

Prediction of Imbalanced Data Using Cluster Based Approach

¹B.V. Sumana and ²T. Santhanam

¹Department of Computer Science, Vijaya College, Jayanagar, India

²Department of Computer Applications, DG Vaishnav College, Chennai, India

Abstract: Dataset with class imbalance is a challenging problem in many real-world application domains in the field of machine learning and data mining community which is the main cause for the degradation of the classifier performance. A data set is said to be imbalanced if the distribution of instances belonging to each class is not in equal proportion. Researchers worked on class imbalance have identified that combination of class overlapping with class imbalance and high dimensional data is crucial problems which are the important factors for the deterioration of the classifier performance. To overcome this problem a model with two phases of preprocessing is proposed. The objective of the proposed model is 3 fold: increase the minority class instances to address the class imbalance problem, to remove class overlap using the proposed model and to reduce Type 1 and 2 error of the classifier, i.e., false positive and false negative rate which means that the patients who actually does not have disease but predicted as to have disease and vice versa which is a serious problem in reality as it is a matter of life of a patient. The efficiency of the proposed model was evaluated with performance measures like precision, recall, F-measure, AUC, accuracy, kappa, false positive rate and false negative rate. Results proved that proposed model is more efficient than the existing models in the literature as all the 9 classifiers on all the three datasets showed accuracy above 99% and a significant reduction in false positive and false negative rate and also the proposed model was successful in overcoming the issues associated with the real world data sets like class overlap and class imbalance and finally improving the performances of the classifier like false positive, false negative rate, Auc and accuracy of the classifier.

Key words: Class imbalance, class overlap, classification, clustering, K-means, medical dataset

INTRODUCTION

Due to our daily day activities huge amount of data is getting accumulated which often has knowledge hidden which is not available until and unless analyzed. Data mining is the process used for discovering hidden knowledge from these huge real time databases which often suffer from the problem of class imbalance. Class imbalance commonly occurs in real-world datasets such as text classification, detection of oil spills, medical diagnosis, banking operations, failure in a manufacturing unit and credit card fraud detection (Kaveri and Abhilisha, 2016; Elrahman and Abraham, 2013). Classification of medical data is one of the most challenging tasks as they are often imbalanced.

Class imbalance: The data is said to be imbalanced if the classes are not represented in equal proportion. The class representing with higher number of instances is called majority class and the class representing with lesser number of instances is called minority class. Due to class imbalance nature of the dataset classification task becomes very difficult because the classifiers gets biased

towards the majority class as it does not get the necessary information about the minority class to make an accurate prediction therefore show poor classification rates on minority class, as it treats the instances of the minority class as noise. If the majority class is a positive class then sensitivity will be high means the classifier gets biased towards positive class and if the majority class is negative class then specificity will be high means the classifier gets biased towards negative class (Lin and Chen, 2013). Therefore, a balanced dataset is essential for building a good prediction model because most of the algorithms give a better result when the number of instances of each class is approximately equal in proportion (Guo *et al.*, 2016). However in the literature there are few researches discussing about the other issues of class imbalances like Class overlap, Lack of Density of Minority Class. Prati *et al.* (2004) conducted experiments and proved that degradation in the performance of the classifiers is not only due to the class imbalance but also due to the class overlap as they have strong correlation between them. Gracia *et al.* (2006, 2007) also proved that class imbalance and class overlap are the two important factors for the

deterioration performance of the classifier. Gracia *et al.* (2007) concluded that class overlap is equally responsible for the deterioration of the classifier performance and proved that increase in class overlap will degrade the classifier performance. Xiong (2010), conveyed that combination of class imbalance and class overlap makes the prediction task a challenging and crucial as misclassification often occurs near class boundaries where overlapping occurs and presence of high dimensional data is an added issue which makes the prediction task more complicated (Lin and Chen, 2012).

Class overlap: When samples from different classes have similar characteristics they do not form separate clusters and are not linearly separated instead few samples overlap in the data space known as overlapping samples Liu (2008) stated that the overlapping region contains data from more than one classes and misclassification often occurs near the class boundaries where overlapping occurs. Kaveri and Abhilasha (2016) specified that two classes are linearly separable when there exists a hyper plane that can divide the data space such that all instances of one class are on one side of the hyper plane and all instances of other class are on other side of the hyper plane and if the classes of a dataset are not linearly separable then there exists a class overlap problem. Perez *et al.* (2015) specified that classifier performs better when there exist a clear separation in class area.

In the literature numerous techniques have been proposed to solve the problems of the above discussed issues.

Methods to overcome class imbalance: The strategies to tackle the imbalance problem can be broadly divided into two categories, Data level approach and algorithm level approach (He and Garcia, 2009).

Data level approach: Data level approaches aim to modify an imbalanced data into balanced data to overcome the classifier getting biased towards majority class using sampling methods or synthetic data generation methods.

Sampling methods: The sampling-based methods can be further divided into three different categories Under sampling, Over sampling and Hybrid sampling methods.

Under sampling methods: Balances the proportion of the class distribution by randomly eliminating the samples of the majority class retaining the minority class samples. The disadvantage of this method is that there is loss of valuable information of the majority class.

Data cleaning methods: Data cleaning methods is a kind of under sampling method whose aim is to remove the overlapping samples that are generated by the sampling techniques (He and Garcia, 2009). The various data cleaning methods are:

Condensed nearest neighbor: Condensed nearest neighbor selects the subset of instances that are able to correctly classify the original datasets using a one-nearest neighbor rule.

Edited nearest neighbor: Edited nearest neighbor removes samples whose class label differs from the class of at least two of its three nearest neighbors.

Neighborhood cleaning rule: Neighborhood cleaning rule is a modification of the edited nearest neighbor method. Firstly, NCL removes negatives samples which are misclassified by their 3-nearest neighbors. Secondly, the neighbors of each positive sample are identified and the ones belonging to the majority class are removed.

Tomek link: This method works as follows: given two examples e_i and e_j belonging to different classes, with $d(e_i, e_j)$ the distance between e_i and e_j . A (e_i, e_j) pair is called a Tomek link if there is no example e_l , such that $d(e_i, e_l) < d(e_i, e_j)$ or $d(e_j, e_l) < d(e_i, e_j)$ then either one of these examples is noise or both examples are borderline.

One side selection: One side selection is a combination of Tomek links and Condensed Nearest Neighbor which finds the points in the dataset that are totem link using 1-NN and then removes only majority class instances that are totem links.

Oversampling methods: Balances the proportion of the class distribution by randomly replicating the samples of the minority class from the existing samples retaining the majority class samples. The advantage of this method is that there is no loss of information whereas the disadvantage is that addition of replicated samples may lead to over fitting.

Synthetic data generation: This method generates artificial data using bootstrapping and K-nearest neighbors to balance the class distribution. Some of the synthetic data generation method are SMOTE, ROSE, ADASYN.

SMOTE: The Synthetic Minority Over-sampling Technique (SMOTE) approach over-samples the minority class by creating “synthetic” examples instead of

resampling with replacement. To generate artificial data, it uses bootstrapping and k-nearest neighbors. It takes the difference between the feature vector (sample) under consideration and its nearest neighbor then multiply the difference by a random number between 0 and 1 and finally add it to the feature vector under consideration (Chawla *et al.*, 2002).

ROSE: Rose also known as Random Over-Sampling Examples is a balancing method available in ROSE package in R language which allows us to generate artificial data based on sampling methods and smoothed bootstrap approach. ROSE creates a sample of synthetic data by enlarging the features space of minority and majority class examples. Operationally, the new examples are drawn from a conditional kernel density estimate of the two classes (Menardi and Torelli, 2014).

Hybrid methods: Hybrid methods is combination of both under sampling and oversampling methods which balances the class distribution by under sampling the majority class and oversampling the minority class. It has the advantages of both the methods as well as disadvantages.

There are other advanced methods as well for balancing imbalanced data sets. These are Cluster based sampling, adaptive synthetic sampling, border line SMOTE, SMOTEboost, DataBoost-IM, kernel based methods.

Algorithm level approach

Cost based approach: It does not create balanced data distribution. Instead, it learns the imbalanced learning problem using cost matrices. During the model construction a higher misclassification cost is assigned for minority class objects and classification is performed such that it has a lower cost. Let $C(i, j)$ denote the cost of estimating an example from class i as class j . In a two class problem $C(+, -)$ signifies the cost of misclassifying a positive sample as the negative sample and $C(-, +)$ denotes the cost of the contrary case. Cost sensitive learning methods take advantage of the fact that it is more expensive to misclassify a true positive instance than a true negative instance that is $C(+, -) > C(-, +)$. For a two class problem a cost sensitive learning method assigns a greater cost to false negatives than to false positives hence resulting in a performance improvement with respect to the positive class (He and Garcia, 2009).

Methods to overcome class overlap: Preprocessing the data before learning a classification model is a solution to address the class overlap problem which is a two step

process. Firstly it identifies the overlap region in the data space followed by handling the samples in the overlapping region using methods like data cleaning methods and cluster based methods (He and Garcia, 2009).

Xiong proposed 3 different schemes to overcome class overlap, namely: discarding, merging and separating schemes. The discarding scheme ignores the data in overlapping region and learns from the data belonging to the non-overlapping region. The merging scheme identifies the overlapping region and considers the overlapping region as a new class and uses a 2-tier classification model. The upper tier classifier focuses on the entire dataset with an additional class representing the overlapping region. The test data is tested on the first model if it is classified as overlapping then it is tested on the second model to identify the original class. In the separating scheme the data from the overlapping and non-overlapping are identified and builds two models. The test data is tested on the model and identifies if it belongs to overlapping or non-overlapping regions.

Literature review: Rahman and Davis (2013) stated that most of the medical datasets are always imbalanced in nature with respect to their class proportion. Ganganwar (2012) and Gu *et al.* (1986) stated that Machine learning algorithms usually assumes that always the datasets are balanced. Batista *et al.* (2004) stated that the performances of the machine learning algorithms are not well when the datasets are imbalanced. Prati *et al.* (2004) specified that performance degradation of the classifiers is not only due to the class imbalance problem but also due to class overlap. Garcia *et al.* (2006) conducted experiments by varying the degree of class overlapping and class imbalance and concluded that degradation of performance of the classifiers is not only due to the class imbalance but also due to the class overlap and also specified that issue of class overlap is more important factor to be considered than class imbalance for the classification performance. Japkowicz and Stephan (2002) and Japkowicz (2003) investigated in his research that the class imbalance problem affect classifiers like decision trees, neural networks and svm and also specified that svm are not sensitive to class imbalance problem when the classes are linearly separable. Yanjun Qi suggested that when the classes are separated linearly in the data space the performance of the classifiers will be to maximum. Weiss *et al.* (2004) mentioned that noise create problem when positive class samples are present in the negative class and negative class samples are present in the positive class which is known as class overlapping.

