SUBSET FEATURE SELECTION APPROACH FOR CLASS IMBALANCE

Thesis submitted in partial fulfillment of the requirements for the award of degree of

Master of Engineering

in

Software Engineering

Submitted By

Pawan Lachheta

(Roll No. 801431018)

Under the supervision of:

Dr. Seema Bawa

Professor

Thapar University, Patiala

COMPUTER SCIENCE AND ENGINEERING DEPARTMENT

THAPAR UNIVERSITY

PATIALA – 147004

June 2016

Certificate

I hereby certify that the work which is being presented in the thesis entitled, "*Subset Feature Selection Approach For Class Imbalance*", in partial fulfillment of the requirements for the award of degree of Master of Engineering in *Software Engineering* submitted in Computer Science and Engineering Department of Thapar University, Patiala, is an authentic record of my own work carried out under the supervision of *Dr. Seema Bawa* and refers other researcher's work which are duly listed in the reference section.

The matter presented in the thesis has not been submitted for award of any other degree of this or any other University.

(Pawan Lachheta)

This is to certify that the above statement made by the candidate is correct and true to the best of my knowledge.

03 June 2016

(Dr. Seema Bawa)
Professor, CSED

Countersigned by

13/6/16

(Dr. Deepak Garg)
Head
Computer Science and Engineering Department
Thapar University
Patiala

(Dr. S. S. Bhatia)
Dean (Academic Affairs)
Thapar University
Patiala

Abstract

In machine learning, building an effective classification model, when the high dimensional data is suffering from class imbalance problem, is a major challenge. The problem becomes severe when negative samples have large percentages than positive samples. Various techniques like cost sensitive learning techniques, recognition based techniques, and sampling based techniques, etc. exist to handle data imbalance problem. However, these techniques suffer from data loss and over fitting because they invariably change the original distribution of data. To surmount the data imbalance and high dimensionality issues in dataset, in this thesis we propose a framework named Subset Feature Selection (SFS). The proposed SFS framework comprises of SMOTE filters are used for balancing the datasets, as well as feature ranker for pre-processing of data. The framework SFS is developed using R language and various R packages. The performance of SFS framework is evaluated and results show that SFS framework outperforms than other existing techniques like cost sensitive learning, recognition based techniques etc.

Acknowledgement

First of all I would like to thank the Almighty, who has always guided me to work on the right path of the life. It is a great privilege to express my gratitude and admiration towards my respected supervisor **Dr. Seema Bawa** Professor Computer Science & Engineering Department. She has been an esteemed guide and great support behind achieving this task. This work would not have been possible without the encouragement and able guidance of her. I also thank my supervisor for her time, patience, discussions and valuable comments. Her enthusiasm and optimism made this experience both rewarding and enjoyable. I am truly grateful to her for extending her total co-operation and understanding whenever I needed help and guidance from her. I am also heartily thankful to **Dr. Deepak Garg**, Associate Professor and Head, Computer Science & Engineering Department and **Dr. Rupali Bhardwaj**, PG coordinator, for motivation and providing uncanny guidance and support throughout the preparation of the thesis report.

I will be failing in my duty if I do not express my gratitude to **Dr. S. S. Bhatia**, Senior Professor and Dean of Academic Affairs, for making provisions of infrastructure such as library facilities, computer labs equipped with net facilities, immensely useful for the learners to equip themselves with the latest in the field.

I am also thankful to Nishtha Hooda, PHD Scholar, the entire faculty and staff members of Computer Science and Engineering Department for their direct-indirect help, cooperation, love and affection, which made my stay at Thapar University memorable. Last but not least, I would like to thank my family for their wonderful love and encouragement, without their blessings none of this would have been possible.

Pawan Lachheta

(**801431018)**

Table of Contents

List of Figures

List of Tables

Chapter 1: Introduction

This chapter tells about class imbalance problem and methods, techniques to deal with high dimensionality and class imbalance problem.

1.1 Class Imbalance

The class imbalance is a crucial problem in machine learning, which has become an emerging research area in recent years. Classification algorithms afflicted by the class imbalance problem for a dataset would see strong overall accuracy but very low performance on the positive samples (minority class) [4]. For example if we are working on our dataset and we create classification model and get 99% accuracy immediately [32]. We think that it is fantastic but if we plunge a slight deeper and discover that 99% of the data belongs to only one class. This is an example of an imbalance problem and it can cause the frustrating results.

