Global Journal of Pure and Applied Mathematics.
ISSN 0973-1768 Volume 12, Number 2 (2016), pp. 1547–1565
© Research India Publications
http://www.ripublication.com/gjpam.htm

Performance of non-parametric classifiers on highly skewed data

Fatima Siddiqui and Qazi M. Ali
Department of Statistics & O.R.
A.M. University, Aligarh 202002, India.

Abstract

A wide range of non-parametric classifiers have been suggested and developed in the recent years in order to overcome the compulsion of using the classical parametric Maximum Likelihood Classifier (MLC) for non-normal data. The most advanced of these classifiers being the ones based on the Artificial Neural Network (ANN) algorithm, the Support Vector Machines (SVMs) and the Random Forests (RFs). Although a number of researches have established the efficiency of these distribution-free classifiers over the MLC, nothing much has been contributed in to compare the performance of these non-parametric classifiers against each other. With the objective of filling this gap, this study conducts an empirical study to compare the performances of these three machine learning classification algorithms while classifying asymmetric data. RF classifier was found to be best performing among the three classifiers and robust enough to even very high levels of skewness.

AMS subject classification:
Keywords: Random forests, Support vector machines, Artificial neural networks, skewed data.

1. Introduction

Pattern recognition or specifically classification is the problem of allocating an unknown object based on a set of features in to one of the several possible classes (or populations). These features can be thought of as p-dimensional vectors of measurements describing the object. Classifiers can be broadly classified into parametric and non parametric classifiers depending upon whether any distributional assumptions are imposed on the underlying classes or not.

Parametric classification techniques are most frequently used for pattern recognition due to their easy interpretation, lesser number of underlying parameters and the fact that no efforts are required to train them. But their performance is found to be highly affected by the violation of normality assumption for the underlying populations. In fact the real datasets are generally found to be asymmetric or specifically skewed in nature. For example in complex land cover classification problem, if area under crop is one of the several land use categories then the presence of trees along with crops as is the case in agro forestry or the presence of stressed crops will result in the skewed spectral feature distribution of the crop class as healthy crops, stressed crops and trees will have different spectral signatures. This limitation of distributional assumption poses a threat to the efficient performance of the parametric classifiers when the data is non-normal in nature. Hence, in the presence of such non-optimal situations for the conventional MLC, the researchers and experts suggest to look out for the alternative non-parametric classifiers which are free of any distributional assumptions and hence are expected to perform well with a variety of distributions as long as the class signatures are reasonably distinct.

Among the non-parametric classifiers available, parallelepiped and minimum distance classifiers fall under the statistical classifiers category. Parallelepiped classifiers are the simplest ones of all the non parametric classifiers and require minimal information in the form of minimum and maximum values of all the feature in each of the classes which define the boundaries of the parallelepipeds and each observation is then checked if it lies in any of the defined parallelepipeds. This classifier is highly affected by the presence of overlapping parallelepipeds and inability of locating a new observation in any of the defined parallelepipeds and hence is not considered a robust choice for most of the classification problems. The second one i.e. the Minimum distance classifier calculates the distance between an observation and the centroids of the different training classes using the Mahalanobis distance measure and accordingly decides to allocate the observation to the class which is nearer to the observation in terms of lower value of the distance measure. This classifier is also found to be mathematically fast and does not include any complex underlying mathematical concepts but its performance has always been found to be inferior to the more robust parametric maximum likelihood classifier (MLC) (Benediktsson et al. (1990)). Moreover the performance of both of the above discussed classifiers is expected to be affected a lot by the presence of heterogeneity and outliers in the data classes.

Thus, keeping in mind the limitations of these statistical non parametric classifiers we turn our attention to the more advanced machine learning algorithms for classifying skewed datasets. Among the class of non-parametric machine learning algorithms, artificial neural networks, support vector machines and random forests classifiers have gained considerable popularity among the researchers and the analysts in the field of remote sensing, voice recognition, text classification, medical diagnosis of terminal diseases etc. After a comprehensive review of the literature of classification techniques in Section 2, we found that although a number of studies have been conducted for comparing the performances of the parametric MLC with its non-parametric counterparts for particular case based studies, nothing much has been said about the performance of the non-parametric classifiers when the data is severely skewed. Hence, in the present study we attempt to fill this gap by particularly focussing on the performance of the three machine learning algorithms for classifying severley skewed data. The present manuscript is divided in to the following section, Section 2 highlights other comparative works carried out in past with the machine learning algorithms taken up for study in the present chapter, Section 3 gives a brief discussion of the methods and the classifiers used for comparison in the present work, a detailed investigation on the comparative performances of the non-parametric classifiers for real and skewed simulated datais given in Section 4 and the results and conclusions of the study have been discussed in Section 5.

2. Background

Classification procedures are widely used in a variety of field due to which a large number of studies comparing the performance of different types of classifiers have been produced. Hence for the sake of comprehensiveness and better understanding we give an account of some of the recent comparative works with respect to the fields in which they were conducted.

In remote sensing, Huang and Davis Huang *et al.* (2002) compared the performance of SVM with MLC, ANN and decision tree classifiers for the classification of a six band Thematic Mapper (TM) image and found SVM to be competitive enough with the other two methods. Erbek *et al.* (2004) compared the performance of MLC with multilayer perceptron (MLP) and Linear Vector Quantization (LVQ) ANN classifiers for classifying a Landsat TM data. Kavzoglu & Kolkesen (2009) assessed the effect of kernel choice on the SVM classifiers and concluded that SVM classifiers based on rbf kernels outperform the MLC for the classification of landcover images. Otukei & Blaschke (2010) found decision tree classifiers to be performing better in general in terms of the classification accuracies than the SVM and the MLC classifiers for classifying Landsat TM datasets. Apart from these, Lu *et al.* (2004); Olthof *et al.* (2004); Pal & Mather (2004) are some other comparative works conducted for specific case based classification problem.

In bioinformatics and diagnostics, Dudoit *et al.* (2002) employed three microarray datasets for the classification of tumours and interestingly found DLDA and ANN classifiers to be performing remarkably well as compared to more sophisticated aggregated or bagged decision tree classifiers. Diaz-Uriarte & de Andres (2006) investigated the performance of Random Forest, Diagonal linear discriminat analysis (DLDA) technique, KNN and SVM classifiers for classifying microarray datasets. Statnikov *et al.* (2008) in his study on the microarray based cancer found SVM classifiers to be performing better than the RF classifiers with and without adopting any feature selection procedures. Khondoker *et al.* (2013) conducted an extensive simulation study to compare the performance of LDF, SVM, ANN and RF under various settings of the data characteristics. They concluded that different classifiers performed optimally under different settings. Apart from these works, other significant comparative assessments of the classifiers for microarray data classification for cancer diagnosis can be found in Huang *et al.* (2005); Lee *et al.* (2005); Pirooznia *et al.* (2008); Rocke *et al.* (2009), Yousefi *et al.* (2011) etc.