Japkowicz and Stephan (2002) concluded that linearly separable domains are not sensitive to class imbalance. Das *et al.* (2014) specified in his research that preprocessing data before learning is a solution to overcome class overlap problem where preprocessing is done two steps. In the first step, overlap regions is identified in the data space and in the second step methods like discarding, separating and merging should be used to handle the overlap region. Denil and Trappenberg (2010) conducted experiments to check if class imbalance and class overlap are independent factors. Results proved that the two factors are not independent and their combination has more performance degrading effect on the classifier. Further results also proved that for small training sets with high levels of imbalance there is degradation in the classifier performance whereas for large training sets with any level of imbalance the degradation of the classifier performance is not much and sometimes almost negligible. Xiong explained the various schemes to overcome class overlapping problem. Ali *et al.* (2015) stated that feature selection is one of the strategy used to remove class overlap.

Background problems: The classification accuracy of a given algorithm generally depends on the nature of data set rather than the algorithm. Usually real world datasets will be class imbalanced in nature with high dimensional redundant or irrelevant attributes, noisy instances, outliers and class overlap. In literature there are some papers discussing about the issues pertaining to class imbalance data. They are as follows: due to class imbalance nature there will be degradation in the performance of the classifiers Prati *et al.* (2004) and Garcia *et al.* (2006). As such class imbalance is not a crucial problem but combination of class overlap with class imbalance is a crucial problem and cause for the degradation performance of the classifier Prati *et al.* (2004) presence of small disjuncts in minority class Japkowicz and Ali *et al.* (2015) lack of data for learning about the minority class data Garcia *et al.* (2006) and presence of noise Garcia *et al.* (2006).

Despite of many researches in the literature, most of the works concentrate on overcoming the imbalance problem rather than class overlap problem. Many methods are proposed to overcome the combined effect but still failed in reducing the false negative and false positive which is a very serious problem in the medical field because a patient without disease is predicted as having disease and wrongly treated and a patient with disease is predicted as not having disease and not given a proper treatment. In both the cases the patient is suffering from

ill treatment hence to overcome this problem a model is proposed to reduce the false positive rate and false negative rate consistently. Quality of prediction depends on quality of data Sumana and Santhanam (2016). According to the literature review of Ali *et al.* (2015) on class imbalance, class overlapping is caused by the presence of irrelevant and redundant features and feature selection is one of the strategies used to address the issue. This research gap has motivated to propose a model to remove class overlap by eliminating the redundant and irrelevant features and thus improve the False positive and False negative rate.

MATERIALS AND METHODS

The proposed model is an extension of Sumana and Santhanam (2016) applied for highly imbalanced and class overlap dataset evaluated using R language on Heart and Appendicitis datasets collected from Keel Repository Alcalá-Fdez *et al.* (2011) and Parkinsons from UCI repository UCI using stratified 10-fold cross validation to test the performance of the classifiers on an imbalanced dataset. In the following subsections the dataset used, experimental setup of the proposed model and results are discussed. The impact of how classifier performance on an imbalanced dataset is increased with the proposed preprocessing is examined.

The objective of the proposed model is 3 fold increase the minority class instances to overcome class imbalance problem, to remove class overlap using the proposed model, to reduce the Type1 and 2 error of the classifier, False positive and False negative rate which means that the patients who actually does not have disease but predicted as to have disease and vice versa as in reality it is a matter of life of a patient. Therefore a model is proposed in this paper to overcome the issues as explained above by addressing each issue using a suitable strategy.

Proposed model: To address the issues discussed in the background problem, two models were proposed in two different ways one using Rose balancing method and second using Smote balancing method. The proposed model was developed in three phases with two phases of preprocessing and a classification phase. Phase1 addressing the class imbalance problem and phase2 addressing the combined effect of class overlap and high dimensional data and phase3 is the classification phase for prediction. Initially in phase1 missing values are replaced if present followed by normalization using min-max which transforms the data to a common scale to provide equal importance for all the attributes in

prediction and then balance the dataset using hybrid resampling methods like Rose and Smote to produce a balanced dataset with approximately equal representation of classes as a solution to the class imbalance problem and in the second phase to address the class overlap problem, overlap was eliminated in 4 steps Firstly, Tomek links were identified and removed if present followed by feature selection or feature extraction method to remove the redundant and irrelevant attributes and in the next step method to improve the quality of clustering algorithm and to form well defined clusters outliers were eliminated using Box plot then K-means clustering algorithm was applied to identify and eliminate the overlap instances which is further optimized using SVM to remove the misclassified instances which were not eliminated by K-means algorithm. Finally, the classifier was applied after the elimination of class overlap region to build the final classifier model using stratified 10 fold cross validation.

Balance phase

Treating missing values: Real world data are usually incomplete with values not present in one or more attributes of an instance. Presence of missing values is due to various reasons like manual data entry procedures, equipment errors and incorrect measurements, etc., each piece of data has its own contribution in predicting the class. Therefore, instances with missing values should not be deleted as the valuable information present in the other attributes will be lost. If they are not treated properly then the performance of the data mining algorithm will be reduced. The most popular ways to handle the missing value are deleting the observations when the percentage of missing values is too small compared to the data, deleting the variable when there are more missing values compared with the rest of the variables in the dataset. Imputation by mean, median if it is numeric value or mode if it is categorical value Prediction where the missing values are predicted using data mining algorithm like Knn, regression, neural networks, decision trees, etc. (Sumana Santhanam, 2016).

Normalization: Normalization is an important preprocessing step in Data Mining which transforms the input values of all attributes into a common scale to avoid attributes having greater numeric values dominating the smaller numeric range values. Doing so, it gives equal importance to all the variables. Min-max normalization, Z-score normalization and Decimal Scaling normalization are the various types of normalization techniques available in the literature Saranya and Manikandan (2013) min-max normalization performs a linear transformation on

the original data values maintaining the relationships among the original data values. The values of an attribute are mapped from a range [minA, maxA] to a new range [new_minA, new_maxA], where every value v from the original interval will be mapped into value new_v using the following formula:

$$\text{new_v} = (v - \text{minA} / \text{maxA} - \text{minA}) \\ (\text{new_maxA} - \text{new_minA}) + \text{new_minA}$$

This is adapted in the proposed model before performing dimensional reduction and K-means because to provide equal weight for all attributes while generating clusters. Since Euclidean distance is used for calculating centroids the clusters generation will be strongly influenced by the magnitudes of the outliers. To overcome this data is normalized. Normalizing the data not only generate good quality clusters it will also speed up the learning phase of the classifier.

Balance the data

ROSE: ROSE uses smoothed bootstrapping to draw artificial samples from the feature space neighborhoods around the minority class.

Smote: Smote draws artificial samples by choosing points that lie on the line connecting the rare observation to one of its nearest neighbors in the feature space.

Tomek link: Tomek link is applied to remove the overlapping introduced from sampling methods. Tomek links are the pair nearest neighbor instances of opposite classes x_i and x_j where x_i belongs to x_{maj} and x_j belongs to x_{min} and the distance between them is $d(x_i, x_j)$. They are said to be totem link if there exists no instance x_k between these two instances such that $d(x_i, x_k) < d(x_i, x_j)$ or $d(x_j, x_k) < d(x_j, x_i)$ then they form Tomek link means either one of these instances is noise or both are near a border therefore Tomek link in R finds the points in the dataset that are totem link using 1-NN and then removes only majority class instances that are totem links to cleanup unwanted overlapping between classes after synthetic sampling such that all minimally distanced nearest neighbor pairs belongs to the same class. Doing so well defined clusters can be formed He and Garcia (2009).

Feature selection: Data may contain many redundant or irrelevant features. Redundant features are those that provide no more information and irrelevant features are those which provide no useful information. The classification accuracy of a given algorithm generally depends on the nature of the dataset rather than the

algorithm itself. The main characteristics of a dataset are its attributes, classes and number of instances. Feature selection is a form of dimensionality reduction where in the input data will be transformed into a reduced representation set of features eliminating irrelevant features and selecting a subset of relevant features for the model construction which optimizes the accuracy of the classifiers (Gangawar, 2012; Sumana and Santhanam, 2016).

Fuzzy Rough set Feature Selection (FRFS) was adapted, as it can analyze both quantitative and qualitative features and can reduce mixture of nominal and continuous valued features based only on the original data without any additional information about the data. The only additional information required is fuzzy partitions for each feature which can be derived from the data itself. Though Rough set theory proposed by Pawlak has many successful advantages in the extraction of feature subsets it has the limitation of handling only nominal data therefore fuzzy set theory is combined with Rough set to handle continuous data. Hence, the hybrid FRFS can handle mixture of nominal and continuous valued features. The selection of appropriate membership function is the main bottleneck of fuzzy set. Rough set theory is useful for decision making in situation where indistinct. The merits of rough sets and fuzzy sets are integrated to develop a much more powerful and efficient tool known as Fuzzy-Rough Feature Selection (FRFS) (Jensen *et al.*, 2005).

CFS: The CFS with Best First search algorithm is adapted as a feature selection method to select the best attributes and Clustering as a reduction technique applying which the wrongly clustered instances are eliminated to get final samples. The Correlation based Feature Selection (CFS) works on the: hypothesis “Good feature subsets contain features highly correlated with the classification, yet uncorrelated to each other”. In correlation-based Feature selection, feature selection does not depend on any particular data transformation all that must be supplied is a means of measuring the correlation between any two attributes (Hall, 2000).

Recursive feature elimination: Recursive feature elimination is a greedy optimization method that finds the best performing subset of features by repeatedly constructing a model and chooses either the best or worst performing feature keeping the feature aside then repeating the process with the rest of the features. This process is applied until all features in the dataset are exhausted then the features are ranked according to their elimination.

Principal component analysis: Principal Component Analysis (PCA) is a popularly known data preprocessing

technique used for dimensionality reduction that generates principal components which are linear combination of the original data. These principal components are orthogonal and uncorrelated to each other explaining the variation in the data. The principal components with higher variance have higher weightage than the principal component with lower variance hence normalizing the data to a common scale before PCA is important. If d dimensional dataset is considered excluding the class variable, compute the covariance or correlation matrix of the dimensions Find eigenvectors and eigenvalues from the covariance or correlation matrix. Sort eigenvectors in decreasing order of eigenvalues and choose k eigenvectors with the largest eigenvalues to form a $d \times k$ dimensional matrix and transform on to a new subspace called principal components. The principal components are linear combinations of the original variables explaining the variance in orthogonal dimensions. The first principal component explains the largest variability in the data as possible and each succeeding component explains next highest variance. This continues until it is equal to the original number of variables. The PCs with high variability are selected neglecting the PCs with less variability thus dimensionality reduction is achieved. Principal component analysis (PCA) is a most widely used feature extraction method for linear datasets adapted in the proposed model for 3 reasons:

- To transform higher dimensional data into a lower dimensional data as it is easier to visualize the data when reduced to lower dimensions
- To transform correlated variables into a set of linear combinations of the original data called principal components which are uncorrelated with each other because presence of correlated variables degrades the prediction accuracy (Sumana and Santhanam, 2016)

Outlier elimination using Boxplot: Outliers are observations present in the data which are different from all other observations in the data. Clustering algorithms use distance metrics to calculate centroids of the clusters. Presence of outliers deviate the cluster centroid and degrades the performance of clustering algorithms. To overcome this, a research direction is suggested by Sumana and Santhanam (2016) to remove outliers before performing K-means clustering. Hence Boxplot technique is adapted in this proposed model for the removal of outliers.