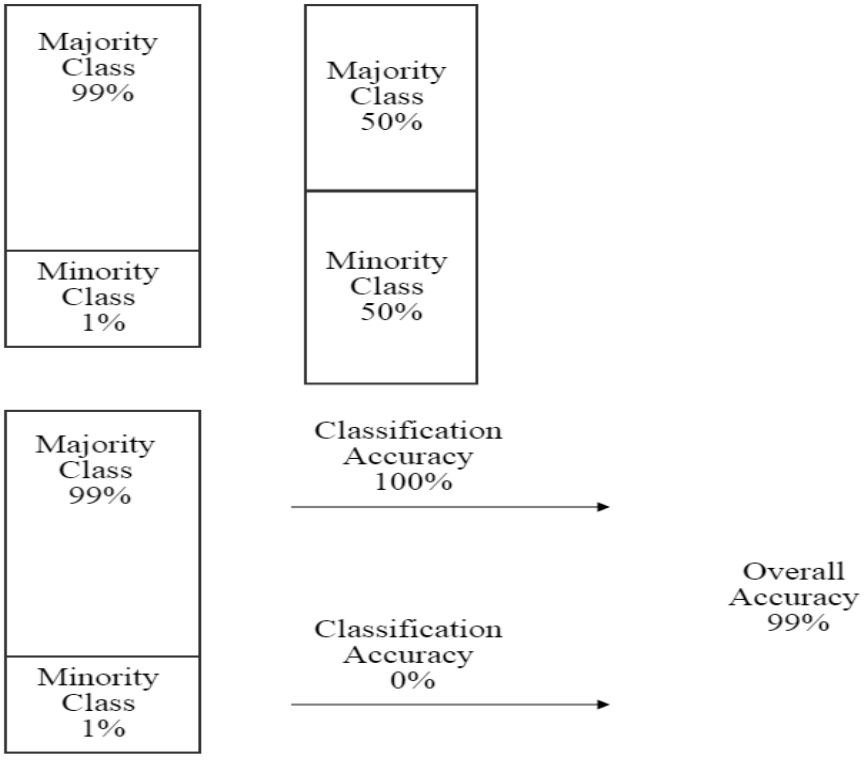


Figure 1.1 Class Imbalance Problem

Imbalanced dataset is a dataset where the classes are not represented equally. Imbalanced dataset exists in many real-world [51] areas like oil spills detection from satellite images, fraud detection, anomaly detection, software defect prediction, text classification, medical diagnosis, image annotation and identifying fraudulent credit card transactions [37] etc, which is composed with large percentage of negative samples and fewer percentage of positive samples because of which such classification algorithms works very well for negative samples but downgrade in case of positive samples but their accuracy imply very good.

Therefore, researchers have given more focus to high dimensionality issues and class imbalance problems and several conferences and workshops were held [34]. For Binary or two class problem, samples are divided into two classes namely minority and majority class. As most traditional machine learning algorithms, such as RIPPER, KNN, generate models which is ignore the minority class and maximize overall classification accuracy.

For example, for a data set where only 1% instances belong to positive samples, still the accuracy of the classifier will be 99% when the classifier classifies all the instances of negative samples (majority class) [40].

Most classification works best when the numbers of features of each class are equal. When features of one class exceeds then other, problems arise. There are various problems occurring by imbalance dataset and they affect the performance of classification algorithms [15].

1.2 Loss of Data

Another problem with imbalanced dataset is lack of data. For building good classifier, size of data plays very important role. Figure 1.2 (a) shows the decision boundary obtained during training, whereas Figure 1.2 (b) shows the result when small number of samples is used [5]. The dashed line represents estimated decision boundary and solid line represents actual decision boundary.

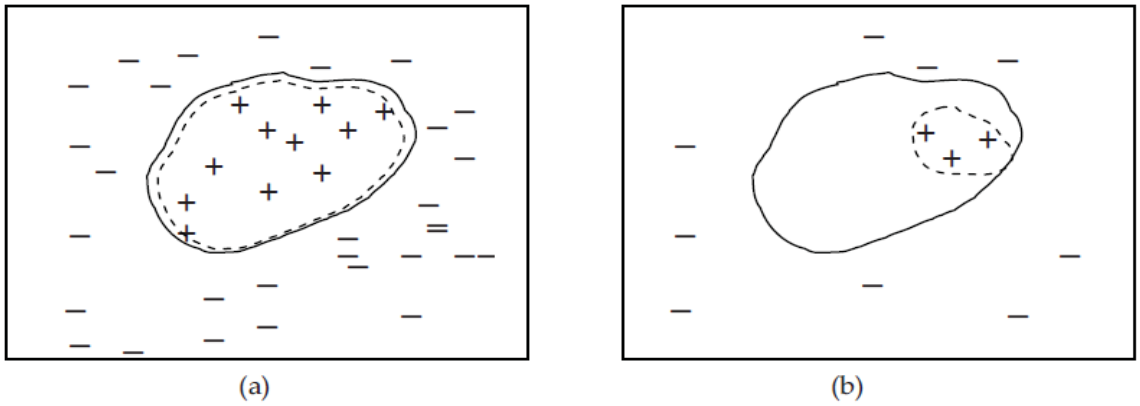


Figure 1.2 the Effect of Loss of Data on Class Imbalance [5]

In that respect there are many challenges arise with imbalanced datasets. The major challenge with imbalance dataset is measurement of performance. Evaluation parameters like Accuracy, AUC, F-measure, etc are known to play a vital role in machine learning. Therefore, if evaluation parameter does not include the positive samples, the classification algorithms will not cope with data imbalance problem.

