelevant	1	0	1	3
0	0	irrelevant	irrelevant	irrelevant	1	1	0	
0	0	irrelevant	irrelevant	irrelevant	1	1	1	
0	1	0	0	0	irrelevant	irrelevant	irrelevant	1
0	1	0	0	1	irrelevant	irrelevant	irrelevant	
0	1	0	1	0	irrelevant	irrelevant	irrelevant	
0	1	0	1	1	irrelevant	irrelevant	irrelevant	2
0	1	1	0	0	irrelevant	irrelevant	irrelevant	
0	1	1	0	1	irrelevant	irrelevant	irrelevant	3
0	1	1	1	0	irrelevant	irrelevant	irrelevant	
0	1	1	1	1	irrelevant	irrelevant	irrelevant	
1	0	0	0	0	irrelevant	irrelevant	irrelevant	1
1	0	0	0	1	irrelevant	irrelevant	irrelevant	
1	0	0	1	0	irrelevant	irrelevant	irrelevant	
1	0	0	1	1	irrelevant	irrelevant	irrelevant	2
1	0	1	0	0	irrelevant	irrelevant	irrelevant	
1	0	1	0	1	irrelevant	irrelevant	irrelevant	3
1	0	1	1	0	irrelevant	irrelevant	irrelevant	
1	0	1	1	1	irrelevant	irrelevant	irrelevant	
1	1	irrelevant	irrelevant	irrelevant	0	0	0	1
1	1	irrelevant	irrelevant	irrelevant	0	0	1	
1	1	irrelevant	irrelevant	irrelevant	0	1	0	
1	1	irrelevant	irrelevant	irrelevant	0	1	1	2
1	1	irrelevant	irrelevant	irrelevant	1	0	0	
1	1	irrelevant	irrelevant	irrelevant	1	0	1	3
1	1	irrelevant	irrelevant	irrelevant	1	1	0	
1	1	irrelevant	irrelevant	irrelevant	1	1	1	

Table 11. Definition of the synthetic problem for the Local relevance data set.

Classifier	Training accuracy, %	Generalization accuracy, %
Nearest Mean	83.59	75.78
1-Nearest Neighbor	100.00	74.22

Table 12. Training and generalization accuracy of the Nearest Mean and 1-Nearest Neighbor individual classifiers on the synthetic data set Local relevance.

decisions too, which can hurt the ensemble accuracy. Thus, in the situation with many irrelevant features when RSM is applied, the strategy for integrating classifiers should be selection rather than combination. Appearance of the classifier with all relevant features should be guaranteed.

When RSM is applied to the problem with local relevance and control features, the difference in relevance level is exhibited: locally relevant (primary) features are more important than control features. RSM does not work for local relevance. A classifier that is constructed on the locally relevant features for a particular subproblem is trained on the instances from all subproblems. Thus, accuracy of such classifier is low. When the classifier uses only features that are relevant for all subproblems, it may give the best accuracy. Sampling technique may select instances covering different subproblems in different component classifiers. Then each component classifier will use both relevant and irrelevant features. Thus, feature subspace projection is needed to eliminate those irrelevant features. Combination of RSM and sampling technique that would be biased to select instances from the particular subproblem would have an advantage.

5 DISCUSSION ON COMBINING ENSEMBLE TECHNIQUES

In this section we consider advantages of combining feature subspace projection, sampling and class-decomposition techniques as a solution for heterogeneity problem. Related works on combined ensemble techniques are discussed.

In several recent articles a combination of different ensemble techniques for particular learning algorithms has been studied. Mostly, different sampling techniques have been combined with Boosting (Freud & Schapire, 1996), which is effective for a wide variety of classification tasks. However, only a few studies of combining feature subspace projection and sampling ensemble techniques (Zheng *et al.*, 1998; Zheng & Webb, 1999; Brieman, 2001) and class-decomposition (Oza & Tumer, 1999) that is potentially beneficial for the heterogeneity problem has been done.