Clustering: Clustering algorithm is used to classify the data into groups with similar features in the same cluster and different features in other cluster but there often

Table 1: Data set Descriptions

Dataset	No. of attributes	No. of Instances	No. of missing values	Imbalance Ratio (IR)	No of instances		Proportion of instances	
					Majority class	Minority class	Majority class	Minority class
Heart	13	177	4	12.62	164	13	92.66	7.34
Appendicitis	7	106	0	4.05	85	21	80.19	19.81
Parkinsons	22	195	0	3.06	147	48	75.38	24.62

exists samples with few similar characteristics among the different clusters which exist in the overlap region among the clusters. Presence of these samples in the overlap region arises a ambiguous situation for the classifiers in predicting which hinders its performance in prediction to overcome this clustering algorithm is used which groups the data into different groups eliminating the overlapping region. Since in this paper the datasets considered are binary class problems K-means algorithm is used which takes the number of clusters to be grouped priory before executing eliminating the instances in the overlapped region.

K-means clustering: The K-Means Jain (2010) is a well-known partition algorithm which groups the data into k clusters in which the resulting objects of one cluster are similar in the same cluster and dissimilar to that of other cluster using the following steps:

- Takes k as an input to cluster into k groups
- Randomly select k points as the initial centroids
- Calculate the distance between each point and cluster centers
- Assign all data points to the closest centroid
- Recalculate the new centroid of each cluster using the distance formula
- Repeat steps 3 and 5 until a reassignment stops

Misclassified instances: In a binary class problem there always exist misclassified instances which are wrongly classified by the clustering algorithm such that the minority class instances are being present in the majority class cluster and majority class instances being present in the minority class cluster. Therefore it is necessary to identify these wrongly clustered instances and eliminate for better prediction by the classifier hence to optimize the performance of clustering algorithm in this proposed model these wrongly clustered instances are eliminated using SVM.

Optimizing K-means with SVM: Support Vector Machines (SVMs) are supervised machine learning binary classifier which use hyper-planes to separate instances of different class labels. It calculates the hyper plane maximizing the minimum distance between the plane and the training points. "Support Vectors" are defined as

subset of data instances used to define the hyper plane. An optimal separating hyper plane is the hyper plane that maximizes the margin. To classify the data SVMs first find the maximal margin which separates two classes and then outputs the hyper plane separator at the center of the margin. New data is classified by determining on which side of the hyper plane it belongs and hence to which class it should be assigned.

The goal of classification is to minimize the test error therefore classification methods are also used for optimization problems. To enhance the prediction accuracy of the classifier clustering algorithms are used as a preprocessing algorithm similarly classifiers are used to validate the results produced by clustering algorithms. In this proposed model, SVM is used as an optimization algorithm to remove the instances which are wrongly clustered by the clustering algorithm so as to reduce the error of the clustering algorithm. SVM, a binary classifier is adapted in this proposed because the datasets considered are binary classification problem (Sumana and Santhanam, 2016).

Classification: Finally, the relevant instances identified from the preprocessing phase were trained by 9 different classifiers, one from each type of classifiers of WEKA 3.7.2 using stratified 10 fold cross validation. The performances of the classifiers were evaluated based on the confusion matrix. Table 1 illustrates the defined process.

Data set description

Estimations for model performance

Stratified 10 fold cross validation method: In this study, stratified Cross Validation with 10 folds has been used for evaluating the classifier models. Cross validation is a statistical technique used for evaluating the performance of the predictive model and also used to compare learning algorithms by dividing data into 2 segments one used to train a model and the other used to validate the model. Stratification is a process of partitioning the data such that each class is properly represented in both training and test sets. In a stratified 10-fold Cross-Validation the data is divided randomly into 10 parts in which the class is represented in approximately the same proportions as in the full dataset. Each part is held out in turn and the learning scheme trained on the remaining nine-tenths;

then its error rate is calculated on the holdout set. The learning procedure is executed a total of 10 times on different training sets and finally the 10 error rates are averaged to yield an overall error estimate. When seeking an accurate error estimate, it is standard procedure to repeat the CV process 10 times (Sumana and Santhanam, 2016).

Performance measures

Metrics for evaluating model performance: Measures of the quality of classification algorithms are based on the confusion matrix which records correctly and incorrectly recognized examples for each class Elrahman and Abraham (2013), Guo *et al.* (2016) and Aida Ali *et al.* (2015). Table 2 and 3 presents a confusion matrix for binary classification, where TP are True Positive TN is True Negative, FP is False Positive, FN are False Negative. The different measures used with the confusion matrix are (Table 4-11):

Accuracy: The accuracy of a classifier is the percentage of the test set tuples that are correctly classified by the classifier:

$$\text{Accuracy} = (TP+TN)/(TP+TN+FP+FN)$$

Kappa: Kappa is a statistical measure of agreement between the predicted class values to the actual class values. A Kappa value lies between a -1 to 1 scale. A kappa of 1 indicates perfect agreement, whereas a kappa of 0 indicates agreement equivalent to chance and negative values indicate agreement less than chance. Kappa value is calculated using the following equation:

$$K = [P(A)-P(E)]/[1- P(E)]$$

Where:

P(A) = The percentage of agreement between the classifier and the underlying truth calculated

P(E) = The chance of agreement calculated

Table 2: Confusion matrix

Actual class	Predicted class	
	Test negative (T-)	Test positive (T+)
Disease absent (D-)	True Negative (TN)	False Positive (FP)
Disease present (D+)	False Negative(FN)	True Positive (TP)

Table 3: Experimental results of heart data using proposed model

Balancing method (Feature selection /Classifiers (Estimators))	Original	ROSE				SMOTE			
		CFS	RF	FRFS	PCA	CFS	RF	FRFS	PCA
Naive bayes									
Precision	0.59	1.000	0.990	0.990	0.980	0.980	0.990	1	1.000
Recall	0.77	1.000	1.000	1.000	0.980	0.970	100.000	1	1.000
F-measure	0.67	1.000	0.990	0.990	0.980	0.970	0.990	1	1.000
AUC	0.90	1.000	1.000	1.000	1.000	0.990	100.000	1	1.000
Accuracy	94.35	100.000	99.340	99.320	98.570	96.910	99.320	100	100.000
Kappa	0.64	1.000	0.990	0.990	0.970	0.940	0.990	1	1.000
FPR	0.04	0.000	0.013	0.014	0.010	0.028	0.016	0	0.000
FNR	0.23	0.000	0.000	0.000	0.020	0.033	0.000	0	0.000
Support vector machine									
Precision	0.80	1.000	0.990	1.000	1.000	0.990	0.990	1	1.000
Recall	0.62	1.000	0.990	1.000	0.980	1.000	1.000	1	1.000
F-measure	0.70	1.000	0.990	1.000	0.990	0.990	0.990	1	1.000
AUC	0.80	1.000	0.990	1.000	0.990	0.990	0.990	1	1.000
Accuracy	96.05	100.000	98.680	100.000	99.290	99.380	99.320	100	100.000
Kappa	0.67	1.000	0.970	1.000	0.990	0.990	0.990	1	1.000
FPR	0.01	0.000	0.013	0.000	0.000	0.014	0.016	0	0.000
FNR	0.39	0.000	0.013	0.000	0.016	0.000	0.000	0	0.000
IBK									
Precision	0.58	0.990	1.000	0.990	0.980	0.990	0.990	1	0.960
Recall	0.54	0.990	0.970	1.000	1.000	1.000	1.000	1	1.000
F-measure	0.56	0.990	0.990	0.990	0.990	0.990	0.990	1	0.980
AUC	0.79	0.990	0.980	1.000	0.990	1.000	0.990	1	0.950
Accuracy	93.79	98.640	98.680	99.320	99.290	99.380	99.320	100	97.420
Kappa	0.53	0.970	0.970	0.990	0.990	0.990	0.990	1	0.940
FPR	0.03	0.013	0.000	0.014	0.013	0.014	0.016	0	0.080
FNR	0.46	0.015	0.026	0.000	0.000	0.000	0.000	0	0.000
J48									
Precision	0.30	0.970	0.950	0.970	0.980	0.950	1.000	1	0.990
Recall	0.23	0.930	0.960	0.950	1.000	0.970	1.000	1	1.000
F-measure	0.26	0.950	0.960	0.960	0.990	0.960	1.000	1	0.990
AUC	0.47	0.980	0.960	0.960	0.990	0.940	1.000	1	0.990
Accuracy	90.40	95.240	95.390	95.920	99.290	95.060	100.000	100	99.140

Table 3: Continue

Balancing method (Feature selection /Classifiers (Estimators))	Original	ROSE				SMOTE			
		CFS	RF	FRFS	PCA	CFS	RF	FRFS	PCA
Kappa	0.21	0.910	0.910	0.920	0.990	0.900	1.000	1	0.980
FPR	0.04	0.025	0.053	0.028	0.013	0.069	0.000	0	0.026
FNR	0.77	0.074	0.039	0.053	0.000	0.033	0.000	0	0.000
Rep Tree									
Precision	0.20	0.930	0.940	0.920	1.000	0.940	1.000	1	1.000
Recall	0.08	0.930	0.960	0.960	1.000	0.910	1.000	1	1.000
F-measure	0.11	0.930	0.950	0.940	1.000	0.930	1.000	1	1.000
AUC	0.52	0.920	0.960	0.940	1.000	0.920	1.000	1	1.000
Accuracy	90.96	93.200	94.740	93.880	100.000	91.980	100.000	100	100.000
Kappa	0.07	0.860	0.890	0.880	1.000	0.840	1.000	1	1.000
FPR	0.02	0.063	0.067	0.085	0.000	0.069	0.000	0	0.000
FNR	0.92	0.074	0.039	0.039	0.000	0.089	0.000	0	0.000
Ripper									
Precision	0.43	0.940	0.930	0.960	1.000	0.940	1.000	1	0.990
Recall	0.46	0.910	0.970	0.960	0.980	0.910	1.000	1	1.000
F-measure	0.44	0.930	0.950	0.960	0.990	0.930	1.000	1	0.990
AUC	0.81	0.930	0.940	0.940	0.990	0.940	1.000	1	0.990
Accuracy	91.53	93.200	94.740	95.920	99.290	91.980	100.000	100	99.140
Kappa	0.40	0.860	0.890	0.920	0.990	0.840	1.000	1	0.980
FPR	0.05	0.051	0.080	0.042	0.000	0.069	0.000	0	0.026
FNR	0.54	0.088	0.026	0.039	0.160	0.089	0.000	0	0.000
Part									
Precision	0.46	0.950	0.940	0.970	0.980	0.093	1.000	1	0.990
Recall	0.39	0.900	0.950	0.930	1.000	0.940	1.000	1	1.000
F-measure	0.42	0.920	0.940	0.950	0.990	0.940	1.000	1	0.990
AUC	0.74	0.960	0.960	0.950	0.990	0.910	1.000	1	0.990
Accuracy	92.09	93.200	94.080	95.240	99.290	93.210	100.000	100	99.140
Kappa	0.37	0.860	0.880	0.900	0.990	0.860	1.000	1	0.980
FPR	0.04	0.038	0.067	0.028	0.013	0.083	0.000	0	0.026
FNR	0.62	0.103	0.052	0.066	0.000	0.056	0.000	0	0.000
Multilayer perceptron									
Precision	0.57	1.000	1.000	1.000	0.970	1.000	0.990	1	1.000
Recall	0.62	1.000	1.000	1.000	0.980	1.000	1.000	1	1.000
F-measure	0.59	1.000	1.000	1.000	0.980	1.000	0.990	1	1.000
AUC	0.91	1.000	1.000	1.000	1.000	1.000	1.000	1	1.000
Accuracy	93.79	100.000	100.000	100.000	97.860	100.000	99.320	100	100.000
Kappa	0.56	1.000	1.000	1.000	0.960	1.000	0.990	1	1.000
FPR	0.04	0.000	0.000	0.000	0.026	0.000	0.016	0	0.000
FNR	0.39	0.000	0.000	0.000	0.016	0.000	0.000	0	0.000
RBF									
Precision	0.54	0.990	1.000	0.990	0.970	0.960	1.000	1	1.000
Recall	0.54	1.000	0.990	0.990	0.980	0.970	1.000	1	0.990
F-measure	0.54	0.990	0.990	0.990	0.980	0.960	1.000	1	0.990
AUC	0.80	1.000	1.000	1.000	1.000	0.970	1.000	1	1.000
Accuracy	93.22	99.320	99.340	98.640	97.860	95.680	100.000	100	99.140
Kappa	0.50	0.990	0.990	0.970	0.960	0.910	1.000	1	0.980
FPR	0.04	0.013	0.000	0.014	0.026	0.056	0.000	0	0.000
FNR	0.46	0.000	0.013	0.013	0.016	0.033	0.000	0	0.013