Apart from these two fields a number of comparative studies in the field of text recognition, speech recognition, ecology (Cutler *et al.* (2007)) and financial data prediction (Zhang *et al.* (1999)) have been produced.

Acknowledging the case based nature of all the above discussed comparative studies, we felt the need of conducting a comprehensive simulation based comparative assessment on the non-parametric classifiers. Apart from the conclusions based on the simulated datasets, we also illustrated the performances of these classifiers on some benchmark real datasets.

3. Non-parametric classifiers used

3.1. Artificial Neural Networks (ANNs)

Artificial Neural networks comprise of a set of machine learning algorithms which use artificial intelligence techniques for complex problem solving. ANNs have evolved over the years as a robust pattern recognition alternative to other methods with contribution from varied disciplines ranging from neuroengineering, financial data prediction, quality control, modeling and prediction to pattern recognition. Detailed conceptual explanation of ANNs can be found in Haykin (1999). ANN classifiers enjoy pretty attractive advantages over other classifiers of being data driven self adaptive or distribution free methods capable of estimating posterior probabilities (Richard & Lippmann (1991)) and handling multi-source data efficiently (Benediktsson *et al.* (1990)). Additionally, they are hailed as universal approximators and non linear models. In order to make them perform efficiently ANNs should be trained with proper choice of network architecture and optimal parameters in the form of number of nodes and the number of hidden layers used for training the network.

Supervised classification in an ANN classifier is administered through exposure to a known set of input and corresponding output data i.e training data. The training algorithm used trains the network by adjusting the interconnection weights between the neurons through an iterative procedure such that the overall error is minimized and then this trained network is used to determine the classification of unknown set of data. Multilayer perceptron with back-error propagation neural network considered here for classification is the most widely used supervised ANN architecture design (Tso & Mather (2009)).

A Multilayer perceptron (MLP) network can consist of three basic types of layers, first is the input layer, whose nodes take the elements of the external feature vector as inputs, the second type of layer is the hidden layer (which can be more than one) and the third is the output layer in which the number of nodes is equal to the number of classes in the classification problem. These three types of layers are completely connected to each other with weighted interconnections between the processing elements as shown in Figure 1. The value held by each node is called its activity (a_i).

For training or learning the interconnection weights (w_i) between the layers and the activities of the nodes the back-propagation algorithm is used which consists of forward

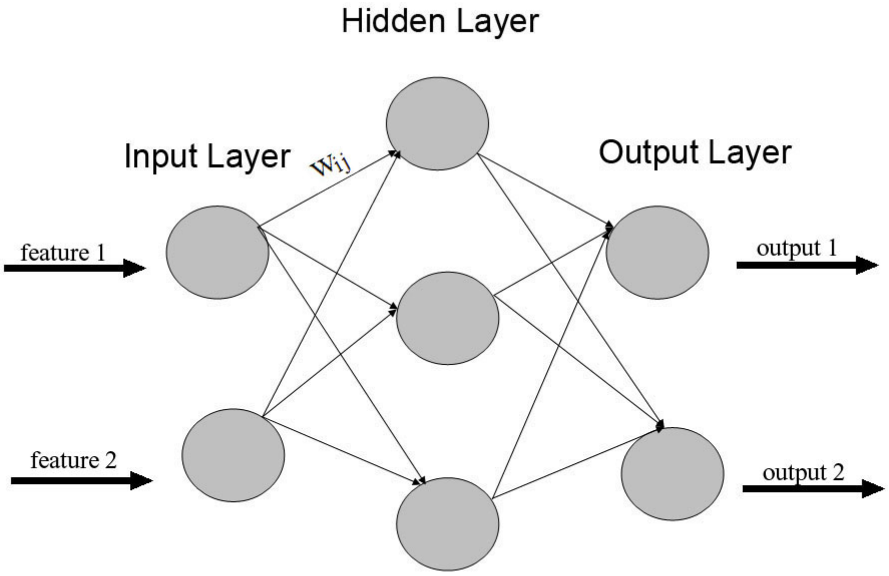


Figure 1: Basic architecture of a multilayer perceptron network.

as well as backward propagation. During forward propagation input signals are supplied to the network through the input layer and the updated activities of the nodes using the interconnection weights, are passed on from layer to layer starting from the input layer to the output layer. Formally the input that a single node say j receives is calculated as the weighted sum of the activities of the sending nodes in the preceeding layer defined as,

$$x_j = \sum_i a_i w_{ji} \tag{1}$$

where, a_i is the activity of the ith node and w_{ji} is the weight of the connection from the ith node to the jth node. And the output from each node to the nodes in the next consecutive layer is calculated by converting the input in equation (1) using a mapping function, sigmoid mapping function being the common choice. This transfer of updated signal continues from one layer to another until the output layer is reached. After which the error between the network output and the desired output is computed using the least squared error criterion. This error is then back-propagated through the network and the interconnection weights (w_{ji}) are updated according to the generalized delta rule described in Rumelhart *et al.* (1986).

This process of forward propagation of signals and back propagation of errors is repeated for training samples until the error is minimized or reaches the desired threshold. In the present work we used MATLAB's *neural network toolbox* for training artificial neural networks for simulated as well as real datasets.

3.2. Support Vector Machines

Support vector machines (SVMs) form a group of one of the most recent and theoretically robust machine learning algorithms (Vapnik & Vapnik (1998)). SVMs are aimed at locating an optimal separating hyper-plane between the two data classes in the multidimensional feature space using some optimization algorithms. Under supervised learning, SVMs use training datasets to locate optimal boundaries or hyper-planes between classes and the unseen test datasets are used to verify their generalizing ability of minimizing the confusion between classes with these optimal boundaries (Mountrakis *et al.* (2011)). SVMs which are binary classifiers can be applied to multiclass classsification problems using one-against-one (Knerr *et al.* (1990)) and one-against-others (Vapnik & Vapnik (1998)) techniques. For a classification problem involving two p-dimensional data classes, there may be $p - 1$ separating hyper-planes but SVMs aim at finding that single optimal hyperplane which minimizes the structural risk by maximizing the distance between the plane and the closest data instances lying on either side of the plane.