Bagging and Boosting are two representatives of sampling that can significantly decrease the error rate of classification. They repeatedly build different classifiers using a base learning algorithm by changing the distribution of the training set. Boosting is often observed as more accurate than Bagging, although the performance of Boosting varies more than that of

Classifier	Features	Random Subspace Method, 3 features per subset			
		Nearest Mean		1-Nearest Neighbor	
		Accuracy	Diversity	Accuracy	Diversity
1	**8**,**6**,**7**	71.09	0.42	100.00	0.52
2	**7**,*4*,2	41.41	0.54	32.03	0.65
3	**8**,2,*5*	42.19	0.56	38.28	0.51
4	*5*,*4*,*3*	26.56	0.70	35.16	0.48
5	2,*5*,**6**	71.88	0.39	52.34	0.46
6	**7**,*3*,*5*	52.34	0.47	31.25	0.51
7	**7**,*5*,*4*	52.34	0.47	38.28	0.56
8	**6**,2,*5*	71.88	0.39	52.34	0.46
9	**6**,**8**,*4*	76.56	0.43	75.00	0.51
10	1,*5*,*3*	25.78	0.67	32.03	0.55
11	*5*,*3*,**7**	52.34	0.47	31.25	0.51
12	2,*4*,*5*	21.88	0.74	31.25	0.49
13	*5*,*4*,**7**	52.34	0.47	38.28	0.56
14	**8**,*5*,*4*	56.25	0.54	36.72	0.60
15	*5*,*4*,**6**	83.59	0.37	65.62	0.41
16	*3*,**7**,**6**	71.09	0.42	81.25	0.52
17	**6**,**8**,*3*	76.56	0.43	75.00	0.51
18	**6**,*5*,*4*	83.59	0.37	65.62	0.41
19	*4*,*3*,1	26.56	0.71	32.81	0.61
20	**6**,*5*,*4*	83.59	0.37	65.62	0.41
21	*5*,**6**,*3*	83.59	0.37	57.81	0.41
22	*5*,**7**,*3*	52.34	0.47	31.25	0.51
23	2,**8**,*5*	42.19	0.56	38.28	0.51
24	*4*,**6**,*5*	83.59	0.37	65.62	0.41
25	*4*,**6**,*3*	83.59	0.37	57.81	0.49
Ensemble	—	82.03	0.48	59.38	0.50

Table 13. Generalization accuracy of the Random Subspace Method (3 features per subset, 25 component classifiers) with the Nearest Mean and 1-Nearest Neighbor base classifiers on the synthetic data set Local relevance.

Classifier	Features	Random Subspace Method, 4 features per subset			
		Nearest Mean		1-Nearest Neighbor	
		Accuracy	Diversity	Accuracy	Diversity
1	**7**,**8**,*5*,*3*	71.09	0.32	100.00	0.40
2	*4*,*2*,**8**,*5*	42.19	0.58	41.41	0.57
3	*5*,*4*,*3*,2	24.22	0.73	31.25	0.58
4	*5*,**6**,**7**,*3*	71.09	0.32	75.00	0.42
5	**7**,*5*,*4*,**6**	71.09	0.32	75.78	0.39
6	*2*,*5*,**6**,**8**	77.34	0.36	74.22	0.41
7	*4*,1,*5*,*3*	25.00	0.72	32.81	0.58
8	*5*,*3*,**7**,2	43.75	0.52	27.34	0.58
9	*4*,*5*,**7**,**8**	34.38	0.54	51.56	0.57
10	*5*,*4*,**6**,*3*	83.59	0.31	70.31	0.37
11	*3*,**7**,**6**,**8**	71.09	0.32	100.00	0.40
12	*3*,**6**,*5*,*4*	83.59	0.31	70.31	0.37
13	*4*,*3*,1,**6**	83.59	0.31	60.16	0.47
14	*5*,*4*,**6**,*3*	83.59	0.31	70.31	0.37
15	*5*,**7**,*3*,2	43.75	0.52	27.34	0.58
16	**8**,*5*,*4*,**6**	78.91	0.34	78.12	0.40
17	**6**,*4*,*5*,*3*	83.59	0.31	70.31	0.37
18	*5*,**6**,*4*,**7**	71.09	0.32	75.78	0.39
19	**7**,*4*,**8**,*5*	34.38	0.54	51.56	0.57
20	*5*,2,**8**,**6**	77.34	0.36	74.22	0.41
21	*3*,*4*,1,**6**	83.59	0.31	60.16	0.47
22	**7**,*4*,1,*3*	46.09	0.53	33.59	0.63
23	*5*,*4*,**8**,*3*	56.25	0.54	37.50	0.57
24	*5*,**6**,**7**,*3*	71.09	0.32	75.00	0.42
25	**8**,**6**,*3*,2	79.69	0.34	75.78	0.45
Ensemble	—	83.59	0.42	78.12	0.47