Table 4: Scatter plot for heart dataset using rose balancing and tomek link

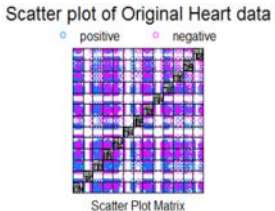
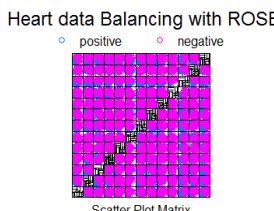
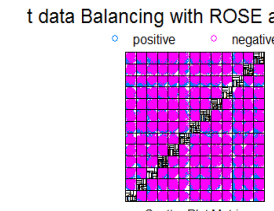
Classes	Original Data	Data after Balancing using Rose method	Data with Tomek link removed
	<p>Scatter plot of Original Heart data</p>  <p>Scatter Plot Matrix</p>	<p>Heart data Balancing with ROSE</p>  <p>Scatter Plot Matrix</p>	<p>t data Balancing with ROSE and Tomek</p>  <p>Scatter Plot Matrix</p>
CFS	Before outlier elimination	After outlier elimination	Heart data with class imbalance and overlap eliminated using proposed model with CFS feature selection

Table 4: Continue

Classes	Original Data	Data after Balancing using Rose method	Data with Tomek link removed
RF	Before outlier elimination	After outlier elimination	Heart data with class imbalance and overlap eliminated using proposed model with Random Forest
FRFS	Before outlier elimination	After outlier elimination	Heart data with class imbalance and overlap eliminated using proposed model with FuzzyRoughset
PCA	Before outlier elimination	After outlier elimination	Heart data with class imbalance and overlap eliminated using proposed model with PCA

Table 5: Scatter plot for heart dataset using Smote Balancing and Tomek link

Classes	Original Data	Data after Balancing using smote method	Data with Tomek link removed
CFS	Before outlier elimination	After outlier elimination	Heart data with class imbalance and overlap eliminated using proposed model with CFS feature selection
RF	Before outlier elimination	After outlier elimination	Heart data with class imbalance and overlap eliminated using proposed model with Random Forest

Table 5: Continue

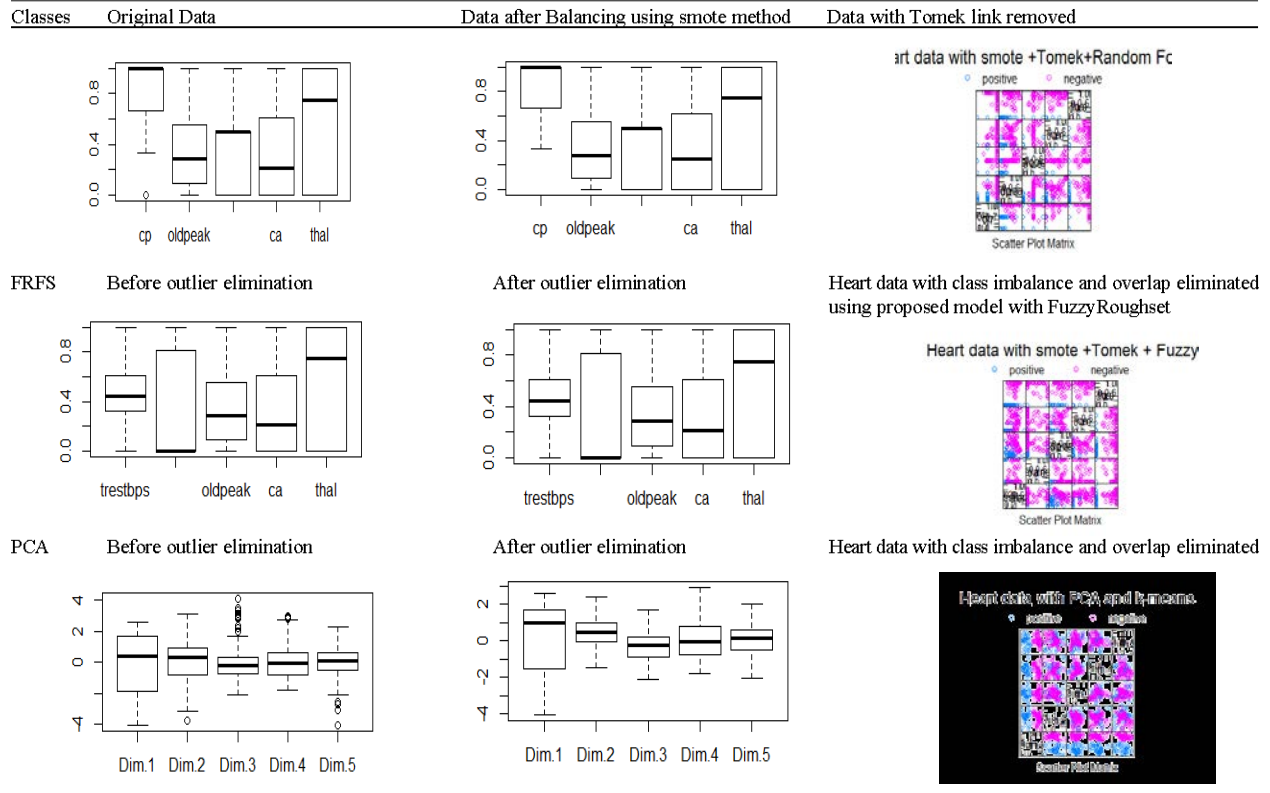


Table 6: Experimental Results of Appendicitis data using proposed model

Balancing method (Feature selection /Classifiers (Estimators))	Proposed model on appendicitis data set								
	Original	Rose				Smote			
		CFS	RF	FRFS	PCA	CFS	RF	FRFS	PCA
Naive Bayes									
Precision	0.64	0.930	0.94	0.97	1.00	1.00	1.00	1.00	1.00
Recall	0.67	0.960	1.00	1.00	1.00	1.00	1.00	1.00	0.96
F-measure	0.65	0.950	0.97	0.99	1.00	1.00	1.00	1.00	0.98
AUC	0.81	1.000	1.00	1.00	1.00	1.00	1.00	1.00	1.00
Accuracy	85.85	95.240	96.97	98.41	100.00	100.00	100.00	100.00	97.96
Kappa	0.56	0.900	0.94	0.97	1.00	1.00	1.00	1.00	0.96
FPR	0.09	0.056	0.05	0.04	0.00	0.00	0.00	0.00	0.00
FNR	0.33	0.037	0.00	0.00	0.00	0.00	0.00	0.00	0.05
Support vector machine									
Precision	0.79	1.000	1.00	0.97	1.00	1.00	1.00	1.00	1.00
Recall	0.52	0.960	1.00	0.97	1.00	1.00	1.00	1.00	0.93
F-measure	0.63	0.980	1.00	0.97	1.00	1.00	1.00	1.00	0.97
AUC	0.74	0.980	1.00	0.97	1.00	1.00	1.00	1.00	0.97
Accuracy	87.74	98.410	100.00	96.83	100.00	100.00	100.00	100.00	96.94
Kappa	0.56	0.970	1.00	0.94	1.00	1.00	1.00	1.00	0.94
FPR	0.04	0.000	0.00	0.04	0.00	0.00	0.00	0.00	0.00
FNR	0.48	0.037	0.00	0.03	0.00	0.00	0.00	0.00	0.07
IBK									
Precision	0.54	0.960	0.96	0.97	1.00	1.00	0.97	1.00	1.00
Recall	0.62	1.000	0.93	0.94	1.00	1.00	1.00	1.00	1.00
F-measure	0.58	0.980	0.95	0.96	1.00	1.00	0.99	1.00	1.00
AUC	0.75	0.980	0.94	0.95	1.00	1.00	0.99	1.00	1.00
Accuracy	82.08	98.410	95.45	95.24	100.00	100.00	98.91	100.00	100.00
Kappa	0.46	0.970	0.91	0.90	1.00	1.00	0.98	1.00	1.00
FPR	0.13	0.028	0.03	0.04	0.00	0.00	0.02	0.00	0.00
FNR	0.38	0.000	0.07	0.06	0.00	0.00	0.00	0.00	0.00