Depending upon the type of separability between the training data classes, SVM algorithms can be divided in to two categories. The first one corresponds to the theoretically lesser complex form of SVM and is used when the training data classes are linearly separable and the other one based on non-linear kernel functions comes in to the picture when the data is found to be linearly inseparable. The simplest way of training an SVM is by using linearly separating cases. If we assume p-dimensional linearly separable training datasets represented as $\{\mathbf{x}_i, y_i\}, i = 1, \dots, n, y_i \in \{1, -1\}, \mathbf{x}_i \in \mathbf{R}^p$, where $\mathbf{x}_i$ represents the p-dimensional set of training vectors and y_i represent the labels of the corresponding classes which is coded as $+1$ for class 1 and -1 for class 2, then the optimum separating hyperplane between the two classes in binary classification problem is found in terms of two parallel separating hyperplanes, one for each class, defined as

$$\mathbf{w}^T\mathbf{x}_i + b \geq +1, \quad \forall \ y_i = +1 \tag{2}$$

$$\mathbf{w}^T\mathbf{x}_i + b \leq -1, \quad \forall \ y_i = -1 \tag{3}$$

Where, w is a vector perpendicular to the linear hyperplane and b is the bias representing the offset of the discriminating hyper-plane from the origin. The training points which lie on these two separating parallel hyper-planes are called *support vectors* (Mathur & Foody (2008)) and have a key role in the establishment of the optimal hyper-plane as they constrain the margin between the training data instances of a class and the separating hyper-plane.

Mostly the real data encountered in various classification fields is much more complex in nature and is usually found to be linearly inseparable. In such cases of linear inseparability between the classes, a non-linear mapping function say Φ is used to map the original training data classes in to higher dimensional feature space where they can be linearly separated. And linear optimal hyper-plane is then fitted between the classes in the new higher dimensional transformed feature space. An appropriately chosen transformed feature space of sufficient dimensionality is found to be capable of discriminating between the data classes (Kotsiantis (2007)). The linear optimal hyper-plane in the transformed space corresponds to the non-linear one in the original feature space. Vladimir

& Vapnik (1995) proposes the computationally efficient *kernel function* approach to map the input data in to the transformed feature space. A kernel function is denoted as $K(\mathbf{x}, y)$ such that $K(\mathbf{x}, y) = \Phi(\mathbf{x}) \times \Phi(y)$ so that the classification decision function for non-linear SVM is defined as

$$f(x) = sign\Big(\sum_{i}^{sv} \alpha_i y_i K(\mathbf{x}, \mathbf{x}_i\}) + b\Big) \tag{4}$$

Studies suggest that out of the three major types of kernel functions used for training SVM classifiers, i.e. the polynomial kernel function, the radial basis function (rbf) and the sigmoid kernel function, the sigmoid kernel usually does not perform ideally for classification problems. Whereas the performance of polynomial kernels and the *rbf* kernel is found to be comparable with *gaussian rbf* kernel usually being the preferable choice (Tso & Mather (2009)). Hence, in the present study *gaussian rbf* has been used for training the SVM classifier and its parameters are learned using the gradient search method. The *gaussian rbf* kernel is defined as,

$$K(\mathbf{x}_i, \mathbf{x}_j) = exp\big(- \parallel \mathbf{x}_i - \mathbf{x}_j \parallel^2 /2\sigma^2\big) \tag{5}$$

where $\mathbf{x}_i, \mathbf{x}_j$ are the feature vectors, σ is the so called free parameter which along with the error penalty parameter C need to be fixed by the user.

The major advantages of SVM classifiers over others is their ability to minimize the misclassification rates for unseen samples, structural risk minimization (SRM) concept based training which always finds a global minimum (Tso & Mather (2009)), higher generalization capabilities as compared to ANNs and lesser efforts required for training the model parameters (Joachims (1998)) and fair performance even with the scarcity of training data. But the extent of success of SVMs in discriminating between the classes depends largely upon how well they are trained in terms of the method used to generate SVM model, choice of kernel parameters and the choice of parameters for the chosen kernel as well (Huang *et al.* (2002)). Also SVM based classifiers are asupposed to be sensitive to the outliers (Shao & S. (2012)).

3.3. Random Forests

In the recent years ensemble learning that generates many classifiers and aggregate their results for making final decisions has gained a lot of research interest as they give better classification accuracies theoretically as well as empirically than an individual classifier. The ensemble of classifiers is generated using re-sampling techniques. Bagging (Breiman (1996)) and boosting are the two well-known re-sampling techniques that are often used to generate the ensembles. In boosting, successive trees give extra weights to points incorrectly predicted by earlier predictors and in the end a weighted vote is taken for prediction. Bagging is a re-sampling technique which works on the concept of aggregated bootstrap samples. In bagging of decision trees, successive m independent fully grown trees are generated using N bootstrap samples of the training dataset of size say N, each of the m fully grown trees without pruning cast a vote in favour of one of the possible k classes and in the end a simple majority vote decides the final prediction of the input feature vector.

Random forest (RF) classifiers originally developed by Breiman (Breiman (2001)) correspond to the relatively latest classification algorithms which attracted wide scale interests of the researchers in a relatively smaller duration since their development. Random forest algorithms belong to the class of *ensemble learning* algorithms which have been shown to be effectively useful in although not numerous due to its most recent discovery but still in a considerable number of significant researches. As its name suggests, an RF classifier's architecture is based on the concept of generating a forest or an ensemble of a large number of bagged classification trees which are grown on random subset of input vectors and splitting nodes on a random subset of features (Prinzie & den Poel (2008)). The main difference between the construction of trees in RFs and in bagging of trees is that in bagging each node is split using the best split among all variables, while in a random forest, each node is split using the best variable among a subset of predictors randomly chosen at that node. This strategy increases the randomness in bagging the trees and hence turns out to perform pretty well as compared to other advanced machine learning algorithms like ANNs and SVMs (Breiman (2001)).

The RF classifiers possess various attractive advantages over other classifiers. They do not need extensive parameter training like SVM and ANN and are required to be provided with only two parameter values i.e. the number of trees to be grown and the number of predictors to be considered for best split at each node. Moreover, the parameters do not need much fine-tuning and often the default parameter values give desirable results. Out-of-bag samples at each boostrapping step can be used to calculate an unbiased error rate and variable importance which eliminates the need for a separate test set for cross validation (Breiman (2001)). It performs embedded feature selection and is found to be relatively insensitive to large number of irrelevant features, and hence spares the user of some pre-processing load of feature selection. Classification by random forest tehniques results in very limited generalization error due to the construction of a large number of trees and hence leaves no or very little scope for overfitting. In contrast to all these appealing advantages, RFs do not have many disadvantages except that it unables the examination of individual trees separately and are relatively slow as compared to SVMs due to the construction of a large number of trees. Apart from all the above discussed advantages, RFs are found to be relatively more robust to outliers and noise and this characteristic of RF classifiers might prove to be beneficial for the classification of highly skewed datasets.