Table 14. Generalization accuracy of the Random Subspace Method (4 features per subset, 25 component classifiers) with the Nearest Mean and 1-Nearest Neighbor base classifiers on the synthetic data set Local relevance.

Classifier	Features	Random Subspace Method, 5 features per subset			
		Nearest Mean		1-Nearest Neighbor	
		Accuracy	Diversity	Accuracy	Diversity
1	**7**,**8**,**6**,*4*,2	78.12	0.31	100.00	0.31
2	**8**,2,*5*,*4*,*3*	47.66	0.57	35.16	0.57
3	2,*5*,**6**,**7**,*3*	71.09	0.41	71.09	0.41
4	**7**,*5*,*4*,**6**,2	71.09	0.38	73.44	0.38
5	*5*,**6**,**8**,*4*,1	76.56	0.34	76.56	0.34
6	*5*,*3*,**7**,2,*4*	43.75	0.63	28.91	0.63
7	*5*,*4*,**7**,**8**,**6**	75.00	0.31	100.00	0.31
8	*3*,**7**,**6**,**8**,*5*	78.91	0.31	100.00	0.31
9	*5*,*4*,*3*,1,**6**	78.91	0.45	55.47	0.45
10	*5*,*4*,**6**,*3*,**7**	71.09	0.41	72.66	0.41
11	*3*,2,**8**,*5*,*4*	47.66	0.57	35.16	0.57
12	**6**,*4*,*5*,*3*,**7**	71.09	0.41	72.66	0.41
13	**7**,*4*,**8**,*5*,2	35.16	0.57	50.00	0.57
14	**8**,*6*,*3*,*4*,1	77.34	0.35	75.78	0.35
15	1,*4*,**6**,**7**,*3*	71.09	0.44	70.31	0.44
16	*5*,*4*,**8**,*3*,**6**	77.34	0.33	74.22	0.33
17	**6**,*5*,**7**,*3*,**8**	78.91	0.31	100.00	0.31
18	**6**,**8**,*3*,2,*5*	77.34	0.35	72.66	0.35
19	2,**6**,*5*,**8**,*3*	77.34	0.35	72.66	0.35
20	*5*,**8**,*3*,**7**,*4*	36.72	0.61	42.97	0.61
21	**6**,*3*,*5*,**8**,*4*	77.34	0.33	74.22	0.33
22	**6**,*5*,*4*,**8**,2	78.91	0.33	74.22	0.33
23	**8**,**6**,*4*,*3*,1	77.34	0.35	75.78	0.35
24	**6**,**8**,*3*,*5*,*4*	77.34	0.33	74.22	0.33
25	*3*,**6**,*5*,*4*,**8**	77.34	0.33	74.22	0.33
Ensemble	—	83.59	0.40	87.50	0.40

Table 15. Generalization accuracy of the Random Subspace Method (5 features per subset, 25 component classifiers) with the Nearest Mean and 1-Nearest Neighbor base classifiers on the synthetic data set Local relevance.