Table 6: Continue

Balancing method (Feature selection /Classifiers (Estimators))	Proposed model on apendicitis dta set								
	Original	Rose				Smote			
		CFS	RF	FRFS	PCA	CFS	RF	FRFS	PCA
J48									
Precision	0.69	0.960	0.96	0.94	1.00	1.00	0.95	1.00	1.00
Recall	0.52	0.930	0.93	0.89	0.97	0.98	0.95	0.97	0.98
F-measure	0.6	0.940	0.95	0.91	0.98	0.99	0.95	0.99	0.99
AUC	0.7	0.970	0.94	0.90	0.98	0.99	0.96	0.99	0.99
Accuracy	85.85	95.240	95.45	90.48	98.41	98.77	95.65	98.48	98.98
Kappa	0.51	0.900	0.91	0.81	0.97	0.98	0.91	0.97	0.98
FPR	0.06	0.028	0.03	0.07	0.00	0.00	0.04	0.00	0.00
FNR	0.48	0.074	0.07	0.11	0.03	0.02	0.05	0.03	0.02
Rep Tree									
Precision	0.64	0.850	0.89	0.94	1.00	1.00	0.97	0.97	1.00
Recall	0.33	0.850	0.83	0.91	1.00	0.98	0.97	0.97	1.00
F-measure	0.44	0.850	0.86	0.93	1.00	0.99	0.97	0.97	1.00
AUC	0.73	0.900	0.83	0.95	1.00	0.99	0.98	0.98	1.00
Accuracy	83.02	87.300	87.88	92.06	100.00	98.77	97.83	96.97	100.00
Kappa	0.35	0.740	0.75	0.84	1.00	0.98	0.95	0.94	1.00
FPR	0.05	0.110	0.08	0.07	0.00	0.00	0.02	0.03	0.00
FNR	0.67	0.150	0.17	0.09	0.00	0.02	0.03	0.03	0.00
Ripper									
Precision	0.67	0.900	0.96	0.89	1.00	0.98	0.95	0.97	1.00
Recall	0.57	0.960	0.86	0.89	0.97	1.00	0.95	1.00	0.98
F-measure	0.62	0.930	0.91	0.89	0.98	0.99	0.95	0.99	0.99
AUC	0.7	0.930	0.92	0.90	1.00	0.99	0.96	1.00	0.99
Accuracy	85.85	93.650	92.42	87.30	98.40	98.77	95.65	98.48	98.98
Kappa	0.53	0.870	0.85	0.74	0.97	0.98	0.91	0.97	0.98
FPR	0.07	0.080	0.03	0.14	0.00	0.03	0.04	0.03	0.00
FNR	0.43	0.040	0.14	0.11	0.03	0.00	0.05	0.00	0.02
Part									
Precision	0.65	0.960	0.96	0.94	1.00	1.00	0.95	1.00	1.00
Recall	0.52	0.930	0.93	0.89	0.97	0.98	0.95	0.97	0.98
F-measure	0.58	0.940	0.95	0.91	0.98	0.99	0.95	0.99	0.99
AUC	0.7	0.970	0.94	0.90	0.98	0.99	0.96	0.99	0.99
Accuracy	84.91	95.240	95.45	90.48	98.40	98.77	95.65	98.48	98.98
Kappa	0.49	0.900	0.91	0.81	0.97	0.98	0.91	0.97	0.98
FPR	0.07	0.030	0.03	0.07	0.00	0.00	0.04	0.00	0.00
FNR	0.48	0.070	0.07	0.11	0.03	0.02	0.05	0.03	0.02
Multilayer pnceptron									
Precision	0.67	0.930	1.00	0.97	1.00	1.00	0.97	1.00	1.00
Recall	0.57	1.000	1.00	0.97	1.00	1.00	1.00	1.00	0.98
F-measure	0.62	0.960	1.00	0.97	1.00	1.00	0.99	1.00	0.99
AUC	0.78	1.000	1.00	1.00	1.00	1.00	1.00	1.00	1.00
Accuracy	85.85	96.830	100.00	96.83	100.00	100.00	98.91	100.00	98.98
Kappa	0.53	0.940	1.00	0.94	1.00	1.00	0.98	1.00	0.98
FPR	0.07	0.060	0.00	0.04	0.00	0.00	0.02	0.00	0.00
FNR	0.43	0.000	0.00	0.03	0.00	0.00	0.00	0.00	0.02
RBF									
Precision	0.64	0.960	0.97	0.97	1.00	1.00	0.98	1.00	1.00
Recall	0.43	0.960	0.97	0.97	1.00	1.00	1.00	1.00	1.00
F-measure	0.51	0.960	0.97	0.97	1.00	1.00	0.99	1.00	1.00
AUC	0.82	0.980	0.96	1.00	1.00	1.00	1.00	1.00	1.00
Accuracy	83.96	96.830	96.97	96.83	100.00	100.00	98.91	100.00	100.00
Kappa	0.42	0.940	0.94	0.94	1.00	1.00	0.98	1.00	1.00
FPR	0.06	0.030	0.03	0.04	0.00	0.00	0.02	0.00	0.00
FNR	0.57	0.040	0.03	0.03	0.00	0.00	0.00	0.00	0.00

Table 7: Scatter plot for Appendicitis dataset using Rose Balancing and Tomek link

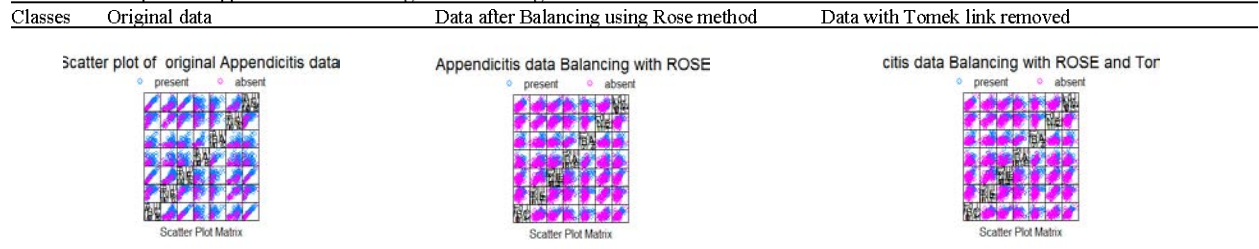


Table 7: Continue

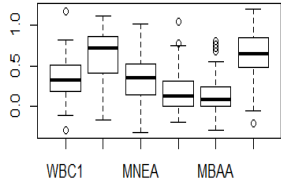
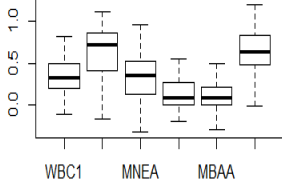
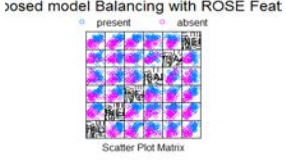
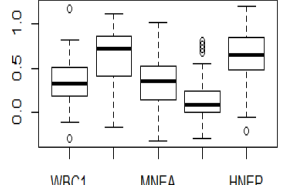
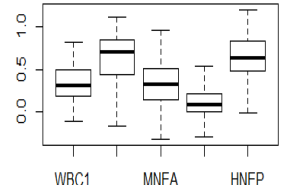
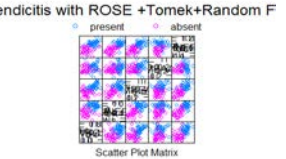
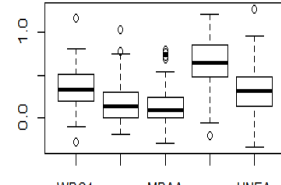
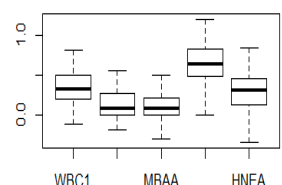
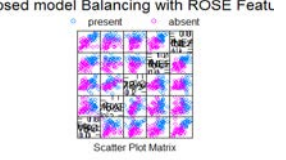
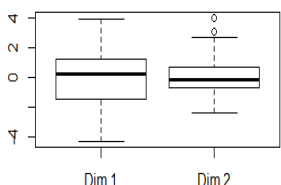
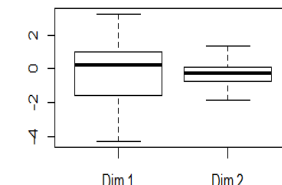
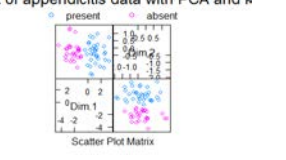
Classes	Original data	Data after Balancing using Rose method	Data with Tomek link removed
CFS	Before outlier elimination 	After outlier elimination 	Appendicitis data with class imbalance and overlap eliminated using proposed model with CFS feature selection 
RF	Before outlier elimination 	After outlier elimination 	Appendicitis data with class imbalance and overlap eliminated using proposed model with Random Forest 
FRFS	Before outlier elimination 	After outlier elimination 	Appendicitis data with class imbalance and overlap eliminated using proposed model with FuzzyRoughset 
PCA	Before outlier elimination 	After outlier elimination 	Appendicitis data with class 

Table 8: Scatter plot for Appendicitis dataset using Smote Balancing and Tomek link

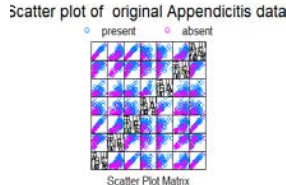
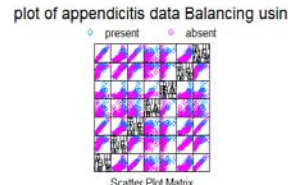
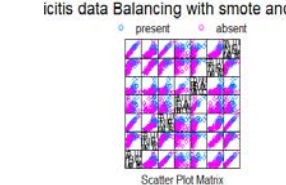
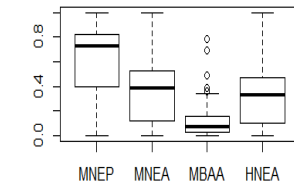
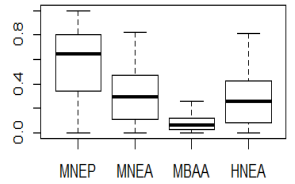
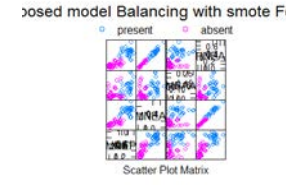
Classes	Original Data	Data after Balancing using smote method	Data with Tomek link removed
	Scatter plot of original Appendicitis data 	plot of appendicitis data Balancing using 	Appendicitis data Balancing with smote and Tor 
CFS	Before outlier elimination 	After outlier elimination 	Appendicitis data with class imbalance and overlap eliminated using proposed model with CFS feature selection 
RF	Before outlier elimination	After outlier elimination	Appendicitis data with class imbalance and overlap eliminated using proposed model with Random Forest