4. Numerical Experiments and Results

4.1. Simulation and data generation

With a purpose to zero in the optimal non-parametric machine learning algorithm in terms of the lesser misclassification error rates for classification of skewed data, an extensive simulation study has been carried out in this section with a variety of simulation settings generating highly skewed datasets. The study in the present manuscript has been limited to the study of bi-variate two-group classification problem. Training as well as the test datasets are generated for diverse combinations of various data characteristics such as variability, training data size, separability between the groups, feature set size etc., which can affect a classifier's performance apart from the skewness of the data. The first population was simulated from standard bivariate normal distribution and was kept fixed over all replications while several configurations of the population parameters were considered for simulating the other bivaraite normal population. An account of the configurations of factors which were considered for simulations is given in Table 1. After simulating the normal populations using the parameter values given in Table 1, the transformations in equation (6) were used to generate higly skewed data from the simulated bivariate normal data. Each of the three classification algorithms namely ANN, SVM and RF was trained using the simulated training datasets. A separately generated skewed index sample of size 1000 was classified by the trained classifier and the resulting misclassification error rates were calculated. This process of training and validating a classifier was repeated over 30 replications for training and index datasets simulated for each of the factor combinations given in Table 1 and the observed misclassification error rates were averaged over all the replications to get an unbiased estimate of the the actual error (AE) rates and the apparent error (APE) rates (Clarke *et al.* (1979)). Averaged kappa measures for each of the index sample were also calculated and compared. This computation was repeated for each of the 3 classifiers under investigation in this study and the results are summarized in Table 3. If $\mathbf{X_i} \sim \mathbf{N_p}\,(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ then transformations for generating multivariate skewed data $\mathbf{Y_i}$ are given as

$$\mathbf{Y_i} = \exp\left(\mathbf{X_i}/\delta\right) \tag{6}$$

where δ is used to generate high levels of skewness. The classification process was carried out in *MATLAB*. The ANN classifier was trained with back propagated multilayer perceptron algorithm for different settings of the hidden layer sizes and was found to be performing the best for a value of 15. The non-linear SVM classifier was trained with *gaussian rbf kernel* and the values of the parameters were fixed using grid search method. The number of optimal trees for RF was determined hueristically and it was fixed at 500 over all the simulations. The skewness of the second population was measured by Mardia's multivariate coefficient (S_k) (Mardia (1970)) of skewness and is reported in table 3.

4.2. Real datasets used for comparison

We also evaluated and compared the performance of ANN, SVM and RF classifiers for classifying positively skewed data on some benchmark real life datasets. An account of them is given below.

Dataset 1 is the *Landsat database* (Bache & Lichman (2013)) which consists of 6435 instances on 6 landuse classes namely the red soil, cotton crop, grey soil, damp grey soil with vegetation stubble and very damp grey soil which are present in a (tiny) sub-area of a scene captured by Landsat satellite. Each of the 6435 rows of the data corresponds

Table 1: Factor combinations for simulation from second population.

Mean Vector of second population	$\boldsymbol{\mu_2} = \left(a^1, 0\right)$ $a = (0, 2, 4)$
Covariance Matrix of second population	$\boldsymbol{\Sigma_2} = \sigma^2 I$ $\sigma^2 = (1.5, 3, 8)$
Skewness Parameter	$\delta = .5$
Size of Training sample from each class	$n = (25, 50, 100, 400, 600, 1000)$

to a (3 × 3) square neighbourhood of pixels completely contained within the (82 × 100) sub-area and a number indicating the classification label of the central pixel. Hence, we have used only the central pixels of each of the (3 × 3) neighbourhood of pixels ignoring the other pixels. It implies that each row contains the pixel values in the four spectral bands on 9 pixels. After considering only the central pixels for classification the sizes of the training and the test datasets reduce to 4435 × 4 and (2000 × 4) respectively where rows correspond to each of the 6435 pixels and columns correspond to their spectral values in the four spectral bands. All the 6 classes in this dataset were found to be significantly skewed with values of Mardia's multivariate coefficient of skewness at 2.52, 6.12, 2.12, 1.414, 4 and 2.046 respectively.

Dataset 2 is the *New Thyroid Dataset* (Bache & Lichman (2013)) which is a (215 × 5) data array containing the measurements of 5 attributes (which are 5 Lab tests) on each of the 215 patients in order to predict a patient's thyroid state as normal, hypothyroidism or hyperthyroidism. On the basis of the lab tests, out of 215 instances in the dataset 150 of them were found to be in the normal thyroid range, 35 in the hypothyroid and 30 in the range of hyperthyroidism. All 3 of the classes in the dataset tested positive for significant multivariate skewness with the coefficient of multivariate skewness values 5.14, 6.69 and 11.753 respectively.

Dataset 3 is the *Indian Liver Patient Database (ILPD)* (Bache & Lichman (2013)) which is a (583 × 10) array containing a total of 583 patient records on 10 attributes. Out of 583 cases, 416 are attributed to the liver patient category and the remaining 167 to normal liver functioning patients category. For this dataset too the multivariate skewness coefficient for the two classes were found to be significant at values 543.38 and 97.96 respectively.

Dataset 1 has already been given with separate training and test dataset at Bache & Lichman (2013). Hence the actual error rates (AER) for this dataset was evaluated by training the classifier using the training dataset and validating it with the separate test dataset. While the actual misclassification error rates for datasets 3 and 4 were obtained using Lachenbruch's leave one out method (Lachenbruch (1975)) of cross-validation. ANN, SVM and RF classifiers were used to classify each of the three real datasets independently and the APER, AER and learning times of each of the three classifiers were calculated and are reported in Table 3.