Classifier	Features	Random Subspace Method, 6 features per subset			
		Nearest Mean		1-Nearest Neighbor	
		Accuracy	Diversity	Accuracy	Diversity
1	**7**,**8**,**6**,*4*,2,*5*	73.44	0.11	100.00	0.23
2	*5*,*4*,*3*,2,**6**,**7**	71.09	0.11	67.19	0.30
3	*3*,**7**,*5*,*4*,**6**,2	71.09	0.11	67.19	0.30
4	*5*,**6**,**8**,*4*,1,*3*	77.34	0.23	71.09	0.36
5	*5*,*3*,**7**,2,*4*,**8**	35.16	0.55	36.72	0.65
6	*5*,*4*,**6**,*3*,**7**,**8**	75.00	0.10	96.88	0.22
7	*3*,**6**,*5*,*4*,1,**7**	71.09	0.11	67.97	0.34
8	*3*,2,**8**,*5*,*4*,**6**	78.91	0.21	67.97	0.34
9	**6**,*4*,*5*,*3*,**7**,**8**	75.00	0.10	96.88	0.22
10	*4*,**5**,2,**8**,**6**,*3*	78.91	0.21	67.97	0.34
11	*4*,1,**6**,**7**,*3*,*5*	71.09	0.11	67.97	0.34
12	*5*,*4*,**8**,*3*,**6**,**7**	75.00	0.10	96.88	0.22
13	*3*,**8**,**6**,2,*5*,**7**	74.22	0.10	90.62	0.27
14	*4*,**6**,*3*,*5*,**8**,2	78.91	0.21	67.97	0.34
15	**8**,**6**,*4*,*3*,1,*5*	77.34	0.23	71.09	0.36
16	*4*,*3*,**6**,*5*,**8**,**7**	75.00	0.10	96.88	0.22
17	**6**,*5*,*3*,2,*4*,**8**	78.91	0.21	67.97	0.34
18	**8**,**7**,*3*,*5*,**6**,*4*	75.00	0.10	96.88	0.22
19	**7**,**6**,*3*,*5*,**8**,*4*	75.00	0.10	96.88	0.22
20	*3*,*4*,**7**,**6**,5,**8**	75.00	0.10	96.88	0.22
21	*5*,**6**,2,1,**8**,**7**	75.78	0.13	96.88	0.25
22	**6**,*3*,**7**,*5*,2,*4*	71.09	0.11	67.19	0.30
23	*5*,2,*4*,**7**,**6**,*3*	71.09	0.11	67.19	0.30
24	*3*,*5*,*4*,**7**,2,**6**	71.09	0.11	67.19	0.30
25	1,*5*,**7**,*4*,**6**,2	71.88	0.10	67.97	0.32
Ensemble	—	75.00	0.15	93.75	0.30

Table 16. Generalization accuracy of the Random Subspace Method (6 features per subset, 25 component classifiers) with the Nearest Mean and 1-Nearest Neighbor base classifiers on the synthetic data set Local relevance.

Classifier	Bagging			
	Nearest Mean		1-Nearest Neighbor	
	Accuracy	Diversity	Accuracy	Diversity
1	71.88	0.32	75.78	0.21
2	70.31	0.32	71.88	0.27
3	73.44	0.30	77.34	0.22
4	76.56	0.30	79.69	0.24
5	77.34	0.31	76.56	0.24
6	69.53	0.29	78.91	0.32
7	71.09	0.30	74.22	0.24
8	67.97	0.31	78.91	0.27
9	70.31	0.30	75.78	0.30
10	74.22	0.33	73.44	0.28
11	76.56	0.31	78.12	0.23
12	71.88	0.29	79.69	0.31
13	75.00	0.31	73.44	0.22
14	72.66	0.32	71.88	0.26
15	67.97	0.28	79.69	0.26
16	71.09	0.32	75.00	0.24
17	70.31	0.32	74.22	0.24
18	90.62	0.29	78.91	0.28
19	78.91	0.28	77.34	0.24
20	67.97	0.30	75.78	0.27
21	68.75	0.35	64.84	0.24
22	69.53	0.31	73.44	0.28
23	75.00	0.32	77.34	0.25
24	72.66	0.30	75.78	0.26
25	72.66	0.29	77.34	0.21
Ensemble	81.25	0.31	90.62	0.25

Table 17. Generalization accuracy of Bagging (25 component classifiers) with the Nearest Mean and 1-Nearest Neighbor base classifiers on the synthetic data set Local relevance.