Table 8: Continue

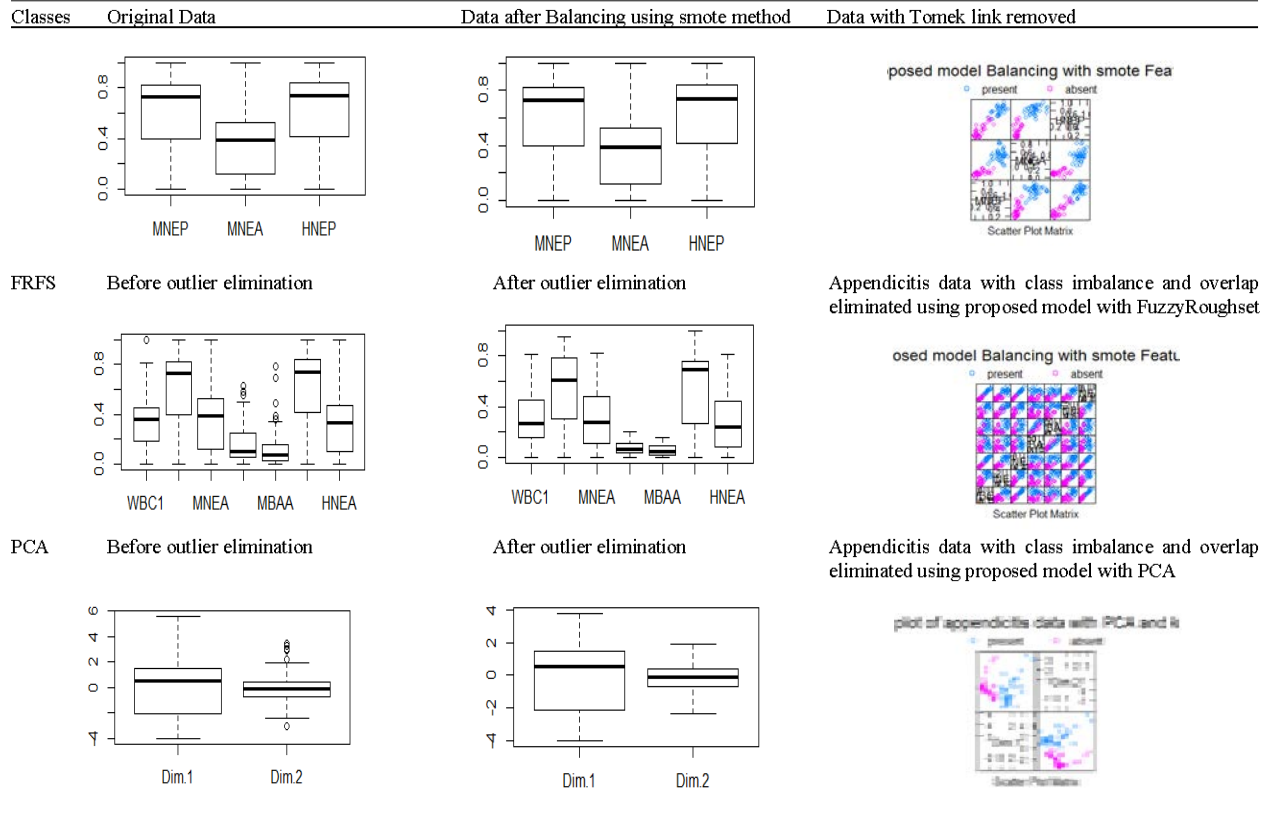


Table 9: Experimental Results of Parkinson data using proposed model

Balancing method (Feature selection /Classifiers (Estimators))	Proposed model on apendicitis dta set								
	Original	ROSE				SMOTE			
		CFS	RF	FRFS	PCA	CFS	RF	FRFS	PCA
Naive Bayes									
Precision	0.96	1.00	1.00	1.00	0.98	0.9	0.96	1.00	0.95
NPV	0.44	1.00	0.98	1.00	1.00	1.00	1.00	1.00	0.96
Recall	0.63	1.00	0.8	1.00	1.00	1.00	1.00	1.00	0.95
F-measure	0.76	1.00	0.89	1.00	0.99	0.95	0.98	1.00	0.95
AUC	0.86	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
Accuracy	69.74	100.00	98.25	100.00	99.14	96.34	97.14	100.00	95.35
Kappa	0.40	1.00	0.88	1.00	0.98	0.92	0.94	1.00	0.91
FPR	0.08	0.00	0.00	0.00	0.01	0.06	0.07	0.00	0.04
FNR	0.37	0.00	0.20	0.00	0.00	0.00	0.00	0.00	0.05
Support vector machine									
Pecision	0.86	1.00	1.00	1.00	1.00	0.93	0.98	1.00	1.00
NPV	0.96	0.98	0.98	1.00	0.97	1.00	1.00	1.00	0.98
Recall	0.99	0.96	0.80	1.00	0.95	1.00	1.00	1.00	0.97
F-measure	0.92	0.98	0.89	1.00	0.98	0.96	0.99	1.00	0.99
AUC	0.75	0.98	0.90	1.00	0.98	0.98	0.98	1.00	0.99
Accuracy	87.18	98.57	98.25	100.00	98.28	97.56	98.57	100.00	98.84
Kappa	0.59	0.97	0.88	1.00	0.96	0.95	0.97	1.00	0.98
FPR	0.50	0.00	0.00	0.00	0.00	0.04	0.04	0.00	0.00
FNR	0.01	0.05	0.20	0.00	0.05	0.00	0.00	0.00	0.03
IBK									
Precision	0.99	1.00	1.00	1.00	0.93	1.00	1.00	1.00	1.00
NPV	0.9	0.98	0.95	1.00	0.95	1.00	1.00	1.00	0.98
Recall	0.97	0.96	0.40	1.00	0.91	1.00	1.00	1.00	0.94
F-measure	0.98	0.98	0.57	1.00	0.92	1.00	1.00	1.00	0.99
AUC	0.97	0.98	0.79	1.00	0.95	1.00	1.00	1.00	0.99
Accuracy	96.41	98.57	94.74	100.00	93.97	100.00	100.00	100.00	98.84
Kappa	0.91	0.97	0.55	1.00	0.87	1.00	1.00	1.00	0.98

Table 9: Continue

Balancing method (Feature selection /Classifiers (Estimators))	Proposed model on apendicitis dta set								
	Original	ROSE				SMOTE			
		CFS	RF	FRFS	PCA	CFS	RF	FRFS	PCA
FPR	0.04	0.00	0.00	0.00	0.04	0.00	0.00	0.00	0.00
FNR	0.03	0.05	0.6	0.00	0.10	0.00	0.00	0.00	0.03
J48									
Precision	0.86	0.91	0.83	1.00	0.98	0.96	0.98	0.98	1.00
NPV	0.63	0.96	1.00	0.96	1.00	0.98	1.00	0.98	1.00
Recall	0.89	0.91	1.00	0.95	1.00	0.96	1.00	0.98	1.00
F-measure	0.88	0.91	0.91	0.98	0.99	0.96	0.99	0.98	1.00
AUC	0.79	0.97	0.99	1.00	0.99	0.96	0.98	0.97	1.00
Accuracy	81.03	94.29	98.25	97.87	99.14	97.56	98.57	98.06	100.00
Kappa	0.47	0.87	0.9	0.96	0.98	0.95	0.97	0.96	1.00
FPR	0.44	0.04	0.02	0	0.01	0.02	0.04	0.02	0.00
FNR	0.11	0.09	0	0.05	0	0.04	0	0.02	0.00
Rep Tree									
Precision	0.9	0.90	1.00	0.88	1.00	0.96	1.00	0.98	0.90
NPV	0.73	0.92	1.00	0.92	0.99	1.00	0.95	1.00	0.94
Recall	0.92	0.82	1.00	0.90	0.98	1.00	0.96	1.00	0.92
F-measure	0.91	0.86	1.00	0.89	0.99	0.98	0.98	0.99	0.91
AUC	0.87	0.89	1.00	0.94	0.99	0.99	0.98	0.98	0.94
Accuracy	86.15	91.43	100.00	90.43	99.14	98.78	97.86	99.03	91.86
Kappa	0.62	0.80	1.00	0.81	0.98	0.97	0.96	0.98	0.84
FPR	0.31	0.04	0.00	0.09	0.00	0.02	0.00	0.02	0.09
FNR	0.08	0.18	0	0.1	0.02	0.00	0.04	0.00	0.08
Ripper									
Precision	0.93	0.87	1.00	0.97	1.00	0.96	0.97	0.98	1.00
NPV	0.76	0.96	0.98	0.91	0.99	0.98	0.98	0.98	0.98
Recall	0.92	0.91	0.8	0.88	0.98	0.96	0.99	0.98	0.97
F-measure	0.92	0.89	0.89	0.92	0.99	0.96	0.98	0.98	0.99
AUC	0.85	0.94	0.99	0.94	0.99	0.96	0.97	0.97	0.99
Accuracy	88.21	92.86	98.25	93.62	99.14	97.56	97.14	98.06	98.84
Kappa	0.68	0.84	0.88	0.87	0.98	0.95	0.94	0.96	0.98
FPR	0.23	0.06	0	0.02	0	0.02	0.06	0.02	0.00
FNR	0.08	0.09	0.2	0.12	0.02	0.04	0.01	0.02	0.03
Part									
Precision	0.86	0.95	0.83	1.00	0.98	0.96	0.98	0.98	1.00
NPV	0.61	0.94	1.00	0.96	1.00	0.98	1.00	0.98	1.00
Recall	0.88	0.86	1.00	0.95	1.00	0.96	1.00	0.98	1.00
F-measure	0.87	0.91	0.91	0.98	0.99	0.96	0.99	0.98	1.00
AUC	0.8	0.96	0.99	1.00	0.99	0.96	0.98	0.97	1.00
Accuracy	80.00	94.29	98.25	97.87	99.14	97.56	98.57	98.06	100.00
Kappa	0.44	0.86	0.90	0.96	0.98	0.95	0.97	0.96	1.00
FPR	0.46	0.02	0.02	0.00	0.01	0.02	0.04	0.02	0.00
FNR	0.12	0.14	0.00	0.05	0.00	0.04	0.00	0.02	0.00
Multilayer perceptron									
Precision	0.95	0.96	1.00	1.00	1.00	1.00	1.00	1.00	1.00
NPV	0.8	0.98	1.00	1.00	1.00	1.00	1.00	1.00	1.00
Recall	0.93	0.96	1.00	1.00	1.00	1.00	1.00	1.00	1.00
F-measure	0.94	0.96	1.00	1.00	1.00	1.00	1.00	1.00	1.00
AUC	0.96	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
Accuracy	90.77	97.14	100.00	100.00	100.00	100.00	100.00	100.00	100.00
Kappa	0.75	0.94	1.00	1.00	1.00	1.00	1.00	1.00	1.00
FPR	0.17	0.02	0.00	0.00	0.00	0.00	0.00	0.00	0.00
FNR	0.07	0.05	0.00	0.00	0.00	0.00	0.00	0.00	0.00
RBF									
Precision	0.85	1.00	1.00	1.00	0.98	0.96	0.98	0.98	0.98
NPV	0.75	1.00	0.98	1.00	0.97	1.00	1.00	1.00	1.00
Recall	0.95	1.00	0.80	1.00	0.95	1.00	1.00	1.00	1.00
F-measure	0.90	1.00	0.89	1.00	0.96	0.98	0.99	0.99	0.99
AUC	0.86	1.00	0.81	1.00	0.99	1.00	1.00	1.00	1.00
Accuracy	83.59	100.00	98.25	100.00	97.41	98.78	98.57	99.03	98.84
Kappa	0.50	1	0.88	1.00	0.94	0.97	0.97	0.98	0.98
FPR	0.50	0	0.00	0.00	0.01	0.02	0.04	0.02	0.02
FNR	0.05	0	0.20	0.00	0.05	0.00	0.00	0.00	0.00