4.3. Results

4.3.1 Results on simulated data

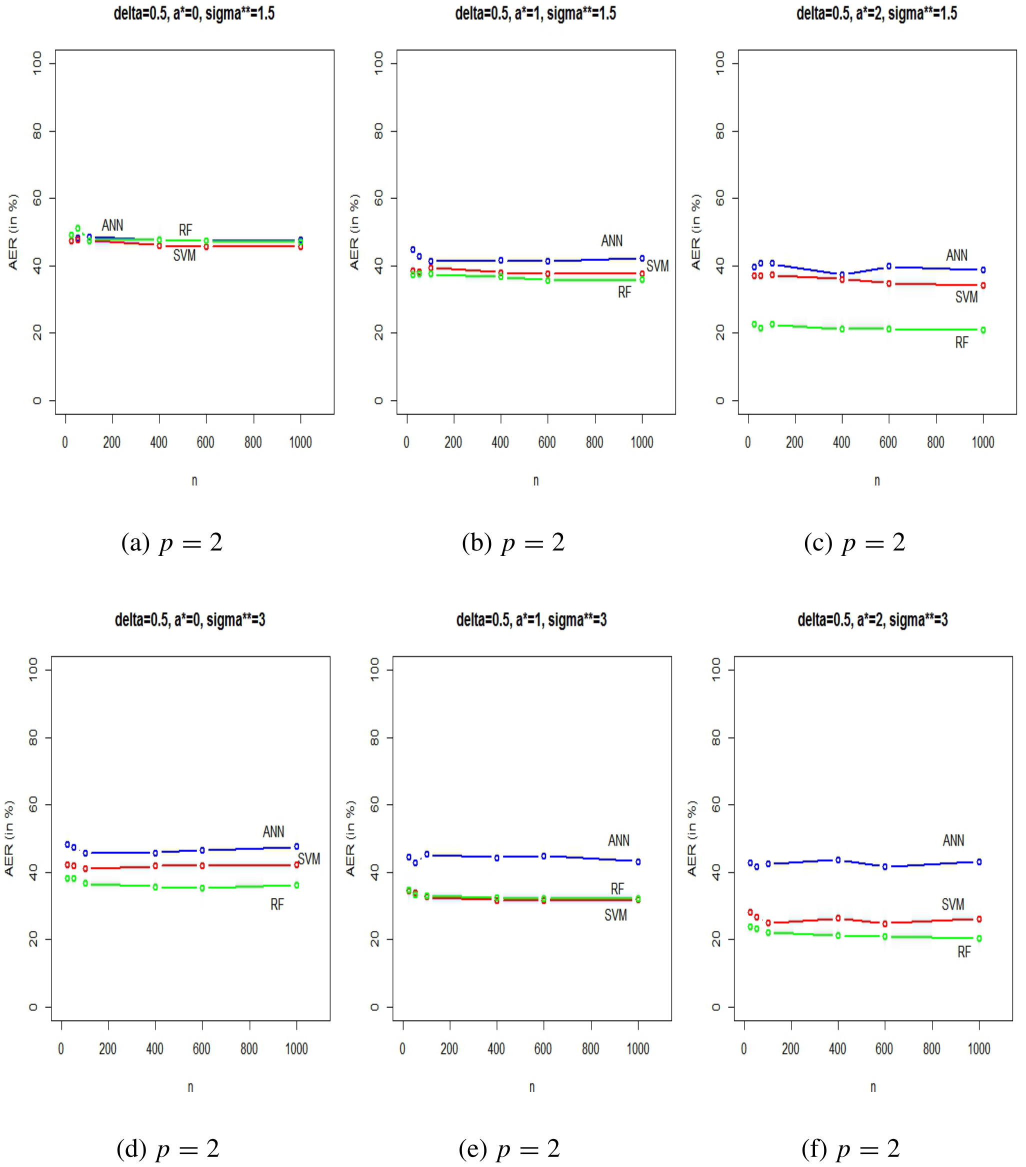
An extensive simulation study was performed in this chapter with an objective to compare the classification performance in terms of misclassification error rates of the non parametric classifiers based on ANN, SVM and Random forest techniques while handling positively skewed datasets. The Actual error rates (AERs) of the simulated index samples and the Apparent error rates (APERs) over the 30 replications of the simulated training datasets for the three classifiers ANN, SVM and RF are tabulated in Table 3. And the plots of the AER against the training sample size and the variability in data are shown in figures 2 and 3. Also the following findings were observed for various levels of the data characteristics.

- Among the three classifiers the RF classifier was found to be the best performer for all the simulated datasets which vary over a number of data characteristics except for the data simulated with ($a = 0, \sigma = 1.5$) i.e. when the means of the two populations were same and the variability of the second population (which affects the skewness of the data) was less where SVM outperformed RF by a small margin. While the ANN classifier's performance was found to be worst in terms of the misclassification error rates produced by the three classifiers.

- *Effect of training sample size:* It is evident from Table 3 that the misclassification error rates decreased with an increase in the training sample size for all the three classifiers under study in general. It can be observed from the plots in figure 2 that for RF classifiers the error rates continuously decrease as the training sample sizes are increased from 25 to 50. SVMs depict same trends for moderately skewed datsets i.e. for ($\sigma = 1.5$) but rather showcased the tendency of producing larger error rates for larger sample sizes as the variability of the datasets increase with σ. It can be observed from these plots that for all the three classifiers the most considerable decrease in the error rates was observed as the sample sizes are increased from 25 to 50 and a very small improvement afterwards. Hence, we have plotted the error rates against variability only for ($n = 25$) and ($n = 100$).

- Effect of variability: As the variability of the second population was increased by increasing values of σ, the within class skewness increased and the performance of ANN and SVM was found to be deteriorating while that of RF was found to be improving as shown in Figure 3. This implies that RF classifiers are highly resistant to the variability or the hetrogeneity of the classes.

- *Kappa measures:* The values of average kappa coefficients for SVM were observed to be lying in the range (0.5, 0.7) implying fair to moderate levels of agreement with kappa measure improving over the separabiltiy between the two classes. For RF classifiers the average values of the kappa measure lied in the range (.5, .8) which reports a fair to good level of agreement. The kappa measure for RF classifiers improved with the increasing separability between the two classes as well as with