Bagging (Opitz & Maclin, 1999). These techniques do not rely on any specific assumption about the training data. They can be successfully combined with the other two ensemble generation techniques, which make certain assumptions. However, one should note, that those assumptions are rather drawn from the application experience than taken *a priory* for the design of ensemble generation principles.

Feature subspace projection techniques are potentially useful for heterogeneous data with unstable features that vary in their importance (relevance) across the instance set. Manipulating feature subsets to construct the component classifiers increases the diversity of ensemble assuming that some feature instability exists.

Stochastic Discrimination theory (Kleinberg, 1990) studies various ways to partition the feature space in order to produce classifiers having weak discriminative power but strong generalization power, and combine them in a way to achieve high accuracy. Pseudo-random feature subspace projection methods (Ho, 1998; Zheng & Webb, 1998) are derivations of this theory. They are based on heuristic search and evaluation of different feature subsets for construction of the component classifiers towards effective ensemble prediction.

Feature subspace projection can be performed using feature selection methods that rank features according to some merit measure, such as Information Gain, Contextual Merit, or ReliefF (Apte *et al.*, 1997; Puuronen *et al.*, 2001). Feature subset evaluation heuristics, such as Correlation-based heuristics (Hall, 1999; Puuronen *et al.*, 2001) can also be used for feature subspace projection.

Another group of techniques, class-decomposition techniques, may work well in the class heterogeneity situation. For many practical problems it is likely that class subsamples on which component classifiers are constructed coincide with the natural class subsamples.

Three basic ensemble techniques that exemplify this principle of ensemble generation are one-per-class method, pairwise method and error-correcting output codes (ECOC) (Masulli & Valentini, 2000; Guruswami & Sahai, 1999; Furnkranz, 2002; Dietterich & Bakiri, 1995). One-per-class method is the most straightforward and commonly used decomposition approach initially designed to resolve multi-class classification problems, and lately explored as an ensemble technique (Oza & Tumer, 1999).

A number of modifications and combined techniques has been proposed for enhancement of these techniques. Webb in (Webb, 1999) has recently suggested Multiple Boosting, a technique for combining Boosting and Wagging (a variant of Bagging). He has shown that Multiple Boosting offers further advantage over AdaBoost (Freud & Schapire, 1996).

In (Zheng *et al.*, 1998) Boosting is combined with Stochastic Attribute Selection and it is shown that the combined model effectively increases ensemble diversity and accuracy. In (Zheng & Webb, 1999) this combination has been enhanced incorporating another sampling technique, Bagging, and got further increase of accuracy and reduction of variability. A combination of feature subspace projection and sampling techniques has also been studied in (Brieman, 2001). He enhances Bagging with random feature selection and uses Bagging for ongoing estimates of generalization error of the ensemble as well as estimates for the strength and correlation. These three approaches to combine feature subspace projection and sampling are developed for a decision tree classifier. Feature subspace projection is performed at each node of the decision tree.

Oza and Tumer in (Oza & Tumer, 1999) combined correlation-based feature subspace projection and one-per-class techniques to build an ensemble. Their Input Decimation method is based on the assumption that features highly correlated with particular class and uncorrelated between each other are important to classify instances from that particular class. This reduces correlation between errors produced by the component classifiers promoting ensemble diversity and accuracy growth.

In (Guruswami & Sahai, 1999; Windeatt & Ardeshir, 2002) the performance of ECOC in solving multi-class learning problems has been extended with the power of Boosting to eliminate the error correlation disadvantages of ECOC.

One can summarize that the previous studies on combined models were encouraging, demonstrating that ensemble techniques that use different mechanisms of ensemble generation can contribute to the performance of the final model.