Table 10: Scatter plot for Parkinson dataset using Rose Balancing and Tomek link

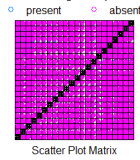
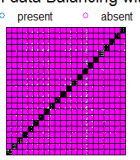
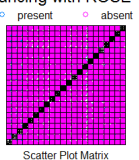
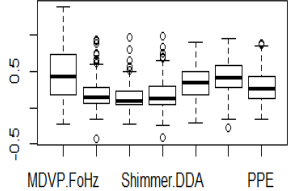
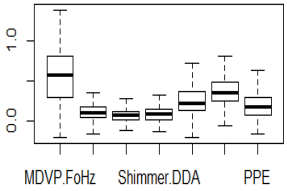
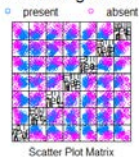
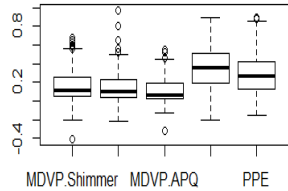
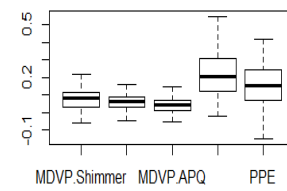
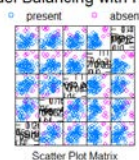
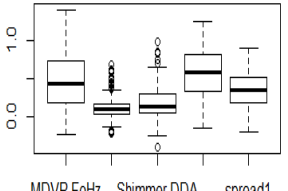
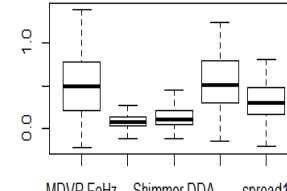
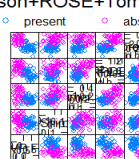
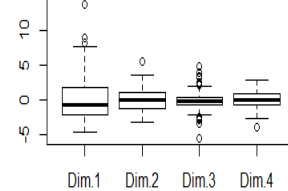
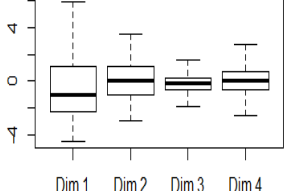
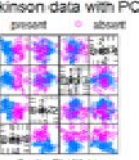
Classes	Original Data	Data after Balancing using smote method	Data with Tomek link removed
	<p>Scatter plot of Original parkinson data</p> 	<p>parkinson data Balancing with ROSE</p> 	<p>kinson Balancing with ROSE + Tomek L</p> 
CFS	<p>Before outlier elimination</p> 	<p>After outlier elimination</p> 	<p>Parkinson data with class imbalance</p> 
RF	<p>Before outlier elimination</p> 	<p>After outlier elimination</p> 	<p>Parkinson data with class imbalance</p> 
FRFS	<p>Before outlier elimination</p> 	<p>After outlier elimination</p> 	<p>Parkinson data with class imbalance</p> 
PCA	<p>Before outlier elimination</p> 	<p>After outlier elimination</p> 	<p>Parkinson data with class imbalance</p> 

Table 11: Scatter plot for Parkinson dataset using Smote Balancing and Tomek link

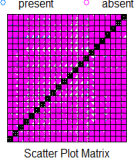
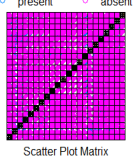
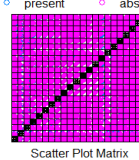
Classes	Original Data	Data after Balancing using smote method	Data with Tomek link removed
	<p>Scatter plot of Original parkinson data</p> 	<p>parkinson data Balancing with smote</p> 	<p>Parkinson smote and Tomek Link</p> 
CFS	<p>Before outlier elimination</p>	<p>After outlier elimination</p>	<p>Parkinson data with class imbalance and overlap eliminated using proposed model with CSF feature selection</p>

Table 11: Continue

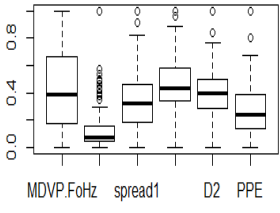
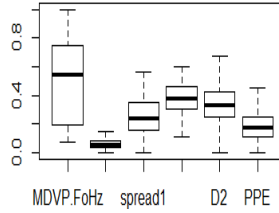
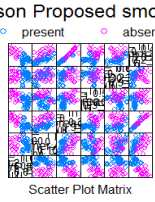
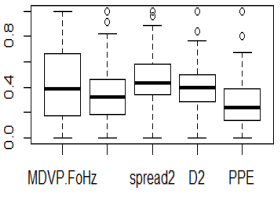
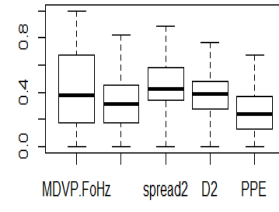
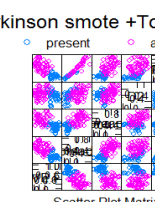
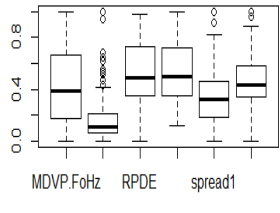
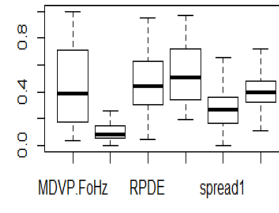
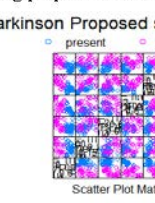
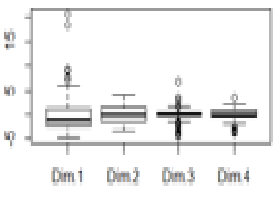
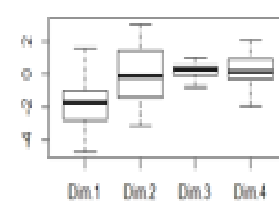

Classes	Original Data	Data after Balancing using smote method	Data with Tomek link removed
RF	 <p>Before outlier elimination</p>	 <p>After outlier elimination</p>	 <p>parkinson Proposed smote+CFS Scatter Plot Matrix</p> <p>Parkinson data with class imbalance and overlap eliminated using proposed model with Fuzzy Rough set</p>
	 <p>Before outlier elimination</p>	 <p>After outlier elimination</p>	 <p>parkinson smote +Tomek + RF Scatter Plot Matrix</p> <p>Parkinson data with class imbalance and overlap eliminated using proposed model with Fuzzy Rough set</p>
FRFS	 <p>Before outlier elimination</p>	 <p>After outlier elimination</p>	 <p>parkinson Proposed smote+FRFS Scatter Plot Matrix</p> <p>Parkinson data with class imbalance and overlap eliminated using proposed model with Fuzzy Rough set</p>
	 <p>Before outlier elimination</p>	 <p>After outlier elimination</p>	 <p>parkinsonSmote+Tomek+PCA Scatter Plot Matrix</p> <p>Parkinson data with class imbalance and overlap eliminated using proposed model with Fuzzy Rough set</p>

Table 12: Summary of preprocessing Phase1: Balancing phase

Dataset	No. of missing values	Replacement method	Normalize method	Balancing method	Total instances	No. of instances		Proportion		Instances removed from majority class using Tomek link
						Majority class	Minority class	Majority class	Minority class	
Heart	4	WtKnn	minmax	Rose	177	90	87	50.85	49.15	5
				smote	182	91	91	50.00	50.00	0
Appendicitis	0			Rose	106	55	51	51.89	48.11	12
				smote	126	63	63	50.00	50.00	3
				Rose	195	99	96	50.77	49.23	23
				smote	192	96	96	50.00	50.00	4

Table 13: Summary of preprocessing Phase2: Discarding overlapping Phase

Dataset	Feature selection						Feature Extraction		
	CFS		Random Forest		Fuzzy Rough Set		PCA		
	Rose	Smote	Rose	Smote	Rose	Smote	Rose	Smote	
Heart									
No of attributes	8	8	5	5	4	5	5	5	
No of instances after outlier removal using Boxplot	158	172	168	172	168	182	158	129	
No of instances after K-means elimination	148	162	153	148	147	153	140	116	
Instances after SVM optimization	147	162	152	148	147	153	140	116	

Table 13: Continue

Dataset	Feature selection						Feature Extraction	
	CFS		Random Forest		Fuzzy Rough Set		PCA	
	Rose	Smote	Rose	Smote	Rose	Smote	Rose	Smote
Appendicitis								
No of attributes	6	4	5	3	5	7	2	2
No of instances after outlier removal using Boxplot	82	107	85	123	94	90	79	116
No of instances after K-means elimination	63	83	66	92	65	69	63	98
Instances after SVM optimization	63	81	66	92	63	66	63	95
Parkinsons								
No. of attributes	7	8	5	5	5	6	4	4
No. of instances after outlier removal using Boxplot	102	121	80	183	133	143	144	130
No. of instances after K-means elimination	72	82	58	143	96	103	119	86
Instances after SVM optimization	70	82	57	140	94	103	116	86
Comparison with Benchmark								

ROC: Receiver Operating Characteristic (ROC) or ROC curve is a graphical plot used to visualize the performance of a binary classifier. In a ROC curve the true positive rate (Sensitivity) is plotted on the y axis and the false positive rate (100-Specificity) plotted on X axis for different cut-off points. The value nearing to 1 means the model is better.

Precision: Precision also known as positive predictive values PPV is the proportion of the predicted positive cases that were correct:

$$PPV = TP/(TP+FP)$$

Recall: The recall or True positive rate (TP) is the proportion of positive cases that were correctly identified:

$$Recall = TP/(TP+FN)$$

F-measure: Is the harmonic mean of precision and recall:

$$F\text{-measure} = (2 \times \text{precision} \times \text{recall}) / (\text{precision} + \text{recall})$$

False positive rate: The False Positive (FP) rate also known as Type I error is the proportion of negative cases that were incorrectly classified as positive:

$$FPR = FP/(FP+TN)$$

False negative rate: The False Negative (FN) rate also known as Type II error is the proportion of positive cases that were incorrectly classified as negative:

$$FNR = FN/(FN+TP)$$

Retaining the number of PC: All the components extracted using PCA is not meaningful only few

components will account for meaningful amounts of variance and the other components will tend to account for only trivial variance. Therefore, according to Kaiser retain the components with eigenvalues >1.00.

RESULTS AND DISCUSSION

Research findings:

- The proposed model was successful in overcoming imbalance class proportion using Smote and ROSE methods thus avoiding the classifier getting biased towards the majority class
- Elimination of class overlap makes the two classes linearly separable and well-defined clusters were formed
- The proposed model was also successful in reducing the False Positive and False Negative which is a must in medical field as it is a question of life of an individual
- Results from table conclude that performance of the classifier depends on the level of class overlap; decrease in overlap increased the performance of the classifier
- When the dataset is balanced and class overlapping is eliminated performance of all the classifiers are almost similar
- Results with Smote balancing method were better when compared to results with Rose balancing method
- Among the proposed models the model with Smote balancing and feature extraction using PCA showed a better performance when compared to Smote with feature selection methods CFS, random forest or fuzzy rough set
- The accuracy of the classifier depends on the type of feature selection or feature extraction method and the balancing method

CONCLUSION

Usually problems associated with data accumulated in different domains are always imbalanced which degrades the performance of the classifier. Class imbalance by itself is not a crucial problem but in combination with class overlap and high dimensional data makes it is a crucial problem and is the cause for the degradation of the classifier performance. A model was proposed to overcome both the issues in which Smote and ROSE were used to balance the dataset. The proposed model using Smote showed better results than ROSE as in some cases ROSE develops small disjuncts. To overcome class overlap among the three schemes discarding, merging and separating as explained in Xiong, discarding scheme was adapted. Though this model was successful in removing the instances in overlap region there is an information loss because instances in the overlap region contains information about different classes hence it has to be separately studied. Results prove that in all the cases pre-processing is necessary to improve the performance of the learning algorithms.