For this intent, many techniques have been proposed for bringing modification to the imbalanced dataset, such as the over-sampling technique over the positive samples, the under-sampling technique over the negative samples variable or feature selection, cost sensitive learning methods, recognition based method, ensemble based methods, synthetic minority oversampling technique, etc.

1.3 Existing Methods

Many Methods have been proposed for solutions of data imbalance problem, few of them include sampling, variable selection, cost sensitive, recognition based and ensemble based methods etc.

1.3.1 Under-Sampling

Here instead of increasing the training set we bring down the size of training position by removing those classes of data which are higher in number [12]. The intent of under-sampling is to balance the class by removing negative samples.

To train the classifier, this approach uses negative samples. Still, the principal obstacle is data loss in ignored samples [12].

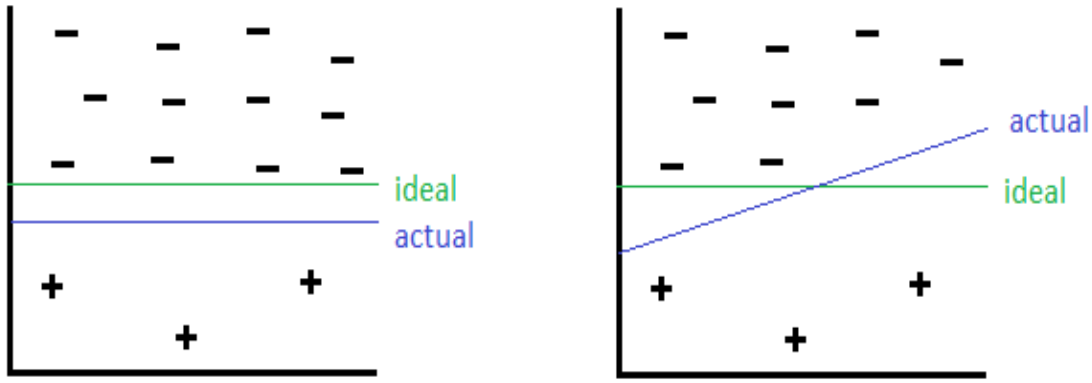


Figure 1.3 Under-Sampling [12]

In above diagram green line shows the ideal decision boundary and blue shows actual result [12]. Left side shows the result of applying classification algorithms without under-sampling and right side shows, under-sampling of majority class but removed some majority samples.

1.3.2 Over-Sampling

Over-sampling is used for balancing class distribution using random copy of positive samples such that the positive samples oversampled, that balance the positive samples within dataset [12]. Nevertheless, this method adds up with a problem that it replicates existing samples in the positive samples, which may cause over fitting.

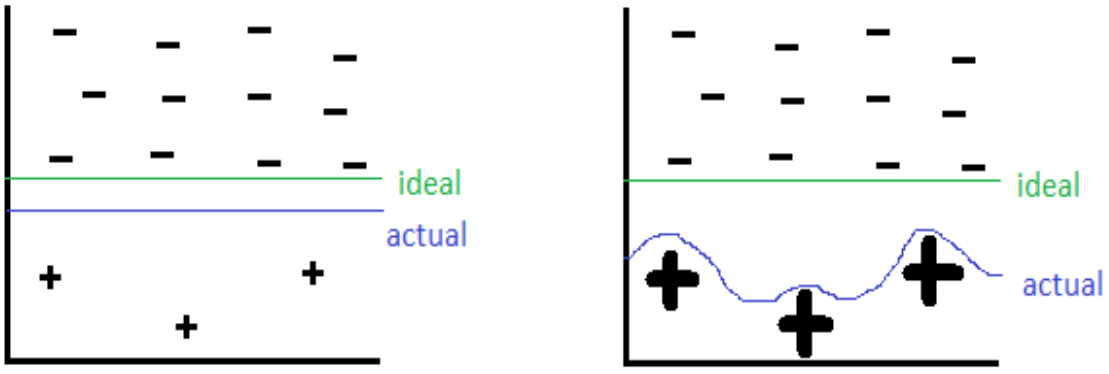


Figure 1.4 Over-Sampling [12]

In above diagram green line shows the ideal decision boundary and blue shows actual result [12]. Left side shows the result of applying classification algorithms without oversampling. Right side shows oversampling of positive class but it can cause over fitting.