The most interesting combination for the feature space heterogeneity situation is a combination of feature subspace projection and sampling. We assume that instead of the bootstrapping effect, or in addition to it, some modification of Selective Sampling (Lindenbaum *et al.*, 1999), can be applied. Selective Sampling, a form of *active learning*, reduces the cost of labeling training instances by asking only for labels of the most informative unlabeled examples. The idea behind using Selective Sampling for heterogeneous data is to find informative instances for each local region. Most Selective Sampling methods focus on choosing instances from regions of uncertainty. A possible approach to define uncertainty is to utilize component classifiers of an ensemble and select instances on with the majority of component classifiers disagree.

6 SUMMARY

Dealing with heterogeneity in the data is critical for most machine learning techniques. In this work we have elaborated the data heterogeneity concept, reviewed relevant research efforts, and studied feature space heterogeneity with application of different ensemble techniques.

Feature space heterogeneity is exhibited via presence of unstable features, which vary in their relevance to the target concept across the set of instances. Relevance of those unstable features may depend on other (control) features or on the query instance location in the feature space. When feature space heterogeneity is present, correlations between features are often substantial. Interacting features are dependent on the class and on other features as well. Most learning algorithms are not able to deal with interactive features, and it results in the decrease of accuracy. Correlations between features may be handled by feature subspace projection techniques while local areas of relevance can be covered by instance set sampling techniques.

A different approach to utilize ensemble learning for heterogeneous data is to consider an ensemble as a collection of locally good learning models and apply them selectively to classify the target instances. This is contrary to a more common view of ensemble as a collection of models large enough to correct biases of individual classifiers and/or reduce variability of the ensemble predictions.

After experimental evaluation of ensemble techniques based on feature subspace projection and sampling on synthetic heterogeneous data, we have concluded that combination of these two techniques in ensemble generation is potentially beneficial for classifying heterogeneous data.

There are two possible ways to develop the idea of feature selection based ensembles: (1) combination of feature subspace projection, sampling, and class decomposition, and (2) adaptive clustering to select locally relevant features and the corresponding instances for classifier construction. These directions will be elaborated in future research. Previous studies on combined models are encouraging demonstrating that ensemble techniques that use different mechanisms of ensemble generation can contribute to the performance of the final model.

Besides experimental studies on synthetic data sets, a pool of real data sets should be selected in order to examine their specifics. In addition, an exploratory analysis should be applied to these data sets calculating feature and class dependencies, characteristics of geometry, and some known measures of heterogeneity/contextual dependency, etc. Then the problem domain characteristics revealed should be used evaluating applicability of feature selection based ensembles.

ACKNOWLEDGEMENTS

This research has been sponsored by the Center for Discrete Mathematics and Theoretical Computer Science (DIMACS), Rutgers University, NJ, founded as a National Science Foundation and Technology Center. Authors would like to thank Prof. Robert Schapire, Princeton University, NJ, a permanent member of DIMACS, for the fruitful discussions on the topic, constructive comments and suggestions.

REFERENCES

Aha, D., Goldstone, R. Learning attribute relevance in context in instance-based learning algorithms. In: Proc. 12th Annual Conf. of the Cognitive Science Society, Lawrence Earlbaum, 141–148, 1990.

Allen, J. Expert Play in Connect-Four. 1990. Available at [http://homepages.cwi.nl/~tromp/c4.html].

Apte, C., Hong, S.J., Hosking, J., Lepre, J., Pednault, E. & Rosen, B. Decomposition of heterogeneous classification problems. In: Proc. 2nd Int. Symp. on Intelligent Data Analysis, Springer-Verlag, 17-28, 1997.

Avilcs-Cruz, C., Guerin-Duguc, A., Voz, J.L., Van Cappel, D. Enhanced Learning for Evolutive Neural Architecture (ELENA). Technical Report R3-B1-P, 1995. Universite catholique de Louvain, Neural Network Group, Belgium. Available at [http://www.dice.ucl.ac.be/neural-nets/Research/Projects/ELENA/elena.htm].

Brieman, L. Bagging Predictors. Machine Learning 24, 123-140, 1996.