Performance of the classifier in the proposed model depends on the level of class overlap therefore the future work will make an attempt to test the proposed model on datasets in other domains. Further, among the three schemes discarding, merging and separating to overcome class overlap problem, discarding is adapted in this work since the instances in overlap region contains information about both the classes, in future along with studying about the instances in the overlap region other schemes like merging and separating should be applied and the same should be tested on the datasets of other domains. Since K-means recognizes only spherical shaped clusters future work should be done using other clustering algorithms for the removal of class overlap.

APPENDIX

Appendix 1: Comparison with Benchmark

Appendicitis dataset	Accuracy	References
SSV beam leaves (2010)	88.7±8.5	Maszczyk and Duch
SVM linear C = 1	88.1±8.6	
SSV default	87.8±8.7	
SSV beam pruning	86.9±9.8	
kNN	86.7±6.6	
SVM Gauss	84.4±8.2	
NO TSS	87.82	Verbiest <i>et al.</i> (2016)
GGA	87.32	
CHC	83.36	
SSGA	87.95	
RMHC	79.77	
FRPS	87.82	

Appendix 1: Continue

Appendicitis dataset	Accuracy	References	
Proposed model (Rose, Smote)			
Highest accuracy using	100		
Highest accuracy using	100		
Proposed model	100		
Cleveland-0_vs_4(Heart)			
Highest accuracy using	100		
	100		
Appendicitis dataset	AUC	Reference	
kNN SMOTE (2015)	0.79±0.12	Beyan and Fisher	
kNN SMOTE wFS	0.78±0.08		
C4.5	0.55±0.07		
C4.5 SMOTE	0.59±0.01		
NB SMOTE	0.71±0.14		
SVM	0.89±0.09		
SVM wFS	0.75±0.16		
SVM SMOTE	0.83±0.14		
SVM SMOTE wFS	0.84±0.09		
RF BT	0.79±0.09		
RF BT wFS	0.84±0.16		
Proposed method with outlier detection parameter	0.96±0.05		
kNN SMOTE (2015)	0.64±0.19	Beyan and Fisher	
kNN SMOTE wFS	0.60±0.11		
C4.5	0.55±0.14		
C4.5 SMOTE	0.37±0.13		
NB SMOTE	0.80±0.18		
SVM	0.91±0.14		
SVM wFS	0.70±0.21		
SVM SMOTE	0.89±0.17		
SVM SMOTE wFS	0.53±0.35		
RF BT	0.84±0.12		
RF BT wFS	0.62±0.17		
Proposed method with outlier detection parameter	0.91±0.09		
SMOTE Bagging			
Base	0.7894	Galar <i>et al</i> (2016)	
BB	0.7933		
BB-Imb	0.8004		
MDM	0.7815		
MDM-Imb	0.7835		
Under Bagging			
Std.	0.8492	Galar <i>et al</i> (2016)	
BB	0.8714		
BB-Imb		0.8305	
MDM		0.7917	
MDM-Imb		0.8069	
Parkinsons	F-measure	Accuracy	Reference
AdaBoost (Target) (2015)	0.404	0.649	Al-Stouhi and Reday
AdaBoost (Src+Tar)	0.504	0.659	
SMOTE (Target)	0.552	0.761	
SMOTE (Src+Tar)	0.733	0.762	
TrAda Boost	0.702	0.862	
Rare Transfer	0.749	0.885	

REFERENCES

Alcala-Fdez, J., A. Fernandez, J. Luengo, J. Derrac, S. Garcia, L. Sanchez and F. Herrera, 2011. KEEL data-mining software tool: Data set repository, integration of algorithms and experimental analysis framework. *J. Multiple-Valued Logic Soft Comput.*, 17: 255-287.

- Ali, A., S.M. Shamsuddin and A.L. Ralescu, 2015. Classification with class imbalance problem: A review. *Int. J. Adv. Soft Comput. Appl.*, 7: 176-204.
- Batista, G.E.A.P.A., R.C. Prati and M.C. Monard, 2004. A study of the behavior of several methods for balancing machine learning training data. *ACM SIGKDD Explorations Newsl.*, 6: 20-29.
- Beyan, C. and R. Fisher, 2015. Classifying imbalanced data sets using similarity based hierarchical decomposition. *Pattern Recognit.*, 48: 1653-1672.
- Chawla, N.V., K.W. Bowyer, L.O. Hall and W.P. Kegelmeyer, 2002. SMOTE: Synthetic minority Over-sampling technique. *J. Artificial Intell. Res.*, 16: 321-357.
- Das, B., N.C. Krishnan and D.J. Cook, 2014. Handling Imbalanced and Overlapping Classes in Smart Environments Prompting Dataset. In: *Data Mining for Service*, Katsutoshi, Y. (Ed.). Springer, Heidelberg, Germany, ISBN:978-3-642-45251-2, pp: 199-219.
- Denil, M. and T. Trappenberg, 2010. Overlap versus imbalance. *Proceeding of the 23rd Canadian Conference on Artificial Intelligence*, May 31-June 2, 2010, Springer, Heidelberg, Germany, ISBN: 978-3-642-13058-8, pp: 220-231.
- Elrahman, S.M.A. and A. Abraham, 2013. A review of class imbalance problem. *J. Netw. Innovative Comput.*, 1: 332-340.
- Galar, M., A. Fernandez, E. Barrenechea, H. Bustince and F. Herrera, 2016. New ordering-based pruning metrics for ensembles of classifiers in imbalanced datasets. *Proceedings of the 9th International Conference on Computer Recognition Systems CORES 2015*, March 1-5, 2016, Springer, New York, USA., ISBN: 978-3-319-26225-3, pp: 3-15.
- Ganganwar, V., 2012. An overview of classification algorithms for imbalanced datasets. *Int. J. Emerging Technol. Adv. Eng.*, 2: 42-47.
- Garcia, V., J. Sanchez and R. Mollineda, 2007. An Empirical Study of the Behavior of Classifiers on Imbalanced and Overlapped Data Sets. In: *Progress in Pattern Recognition, Image Analysis and Applications*, Luis, R., D. Mery and K. Josef (Eds.). Springer, Heidelberg, Germany, ISBN:978-3-540-76724-4, pp: 397-406.
- Garcia, V., R. Alejo, J.S. Sanchez, J.M. Sotoca and R.A. Mollineda, 2006. Combined Effects of Class Imbalance and Class Overlap on Instance-Based Classification. In: *Intelligent Data Engineering and Automated Learning*, Emilio, C., Y. Hujun, B. Vicente and F. Colin (Eds.). Springer, Heidelberg, Germany, ISBN:978-3-540-45485-4, pp: 371-378.
- Gu, Q., Z. Cai and L. Zhu, 1986. Classification of imbalanced data sets by using the hybrid re-sampling algorithm based on isomap. *Proceeding of the 4th International Symposium on Intelligence Computation and Applications*, October 23-25, 2009, ISICA, Huangshi, China, ISBN:978-3-642-04842-5, pp: 287-pp: 24.
- Guo, H., W. Zhi, H. Liu and M. Xu, 2016. Imbalanced learning based on logistic discrimination. *Comput. Intell. Neurosci.*, Vol. 2016, 10.1155/2016/5423204
- Hall, M.A., 2000. Correlation-based feature selection for discrete and numeric class machine learning. *Proceedings of the 17th International Conference on Machine Learning*, 29 June-July 2, 2000, California, pp: 359-366.
- He, H. and E.A. Garcia, 2009. Learning from imbalanced data. *IEEE Trans. Knowledge Data Eng.*, 21: 1263-1284.
- Jain, A.K., 2010. Data clustering: 50 years beyond K-means. *Pattern Recogn. Lett.*, 31: 651-666.
- Japkowicz, N. and S. Stephen, 2002. The class imbalance problem: A systematic study. *Intell. Data Anal.*, 6: 429-449.
- Japkowicz, N., 2003. Class imbalances: Are we focusing on the right issue. *Workshop Learn. Imbalanced Data Sets II.*, 1723: 17-23.
- Jensen, R., 2005. Combining rough and fuzzy sets for feature selection. PhD Thesis, University of Edinburgh, Edinburgh, Scotland. <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.101.7352&rep=rep1&type=pdf>
- Kaveri, S. and Abhilasha, 2016. A study on effects of intrinsic characteristics of datasets on classification performance. *J. Adv. Res. Comput. Sci. Software Eng.*, 6: 198-204.
- Lin, W.J. and J.J. Chen, 2012. Class-imbalanced classifiers for high-dimensional data. *Briefings Bioinformatics*, 14: 13-26.
- Liu, C.L., 2008. Partial discriminative training for classification of overlapping classes in document analysis. *Int. J. Doc. Anal. Recognit.*, 11: 53-65.
- Maszczyk, T. and W. Duch, 2010. Support feature machines: Support vectors are not enough. *Proceeding of the 2010 International Joint Conference on Neural Networks (IJCNN)*, July 18-23, 2010, IEEE, New York, USA., ISBN:978-1-4244-6918-5, pp: 1-8.
- Menardi, G. and N. Torelli, 2014. Training and assessing classification rules with imbalanced data. *Data Min. Knowl. Discovery*, 28: 92-122.
- Perez, O.M., P.A. Gutierrez, P. Tino and M.C. Hervas, 2015. Oversampling the minority class in the feature space. *IEEE Trans. Neural Netw. Learn. Syst.*, 27: 1947-1961.

- Prati, R.C., G.E. Batista and M.C. Monard, 2004. Class imbalances versus class overlapping: An analysis of a learning system behavior. Proceeding of the 3rd Mexican International Conference on Artificial Intelligence, April 26-30, 2004, Springer, Heidelberg, Germany Berlin, ISBN:978-3-540-21459-5, pp: 312-321.
- Rahman, M.M. and D.N. Davis, 2013. Addressing the class imbalance problem in medical datasets. *Int. J. Mach. Learn. Comput.*, 3: 224-224.
- Saranya, C. and G. Manikandan, 2013. A study on normalization techniques for privacy preserving data mining. *Int. J. Eng. Technol.*, 5: 2701-2704.
- Stouhi, A.S. and C.K. Reddy, 2015. Transfer learning for class imbalance problems with inadequate data. *Knowl. Inf. Syst.*, 48: 201-228.
- Sumana B.V. and T. Santhanam, 2016. Optimizing K-means in cascading clustering and classification. *Aust. J. Basic Appl. Sci.*, 10: 184-206.
- Verbiest, N., J. Derrac, C. Cornelis, S. Garcia and F. Herrera, 2016. Evolutionary wrapper approaches for training set selection as preprocessing mechanism for support vector machines: Experimental evaluation and support vector analysis. *Appl. Soft Comput.*, 38: 10-22.
- Weiss, G.M., 2004. Mining with rarity: A unifying framework. *ACM SIGKDD Explorations Newsl.*, 6: 7-19.