1.3.3 Synthetic Minority Over-Sampling Technique

Chawla (2002) proposed SMOTE technique. It uses concept of over-sampling technique in which the positive samples are oversampled by creating synthetic samples of existing positive samples [30]. This creates new minority class instances by:

- For each minority class instance C
  - Neighbour's=Get KNN(5)
  - N=Random pick one from neighbour's
  - Create a new minority class R instance using C's feature vector and the feature vector's difference of N and C multiplied by a random number

    i.e. R. feats = C. feats + (C. feats - N. feats) * rand (0, 1)

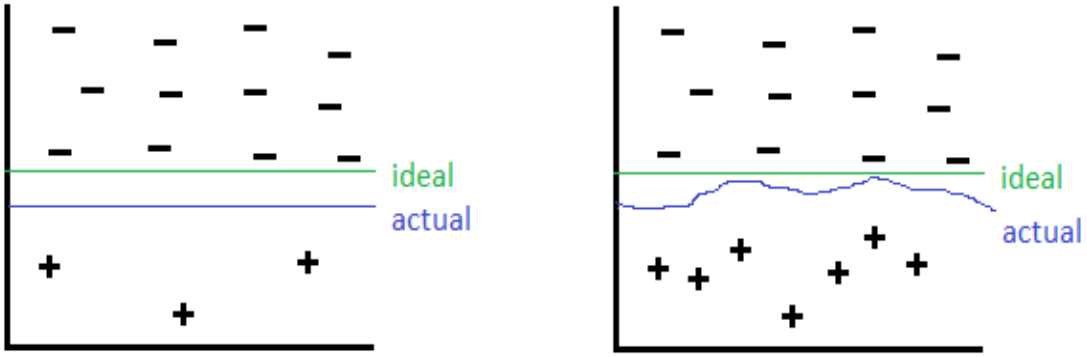


Figure 1.5 SMOTE [12]

In above diagram green line shows the ideal decision boundary and blue shows actual results. Left side shows the result of applying classification algorithms without smote.

1.3.4 Cost Sensitive Learning Methods

In many class imbalance problems misclassification error cost is uneven. Cost sensitive learning methods assign higher cost to minority class and generate lower cost model.

i.e. C(+,-)> C(-,+), but these methods may lead to problem of over fitting [48]. Yang et al. [6] presented SVM based cost sensitive learning approaches that alter margins to achieve unbiased decision boundary.

1.3.5 Recognition Based Methods

Recognition based methods also named as one class learning. In recognition based method classification algorithms learn on minority class (positive samples) [35] [48]. These methods enhance performance of classification algorithms on unseen data. To deal with high dimensional noisy data and imbalanced data, recognition based method can be robust technique.

1.3.6 Ensemble Based Methods

Ensemble based method is the combination of various classification algorithms to enhance prediction accuracy and ability of generalization. Most popular ensemble based techniques are bagging and boosting [38] [48]. In boosting, classification algorithm uses previous one and also focuses on its errors, while bagging trains each classification algorithm by subset of training set.

1.3.7 Feature Selection

Feature selection is important step of removing irrelevant and redundant features for the model construction [39]. Feature selection includes and excludes attribute present in the data without any revisions [43] [46]. Feature selection acts as a filter, as it mutes out those characteristics that aren’t useful in addition to your existing features [45]. Feature selection enhances performance of classification algorithms. Feature selection methods are Filter method, Wrapper method, and embedded method [24] [41] [44]. These methods select those features which are best for performance of model. Filter method measure quality of important selected features, from machine learning algorithms, while wrapper methods needs application of classification algorithm to measure quality of selected features. For learning of optimal parameters, embedded methods perform feature selection [14] [31].

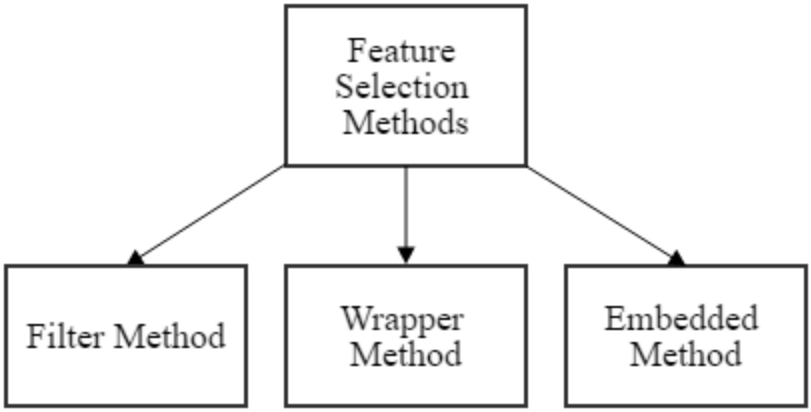


Figure 1.6 Feature Selection Methods

1.4 Evaluation Metrics

Evaluation metrics plays significant role in machine learning. They are employed to measure the learning algorithms. The commonly used metric for these intents is the accuracy. Yet, on an imbalanced data set, accuracy is not an appropriate metric, since the positive course of instruction has little effect on classification rate (accuracy) as compared to negative class hence; other evaluation parameter has been used to enhance performance of classification algorithms [33] [36] [47].