Brieman, L. Random Forests. Machine Learning 45(1), 5-32, 2001.

Brieman, L., Friedman, L., Olshen, R., Stone, C. Classification and Regression Trees. Wadsworth Inc., Belmont, California, 1984.

Dietterich, T. Machine Learning Research: Four Current Directions. AI Magazine 18(4), 97-136, 1997.

Dietterich, T., Bakiri, G. Solving Multiclass Learning Problems via Error-Correcting Output Codes. Artificial Intelligence Research 2, 263-286, 1995.

Domingos, P. Context-Sensitive Feature Selection for Lazy Learners. Artificial Intelligence Review 11, 227-253, 1997.

Efron, B., Tibshirani, R. An introduction to the bootstrap. IEEE Transactions on Software Engineering, SE-13, Chapman & Hall, 222-232, 1993.

Freud, Y., Schapire, R. Experiments with a New Boosting Algorithm. In: Proc. 13th Int. Conf. on Machine Learning, 148-156, 1996.

Furnkranz, J. Pairwise Classification as an Ensemble Technique. In: Proc. 13th European Conf. on Machine Learning. Springer, 97-110, 2002.

Guruswami, V., Sahai, A. Multiclass Learning, Boosting, and Error-Correcting Codes. In: Proc. 12th Annual Workshop on Computational Learning Theory. ACM Press, 145-155, 1999.

Hall, M. Correlation-based Feature Selection for machine Learning. PhD Thesis. University of Waikato, Australia, 1999.

Harries, M., Horn, K. Learning Stable Concepts in Domains with Hidden Changes in Context. In: Proc. 13th Int. Conf. on Machine Learning (ICML 1996), Workshop on Learning in Context-Sensitive Domains. Morgan Kaufmann, 1996.

Ho, T.K. The Random Subspace Method for Constructing Decision Forests. IEEE Transactions on Pattern Analysis and Machine Intelligence 20(8), 832-844, 1998.

Ho, T.K. Nearest Neighors in Random Subspaces. In: Proc. 2nd Int. Workshop on Statistical Techniques in Pattern Recognition. 640-648, 1998.

Kira, K., Rendell, L. A Practical Approach to Feature Selection. In: D. Sleeman and P. Edwards (Eds.) Proc. 9th Int. Conf. on Machine Learning, Morgan Kaufmann, 249-256, 1992.

Kleinberg, E. Stochastic Discrimination. Annals of Mathematics and Artificial Intelligence, 207-239, 1990.

Lazarevic, A., Obradovic, Z., Fiez, T. Adaptive Boosting for Spatial Functions with Unstable Driving Attributes. In: Proc. Pacific-Asia Conf. on Knowledge Discovery and Data Mining, Computer Science Editorial 3, Springer-Verlag, 329-340, 2000.

Lindenbaum, M., Markovitch, S., Rusakov, D. Selective Sampling for Nearest Neighbor Classifiers. In: Proc. 16th National Conf. on Artificial Intelligence and 11th Conf. on Innovative Applications of Artificial Intelligence, AAAI Press / The MIT Press, 366-371, 1999.

Maclin, R., Opitz, D. An Empirical Evaluation of Bagging and Boosting. In: Proc. 14th National Conf. on Artificial Intelligence, 1997.

Masulli, F., Valentini, G. Comparing Decomposition Methods for Classification. In: Proc. 4th Int. Conf. on Knowledge Based Intelligent Engineering Systems and Allied Technologies. IEEE Press, 788-791, 2000.

Opitz, D., Maclin, R. Popular Ensemble Methods: An Empirical Study. Artificial Intelligent Research 11, 169-198, 1999.

Oza, N., Tumer, K. Dimensionality Reduction Through Classifier Ensembles. Computational Sciences Division, NASA Ames Research Center Technical Report NASA-ARC-IC-1999-126. 1999.

Prodromidis, A., Stolfo, S. Pruning classifiers in a distributed meta-learning system. Technical Report CUCS-011-98. Department of Computer Science, Columbia University, New York, NY,1998.