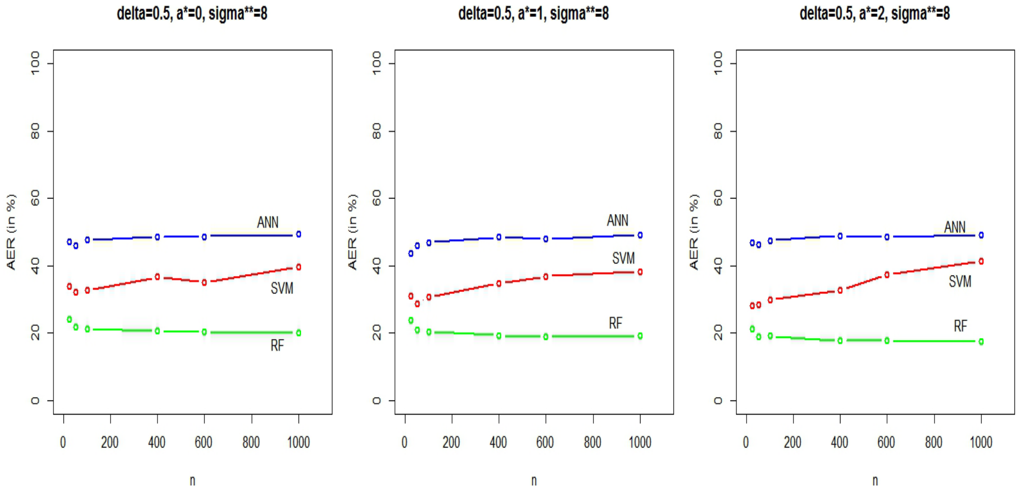
(a) $p = 2$ (b) $p = 2$ (c) $p = 2$

(d) $p = 2$ (e) $p = 2$ (f) $p = 2$

Figure 2: Plots of expected actual error rates of SVM, RF and ANN over simulated index sample for $\delta = .5$ depicting the effect of training sample size on error rates.

the increased skewness of the datasets. The level of agreement for ANN classifiers was not found to be improving at all with a constant value of average kappa measure at .5.

It was observed from the results of the simulation study that the RF classifier performed fairly better than the classifiers based on SVM and ANN for heavily skewed simulated data over all the data characteristics that were considered in this study. Although RF classifier performed comparably well for the skewed datasets for all the



(a) $p = 2$ (b) $p = 2$ (c) $p = 2$

Figure 2: Continued.

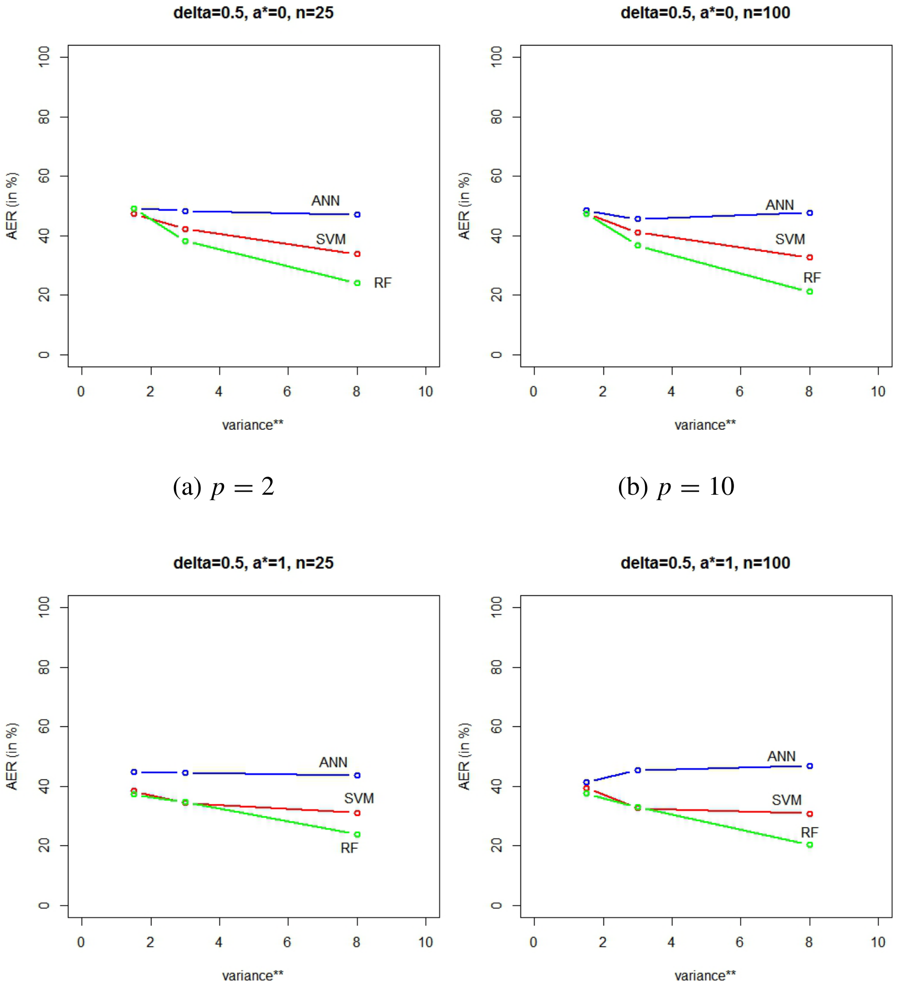
combinations of the different levels of various data characteristics but its tendency of overfitting the training data and the very large amount of computational time it takes as compared to SVM makes it a not so attractive and feasible option for classification of very large datasets.

4.3.2 Results on real datasets

Athough the real datasets considered in the present study were found significantly skewed by Mardia's test for some of the classes but none of the classes in any of the dataset was found to be highly skewed which is the main assumption of the study carried out in this study. Misclassification errors of ANN (MLP-BP), SVM (with gaussian rbf kernel) and RF classification algorithm for the three datasets are reported in Table 2. The values reported in bracket with the AER denote the optimal parameter values which were used for training each of the three classifiers. For ANN the parameter is the size of the hidden layer, for SVM its the kernel parameter and for RF it is the number of trees used for generating the forest. While ANN and RF were found to be performing quite fairly for all the datasets SVM reported maximum classification errors among the three classifier for all the datasets.

5. Conclusion

This work acknowledged the lack of a comprehensive comparative study between the three most widely used machine learning algorithms in the literature of classification algorithms specifically for classifying highly skwed data. Hence, in the present article

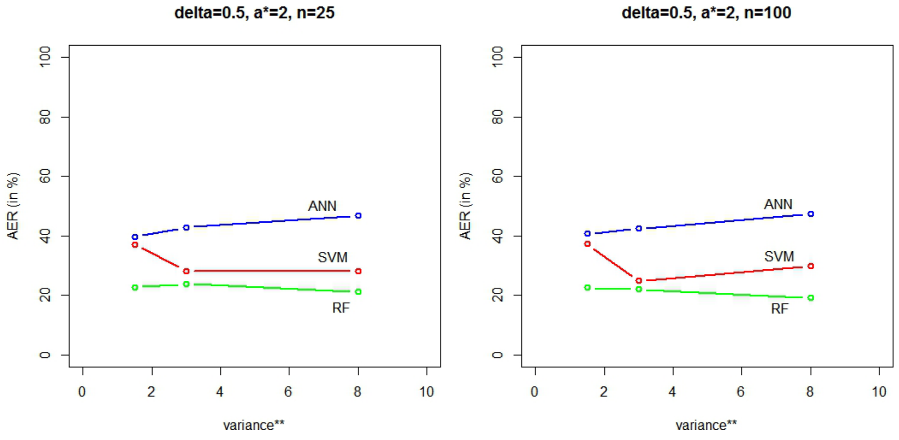


(a) $p = 2$ (b) $p = 10$

(c) $p = 2$ (d) $p = 10$

Figure 3: Plots of expected actual error rates of SVM, RF and ANN over simulated index sample for $n = (25, 100)$ and $\delta = .5$ depicting the effect of variability on error rates.