Table 1.1 Confusion Matrix [47]

		True Class	
		Positive	Negative
Prediction Class	Positive	True Positive (TP)	False Positive (FP)
	Negative	False Negative (FN)	True Negative (TN)

Table 1.2 Performance Measurement Parameter [47]

Name	Formula	Explanation
True Positive Rate (TP rate)	TP / (TP + FP)	The closer to 1, the better. TP rate = 1 when FP = 0. (No false positives)
True Negative Rate (TN rate)	TN / (TN + FN)	The closer to 1, the better. TN rate = 1 when FN = 0. (No false negatives)
False Positive Rate (FP rate)	FP / (FP + TN)	The closer to 0, the better. FP rate = 0 when FP = 0. (No false positives)
False Negative Rate (FN rate)	FN / (FN + TP)	The closer to 0, the better. FN rate = 0 when FN = 0. (No false negatives)

1.5 Performance Measurement Parameter

There are the many metric used for estimating performance of classifiers. For binary class problem some of them are defined below [47].

1.5.1 Accuracy

It is the ratio of correct prediction to number of instances evaluated.

$$Accuracy = (T\ P + T\ N) / (T\ P + F\ N + T\ N + F\ P) \quad \text{.....eq. 1}$$

1.5.2 Sensitivity

It measure fraction of positive patterns that are correctly classified.

$$Sensitivity = T\ P / (T\ P + F\ N) \quad \text{.....eq. 2}$$

1.5.3 Precision

It is the fraction of correctly predicted patterns to total predicted patterns in positive class.

$$Precision = T\ P / (T\ P + F\ P) \quad \text{.....eq. 3}$$

1.5.4 F-measure

It is the harmonic mean between recall and precision.

$$F\text{-measure} = (2 * Recall * Precision) / (Recall + Precision) \quad \text{....eq. 4}$$

1.5.5 Area under the Curve

It is ranking type metrics and reflects overall ranking performance of classification algorithm

$$AUC = Sp - Np (Nn + 1) / (2 / Np * Nn) \quad \text{eq. 5}$$

Where Sp is sum of all positive samples ranked, Np and Nn is no of positive and negative samples.

1.6 Classification Algorithms

There is much class of classification algorithms. Few of them are explain below [29].

1.6.1 Naïve Bayes

It is statistical classification algorithms which predict the probability of given instances. It follows Bayes's rule and assumes that variables are not dependent on each other in a given class called as conditional independence [29]. It has exhibited high performance for large databases.

1.6.2 Random Forest

It is a popular classification algorithm which builds randomized decision tree in bagging algorithm and produces excellent predictors [29].

1.6.3 AdaBoost

Adaptive boosting used to enhance the accuracy of classification algorithms. In adaptive boosting, weights are given to each training instances and after that classification algorithms is applied [29]. It is fast and can be accelerated by weight pruning.

1.6.4 SVM

SVM uses principle of risk minimization principle. This principle divides data or information into classes with maximum margin among classes [29] [42]. It is capable of learn in sparse, high dimensionality spaces with training samples to minimize error rate and complexity of classification algorithm.

Chapter 2: Literature Survey

In the following chapter, analysis is performed on various data imbalance methods and techniques. Below are abstracts of those analyses performed by various research fellows.

2.1 Class Imbalance and High Dimensionality

Zhongbin Sun *et al.* [2] presented an approach to deal with class imbalance problem. Different from other methods proposed method doesn't alter original data distribution and doesn't suffer from unexpected mistakes or important information loss. With the help of clustering or random splitting to majority class instances, proposed approach converts imbalanced binary class into balanced binary class. After that various machine learning algorithm is applied to balanced dataset to build various classifiers and their results are combined.

Cigdem Beyan *et al.* [3] presented a framework to cope with problem of class imbalance, which is different from other methods and doesn't require preprocessing step. Framework presented by researcher was based on outlier detection and clustering, in which outlier detection was used to find out positive class instances, while clustering was used to partitioning of data.

AnnaritaD'Addabbo *et al.* [1] presented an algorithm, called PSS (Parallel Selective Sampling) used for preprocessing to train SVM on large amount of data and imbalanced datasets. Presented algorithm was developed for parallel and distributed computing. PSS retrieve data from negative samples to reduce data imbalance problem in large datasets. It is the selective sampling method which is combined with various classification algorithms.

Shu Zhang *et al.* [4] performed different experiments on five UCI imbalanced datasets with the help of Decision Tree, Naïve Bayes and SVM. In this paper they have determined that SVM was impacted by cross validation, penalty C and kernel functions.

Marcelo Beckmann *et al.* [9] focused on an algorithm to adjust imbalanced datasets. Presented method was the combination of KNN and under-sampling that remove samples from majority class, remove noisy examples, reduce class overbalancing and also clean the decision surface.