Puuronen S., Tsymbal A., Skrypnyk I. Correlation-based and Contextual Merit -based Ensemble Feature Selection. In: Proc. 4th Int. Conf. on Intelligent Data Analysis. Springer, 135-144, 2001.

Quinlan, J.R. Induction of Decision Trees. Machine Learning 1, 81-106, 1986.

Robnik-Sikonja, M., Kononenko, I. Attribute Dependencies, Understandability and Split Selection in Tree Based Models. In: I. Bratko, S. Dzeroski (Eds.) Proc. 16th Int. Conf. on Machine Learning (ICML 1999), Morgan Kaufmann, 344-353, 1999.

Robnik-Sikonja, M., Kononenko, I. Context-Sensitive Attribute Estimation in Regression. In: Proc. 13th Int. Conf. on Machine Learning (ICML 1996), Workshop on Learning in Context-Sensitive Domains. Morgan Kaufmann, 1996.

Shapire, R., Freud, Y., Bartlett, P., Lee, W.S. Boosting the Margin: A New Explanation of the Effectiveness of the Voting Methods. The Annals of Statistics, 25(5), 1651-1686, 1998.

Skurichina, M., Duin, R. Bagging and the Random Subspace Method for Redundant Feature Spaces. Multiple Classifier Systems 2001, LNCS 2096, 1-10, 2001.

Skurichina, M., Kuncheva, L., Duin, R. Bagging and Boosting for the Nearest Mean Classifier: Effects of Sample Size on Diversity and Accuracy. Multiple Classifier Systems, Proc. 3rd Int. Workshop MCS 2002, F. Roli, J. Kittler (eds.), LNCS 2364, 62-71, 2002.

Toussaint, G. Proximity graphs for nearest neighbor decision rules: recent progress. In: Proc. 34th Symposium on Computing and Statistics (Interface-2002), 2002.

Tumer, K., Ghosh, J. Error Correlation and Error Reduction in Ensemble Classifiers. Connection Science 8(3-4), 385-404, 1996.

Turney, P. Robust Classification with Context-Sensitive Features. In: Proc. 6th Int. Conf.

on Industrial and Engineering Applications of Artificial Intelligence and Expert Systems, 268-276, 1993.

Turney, P. The Identification of Context-Sensitive Features: A Formal Definition of Context for Concept Learning. In: Proc. 13th Int. Conf. on Machine Learning (ICML-96), Workshop on Learning in Context-Sensitive Domains. Morgan Kaufmann, 53-59, 1996.

Webb, G. MultiBoosting: A Technique for Combining Boosting and Wagging. In: Machine Learning 40(3), Kluwer Academic Publishers, 159-196, 2000.

Webb, G. MultiBoosting: A Technique for Combining Boosting and Wagging. 1999. Available at [http://www3.cm.deakin.edu.au/~webb/publications.htm].

Windeatt, T., Ardeshir, G. Boosted Tree Ensembles for solving Multiclass Problems. Multiple Classifier Systems, Proc. 3rd Int. Workshop MCS 2002, F. Roli, J. Kittler (eds.), LNCS 2364, 42-51, 2002.

Zheng, Z., Webb, G. Multiple Boosting: A Combination of Boosting and Bagging. In: Proc. 4th Int. Conf. on Parallel and Distributed Processing Techniques and Applications, CSREA Press, 1133-1140, 1998.

Zheng, Z., Webb, G. Stochastic Attribute Selection Committees with Multiple Boosting: Learning More Accurate and More Stable Classifier Committees. In: Proc. 3rd Pacific-Asia Conf. on Knowledge Discovery and Data Mining (PAKDD-99), Springer-Verlag, 123-132, 1999.

Zheng, Z., Webb, G., Ting, K.M. Integrating Boosting and Stochastic Attribute Selection Committees for Further Improving the Performance of Decision Tree Learning. In: Proc. 10th IEEE Int. Conf. on Tools with Artificial Intelligence (ICTAI-98), IEEE Computer Society Press, 216-223, 1998.