an attempt was made using the simulated datasets to select the most robust non-parametric alternative to the maximum likelihood classifier from a group of three most advanced non-parametric classification algorithms which are support vector machines, artificial neural networks and the random forest for classifying highly skewed datasets. Results of the investigations carried out on simulated data provide empirical evidences that the random forest algorithm is highly robust even to the very large levels of positive skewness in the datasets. In the light of other advantages discussed in this article such as lesser learning effort, that random forest classifiers enjoy over its counterparts support vector machines and artificial neural network classifiers, we conclude that random forest classifiers should be preferred over the SVM and ANN while dealing with severely positively skewed data.



(a) $p = 2$ (b) $p = 10$

Figure 3: Continued.

Table 2: Apparent error rate (APER) and Actual error rate (AER) of ANN, SVM and RF with their respective training parameter values for the real datasets.

	ANN		*SVM*		*RF*	
	APE	AE	APE	AER	APE	AE
Dataset 1	12.41	13.81 (15)	6.95	16.40 (1)	4.13	16.35 (100)
Dataset 2	.10	.47 (15)	3.23	9.77 (2)	0	4.65 (50)
Dataset 3	25.77	29.91 (15)	29.71	37.82 (1)	0	29.02 (100)

However, for moderate levels of skewness one can also choose computationally much faster SVM classifier as it was found to be performing comparably well. Moreover, on the basis of the empirical results obtained in this study we keep ANN classifiers at bottom in the list of feasible non-parametric options for classifying highly skewed datasets on account of their poor performance.

Table 3: Misclassification error rates (in %) of SVM, RF and ANN for simulated skewed data

$p = 2$				$\delta = .5$											
				Sample Size											
a	σ^2	S_k	$D.F.$	25		50		100		400		600		1000	
				APE	AE	APE	AE	APE	AE	APE	AE	APE	AE	APE	AE
0	1.5	477.43	SVM	40.80	47.24	42.50	47.62	45.53	47.52	45.15	45.95	46.20	45.62	46.07	45.63
			RF	0	49.20	0	51.22	0	47.51	0	47.73	0	47.37	0	47.15
			ANN	44.75	49.16	45.75	48.13	47.54	48.50	47.71	47.60	47.45	47.32	48.05	47.61
1		478.92	SVM	33.73	38.34	36.50	38.17	36.76	39.32	38.20	38	37.84	37.60	37.97	37.65
			RF	0	37.25	0	37.62	0	37.52	0	36.72	0	35.72	0	35.79
			ANN	46.08	44.85	42.42	42.65	40.73	41.45	41.44	41.49	41.85	41.22	42.55	42.16
2		288.42	SVM	33.06	37.05	34.60	36.92	35.25	37.23	35.00	35.96	34.10	34.86	33.63	34.19
			RF	0	22.65	0	21.63	0	22.58	0	21.11	0	21.11	0	21
			ANN	39.17	39.57	40.08	40.84	39.90	40.63	36.89	37.21	39.56	39.78	38.55	38.77
0	3	458.24	SVM	26.53	42.23	33.53	42.04	37.33	41.07	39.75	41.82	40.40	41.85	40.52	42.21
			RF	0	38.21	0	38.05	0	36.60	0	35.70	0	35.35	0	36.08
			ANN	44.83	48.34	45.58	47.37	45.33	45.58	45.41	45.79	46.24	46.55	47.27	47.54
1		607.63	SVM	27.26	34.32	30.46	33.97	25.31	32.62	30.23	31.63	30.93	31.42	31.92	31.87
			RF	0	34.66	0	33.25	0	32.97	0	32.40	0	32.20	0	32.20
			ANN	42.94	44.44	42.29	42.68	44.75	45.27	44.40	44.30	44.84	44.77	43.13	43.14
2		453.63	SVM	13.33	28.23	18.20	26.57	21.50	24.84	25.50	26.30	24.53	24.76	25.92	26.13
			RF	0	23.80	0	23.29	0	22.03	0	21.07	0	20.94	0	20.20
			ANN	40.75	42.79	40.42	41.55	42.77	42.38	43.56	43.67	41.85	41.56	43.20	43.16
0	8	713.55	SVM	19.73	33.86	26.70	32.13	30.70	32.66	36.60	36.60	35.48	35.14	39.54	39.47
			RF	0	24.07	0	21.86	0	21.17	0	20.63	0	20.39	0	20.01
			ANN	47.50	47.13	45.58	46	47.75	47.66	48.73	48.65	48.30	48.64	48.93	49.29
1		516.40	SVM	18.20	31.01	24.03	28.74	28.45	30.75	34.69	34.69	37.25	36.86	38.35	38.21
			RF	0	23.72	0	20.94	0	20.44	0	19.31	0	18.93	0	19.09
			ANN	43.50	43.54	44.92	45.83	46.67	46.83	48.39	48.51	47.71	47.99	48.97	49.20
2		666.76	SVM	18.46	28.17	25.33	28.45	28.41	29.85	32.39	32.70	37.12	37.37	41.47	41.23
			RF	0	21.15	0	19	0	19.12	0	17.68	0	17.88	0	17.56
			ANN	18.46	28.17	25.33	28.45	28.41	29.85	32.39	32.70	37.12	37.37	41.47	41.23

References

[1] Bache, K. & Lichman, M. (2013). UCI machine learning repository. http://archive.ics.uci.edu/ml.

[2] Benediktsson, Jon A, Swain, Philip H, & Ersoy, Okan K. 1990. Neural Network Approaches Versus Statistical Methods In Classification Of Multisource Remote Sensing Data. *Geoscience and Remote Sensing, IEEE Transactions on*, 28(4), 540–552.

[3] Breiman, L. (2001). Random forests. *Machine Learning*, 45(1):5–32.

[4] Breiman, Leo, (1996). Bagging predictors. *Machine learning*, 24(2):123–140.