Yang Liu *et al.* [6] introduced an approach to solve data imbalance problem. Researchers have solved this problem by making SVM classifiers and used the concept of under – sampling and over-sampling. They have solved class imbalance problem by using concept of parameter selection, which improve the performance of SVM classifiers.

Nitesh V. Chawla *et al.* [18] focused on three major issues pruning, preprocessing effect and quality of probabilistic estimates. They have considered each issue independently and highlight scenarios. Pruning was used for learning from imbalanced datasets and also helps to improve generalization of decision tree algorithm.

Haibo He *et al.* [17] in this paper they have provided review of data imbalance problem. Their focus was to provide comprehensive review of problem, technologies, assessment metrics used to measure performance of classifiers, also focus on major challenges and opportunities and important research directions.

Xueying Zhang *et al.* [19] presented dissimilarity based classification algorithm to cope with imbalanced data. This method firstly removes irrelevant and redundant features from dataset by using feature selection method to reduce impact on selection and transformation of prototype and extracts instances from each class as prototypes from reduced dataset and with the help of new features; projects reduced data into dissimilarity space.

2.2 Feature selection and Feature Ranking

Yok-Yen Nguwi, *et al.* [7] presented an approach which is the combination of emergent self-organizing approach and ranking method.

Researcher was used support vector for feature selection on the basis of feature s selection criteria. Approach depends on weight vector sensitivity and emergent self organizing map which was used for feature mapping that shows distance structure and density structure for high dimensional datasets.. By SVM and ESOM they have formed new hybrid approach, named as SVESOM.

Touraj Varaee *et al.* [14] presented hybrid variable selection approach which uses concept of wrapper technique along with lower cost and also improve performance of classifiers. Presented method was the combination of sample domain filtering; two feature selection method and re-sampling for refining of sample domain. Here, approach was divided into two phase, in first phase filters was used and in second phase they have used the concept of wrapper subset selection and genetic search. First phase refine and analyze the sample domain for better result in second phase. In second phase filtering technique eliminates irrelevant features and wrapper method selects relevant features with higher accuracy and lower cost.

Lin Lin *et al.* [15] presented an algorithm which uses multiple correspondence analyses to find relation between classes and features to reduce semantic gap and feature space. Algorithm presented by researcher was able to find correlation among items and class and features, which expands its ability to cope with class imbalance datasets. Here, presented algorithm handles multimedia semantic problems like semantic gap, high dimensionality, and class imbalance.

Xue-wen Chen *et al.* [16] presented feature selection approach, feature assessment by sliding threshold, that measure importance of features with the help of area under the ROC in one dimensional feature space using sliding decision line. By using Roc curve or rank features they have created another issues i.e. where to place threshold. Possible solution was to use histogram to find where threshold was placed.

Tian-Yu Liu *et al.* [21] presented a method called as mutual information based on feature selection for easy ensemble to deal with load balancing and improve performance of easy

ensemble classifier and compared with support vector machine and easy ensemble. Presented method improves prediction ability. They have used concept of mutual information to describe dependency between two random variable .It is also referred to relative entropy. Proposed method was train model on training set by easy ensemble and calculates mutual information on training set. After that it will select relevant features and ranking them and generate optimal training subset and retrain model on training subset.

George Forman et al. [24] focused on comparative study of feature selection method for class imbalance problem in text classification and also focused on binary class problem and support vector machine with skew class. In this paper they have presented evaluation method that selects more than one metrics to improve performance of datasets.

Isabelle Guyon et al. [25] presented overview of feature selection to increase performance of machine learning algorithm to cope with high dimensional and imbalanced dataset. Aim of feature selection is cost effective predictors, to enhance performance of classification algorithms, and for generated data provide better understanding. In this paper they have also focused on construction of feature, objective function, feature ranking and feature validity assessment methods.

Kehan Gao *et al.* [26] in this paper they have investigated a methodology of variable selection with ensemble learning process and examined 2 learning method and 5 feature selection techniques namely filter based feature ranking techniques i.e., chi squared, symmetrical uncertainty, information gain, wrapper method and embedded method.

Huan Liu *et al.* [31] focused on feature selection or variable selection techniques in machine learning. Variable selection is important step for the machine learning applications and it remove irrelevant and redundant features and also reduce dimension.

2.3 Clustering and Sampling Methods

David A *et al.* [10] presented a framework of local sampling which identifies regions of data and find efficient sampling level within data. Through this framework researcher was observed that how properties of data affect classifiers performance aside from skew class and different levels of sampling produce performance on datasets even skew class is identical. They have also noted that performance of classifiers also improved by global sampling levels. Here, firstly they have discovered segments within data and apply local sampling on each components.

Lara Lusa *et al.* [8] give technique to improve performance of classification models which was used concept of cross validation (CV) to evaluate models and includes sampling in cross validation loop. In cross validation approach dataset was parceled out into m folds, in which m-1 fold was used for building of prediction model and remaining was used for performance evaluation of prediction model. Researcher was used balanced folds, in which level of class imbalance and no of sample was equal. This process was repeated m times to improve performance of prediction model.

Son Lam Phung *et al.* [5] presented under-sampling technique, which is includes concept of clustering. Clustering is used to partition the training instances into set of training prototype patterns. After that, for addressing the class imbalance problem weight is given to each training prototype. For even class distribution they have used cost function and apply unsupervised learning and select cluster centers.

D. N. Davis *et al.* [11] presented modified cluster based sampling that balance the data and create better quality training set for classification model. Cluster based under-sampling choose best training sets from derived clusters.

Juanjuan Wang *et al.* [12] presented a framework to cope with problem of class imbalance. Proposed framework was improved SMOTE algorithm through LLE (locally linear embedding) algorithm.

Here, firstly LLE algorithm was used for low dimensional data in which data is separable and oversampled, with the help of SMOTE. After that SMOTE generate synthetic data points than these data points mapped back into original data space using locally linear embedding.

Xu-Ying Liu *et al.* [13] focused on two algorithms EasyEnsemble and BalanceCascade to overcome data imbalance deficiency. EasyEnsable samples subsets from negative samples (majority class), trains a learner and combines their outputs while BalanceCascade sequentially trains the learner. Main difference was that BalanceCascade uses trained classifier for sampling process and removes majority class sample in each iteration while EasyEnsable samples independent subsets. Both algorithms provide generalization ability and inherit weakness like lack of comprehensibility of ensemble approach.

Taghi M. Khoshgoftaar *et al.* [20] this paper includes comparative study of bagging and boosting techniques for noisy binary class data and imbalanced data. When data are clean but imbalance, boosting and bagging was less significant. Bagging can be used without replacement to deal with imbalanced and noisy data.

Victoria Lopez *et al*. [22] have focused on two issues, first were present methods to cope with imbalance problem, namely cost sensitive learning, preprocessing of instances, ensemble approaches and second was to show effect of intrinsic characteristics on imbalanced datasets. In this article they have pointed out that; ratio of imbalance itself doesn't affect performance of classifiers. In this paper they have presented 6 cases i.e. correct management of borderline samples, lack of density, overlapped class, and presence of small disjuncts, dataset shift and noisy data. For each issue they have described features that make classification algorithms to be biased and presented some proposed solutions.

L. Cleofas1 *et al.* [23] presented a framework which was the combination of genetic algorithm and under sampling method. In this article they have used the concept of under

sampling along with genetic algorithm to reduce processing time of genetic algorithm. Through this approach some chromosome was reduced and genes were ignored. For calculation of fitness function leaving one–out method was used and for selection of best solution in each step elitist method was used and after that genetic operator mutation and crossover was applied.

Lei Yu *et al.* [27] presented a methodology that performs redundancy analysis and relevance analysis. For redundancy and relevance analysis they have developed correlation based methodology which uses C and F correlations for redundancy analysis and C correlations for relevance analysis and conduct experimental study of its performance, effectiveness, and efficiency for supervised learning In this article they have used concept of variable selection, supervised learning to cope with high dimensionality and imbalance dataset. Below diagram shows relation between redundancy and relevance analysis.