[5] Clarke, W., Lachenbruch, P., & Broffitt, B. (1979). How non normality affects the quadratic discriminant function. *Communications in statistics*, A8:1285–1301.

[6] Cutler, D. R., T.C. Edwards, J., Beard, K., Cutler, A., & et al., K.T. Hess, (2007). On the comparison of classifiers for microarray data. *Ecological Society of America*, page 2783.

[7] Diaz-Uriarte, R. & de Andres, A. (2006). Gene selection and classification of microarray data using random forests. *BMC Bioinformatics*, 7:3.

[8] Dudoit, S., Fridlyand, Jane, & Speed, Terence P. (2002). Comparison of discrimination methods for the classification of tumors using gene expression data. *Journal of the American statistical association*, 97(457):77–87.

[9] Erbeka, F. S., Ozkan, C., & Taberner, M. (2004). Comparison of maximum likelihood classification method with supervised artificial network algorithms for land use activities. *International Journal of Remote Sensing*, 25(9):1733–1748.

[10] Haykin, Simon. (1999). Multilayer perceptrons. *Neural Networks: A Comprehensive Foundation*, 2:156–255.

[11] Huang, C., Davis, L., & Townshend, J. (2002). An assessment of support vector machines for land cover classification. *International Journal of Remote Sensing*, page 725.

[12] Huang, X., Pan, W., & et al., S. Grindle, (2005). A comparative study of discriminating human heart failure etilogy using gene expression profiles. *BMC Bioinformatics*, 6:205.

[13] Joachims, Thorsten. (1998). Text categorization with support vector machines: Learning with many relevant features. In *Machine Learning: ECML-98*, volume 1398 of *Lecture Notes in Computer Science*, pages 137–142. Springer Berlin Heidelberg.

[14] Kavzoglu, T. & Kolkesen, I. (2009). A kernel function analysis for support vector machines for land cover classification. *International Journal of Applied Earth Observation and Geoinformation*, page 530.

[15] Khondoker, M., Dobson, R., Skirrow, C., & et al., A. Simmons. (2013). A comparison of machine learning methods for classification using simulation with multiple

real data examples from mental health studies. *Statistical Methods in Medical Research*, page 1.

[16] Knerr, Stefan, Personnaz, Léon, & Dreyfus, Gérard. (1990). Single-layer learning revisited: a stepwise procedure for building and training a neural network. In *Neurocomputing*, pages 41–50. Springer.

[17] Kotsiantis, S. B. (2007). Supervised machine learning: A review of classification techniques. *Informatica*, 31(3).

[18] Lachenbruch, P.A. (1975). *Discriminant Analysis*. Hafner Press, New York.

[19] Lee, Jae Won, Lee, Jung Bok, Park, Mira, & Song, Seuck Heun. (2005). An extensive comparison of recent classification tools applied to microarray data. *Computational Statistics & Data Analysis*, 48(4):869–885.

[20] Lu, D., Mausel, P., Batistella, M., & Moran, E. (2004). Comparison of land cover classification methods in the Brazilian Amazon basin. *Photogrammetric Engineering and Remote Sensing*, 70:723–731.

[21] Mardia, K.V. (1970). Measures of multivariate skewness and kurtosis with applications. *Biometrika*, 57:519–530.

[22] Mathur, A. & Foody, G.M. (2008). Multiclass and binary svm classification: Implications for training and classification users. *Geoscience and Remote Sensing Letters, IEEE*, 5(2):241–245.

[23] Mountrakis, Giorgos, Im, Jungho, & Ogole, Caesar. (2011). An assessment of support vector machines for land cover classification. *ISPRS Journal of Photogrammatery and Remote Sensing*, page 247.

[24] Olthof, I., King, D., & Lautenschlager, R.A. (2004). Mapping deciduous forest ice storm damage using landsat and environmental data. *Remote Sensing of Environment*, 89:484–496.

[25] Otukei, J.R. & Blaschke, T. (2010). Land cover change assessment using decision trees, support vector machines and maximum likelihood classification algorithms. *International Journal of Applied Earth Observation and Geoinformation*, 12:S27–S31.

[26] Pal, M. & Mather, P.M. (2004). Assessment of the effectiveness of support vector machines for hyperspectral data. *Future Generation Computer Systems*, 20:1215–1225.

[27] Pirooznia, Mehdi, Yang, Jack Y, Yang, Mary Qu, & Deng, Youping. (2008). A comparative study of different machine learning methods on microarray gene expression data. *BMC genomics*, 9(1):1.

[28] Prinzie, A. & den Poel, D. Van. (2008). Random forests for multiclass classification: Random multinomial logit. *Expert Systems with Applications*, page 1721.

[29] Richard, Michael D. & Lippmann, Richard P. (1991). Neural network classifiers estimate bayesian a posteriori probabilities. *Neural computation*, 3(4):461–483.

[30] Rocke, D., Ideker, T., & et al., O. Troyanskaya. (2009). Papers on normalization, variable selection, classification or clustering of microarray data. *Bioinformatics*, 25:701.

[31] Rumelhart, DE, Hinton, GE, & Williams, RJ. (1986). Learning representations by back-propagating errors. *Nature*, 323:533–536.

[32] Shao, Yang & S., Lunetta Ross. (2012). Comparison of support vector machines, neural networks and cart algorithms for the land-cover classification using limited training data points. *ISPRS Journal of Photogrammetry and Remote Sensing*.

[33] Statnikov, A., Wang, L., & Aliferis, C. (2008). A comprehensive comparison of random forests and support vector machines for microarray-based cancer classification. *BMC Bioinformatics*, 9:319.

[34] Tso, Brandt & Mather, Paul M. (2009). *ClassificationMethods for Remotely Sensed Data*. CRC Press, Second edition.

[35] Vapnik, Vladimir Naumovich & Vapnik Vlamimir. (1998). *Statistical learning theory*, volume 1. Wiley New York.

[36] Vladimir, V. N. & Vapnik, V. (1995). *The nature of statistical learning theory*. Springer-Verlag, New York.

[37] Yousefi, Mohammadmahdi R, Hua, Jianping, & Dougherty, Edward R. 2011. Multiple-rule bias in the comparison of classi
cation rules. *Bioinformatics*, 27(12), 1675–1683.

[38] Zhang, Guoqiang, Hu, Michael Y., Patuwo, B. Eddy, & Indro, Daniel C. (1999). Artificial neural networks in bankruptcy prediction: General framework and cross-validation analysis. *European journal of operational research*, 116(1):16–32.