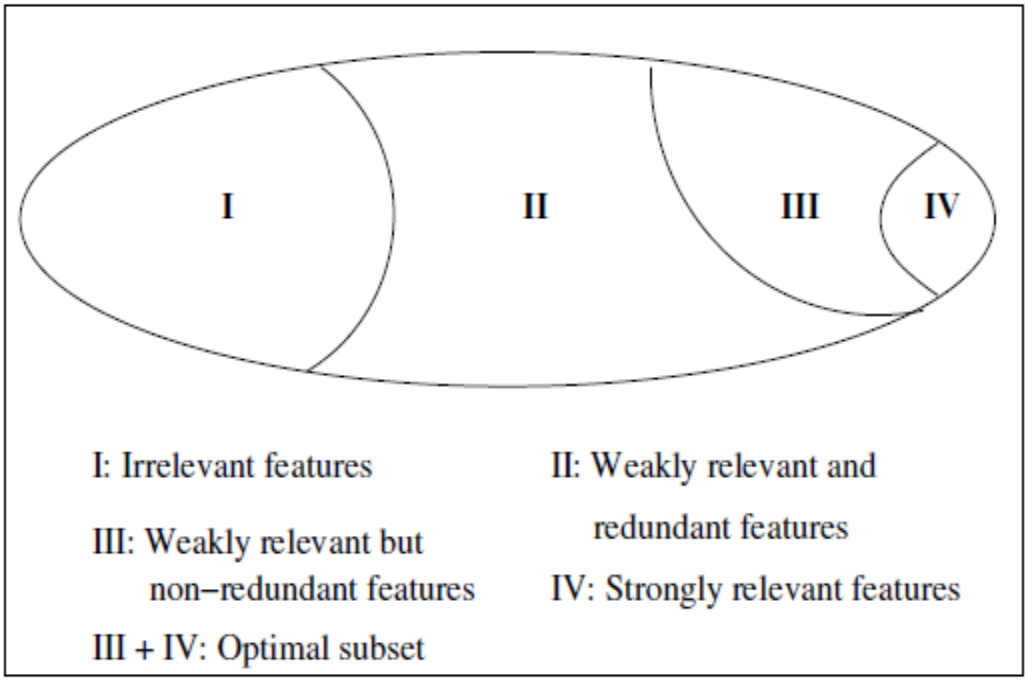


Figure 2.1 Relations between Redundancy and Relevance Analysis

Jesse Davis et al. [28] presented relation between receiver operator characteristic and precision- recall curve. ROC curve used for binary decision problems and PR curve give

information about performance of classification algorithms in machine learning. Researcher was show that curve dominates in precision–recall space it will also dominate in receiver operator characteristic curve. In this paper researcher was also show a method for computing precision-recall curve and receiver operator characteristic curve.

S. Natarajan *et al.* [29] presented an approach for classification and detection of tuberculosis. It is a disease due to mycobacterium which attacks low immune bodies and spreads through air. Presented methodology was the combination of classification and clustering that divide tuberculosis into two parts retroviral and pulmonary tuberculosis. In this paper they have used k-means clustering which divide tuberculosis data into 2 clusters and defined classes for each cluster.

Nitesh V. Chawla *et al*. [30] presented synthetic minority over-sampling technique to cope with class imbalance problem in machine learning. In this approach positive sample was oversampled by creating synthetic sample of existing positive samples. They have generated synthetic samples in feature space. Presented technique improves accuracy and performance of classification algorithms for minority class. Presented technique was the combination of under-sampling and SMOTE. Performance of presented technique was evaluated by area under the curve and ROC curve.

2.4 Research Gaps

This section tells about the gaps encountered during the research by reviewing the already existing literature in the area of class imbalance problem in machine learning.

i. High dimensionality issue in statistics and machine learning. These problems occur when number of training samples is less than the number of features [2].
ii. Class imbalance is a crucial problem in machine learning. Classification algorithms afflicted by the class imbalance problem for a dataset would see strong overall accuracy but very low performance on positive class [2] [17].
iii. Security and privacy issue is major challenge in large amount of data. Where data will be analyzed and mined for pattern [17] [49].

iv. Incremental learning for non-stationary data is the biggest challenge in machine learning [50].
v. Scalability and complexity is a major challenging issue in the field of machine learning. Traditional tools are not able to handle large datasets [12] [49].
vi. Timeliness is another problem for large datasets in machine learning. As the size of datasets is increases, analyzing time is also increases [12] [49].
vii. Addressing data quality is important issue in large datasets. Quality of data for decision making purpose will be risky if data is not correct [50].

2.5 Problem Formulation

The class imbalance is a crucial problem in machine learning, which has become an emerging research area in recent years. Classification algorithms afflicted by the class imbalance problem for a dataset would see strong overall accuracy but very low performance on the positive class. For example if we are working on our dataset and we applied classification algorithm and get 99% accuracy immediately. We think that it is fantastic but if we plunge a slight deeper and discover that 99% of the data belongs to only one class [32]. This is an example of an imbalance problem and it can cause the frustrating results. Imbalanced dataset exists in many real-world areas like oil spills detection from satellite images, fraud detection, anomaly detection, medical diagnosis, identifying fraudulent credit card transactions etc, which is composed with large percentage of negative samples and fewer percentage of positive samples because of which such classification algorithms works very well for negative samples but downgrade in case of positive samples but their accuracy imply very good. Hence, researchers have given more focus on problem of class imbalance.

2.6 Objectives

i. To study and analyze existing algorithms, methods, techniques and models for class imbalance.
ii. To propose a framework for class imbalance.
iii. To design, develop and test the proposed framework for class imbalance.

Chapter 3: Proposed Framework Subset Feature Selection (SFS)

To make experiments more interesting the first phase is to select new imbalanced datasets that have different attributes, different instances and different imbalance ratio. Second phase is Data preprocessing which include data format adaptation and data sampling. In data format adaptation first formats of datasets must be converted into .CSV files which are required by R interface. Data sampling depends on two parameters – percentage and bias. Data classification includes classification with various classifiers such as Naïve Bayes, Random Forest, AdaBoost and SVM. In classifier comparison phase we focus on results achieved by 4 classification algorithms such as Naïve bayes, Random Forest, AdaBoost and SVM, on each datasets under performance evaluation metrics: Accuracy, Sensitivity, AUC, Precision, F-measure and we compare obtained results to determine best classification algorithm for each dataset. Below Figure 3.1 shows phases of SFS framework.

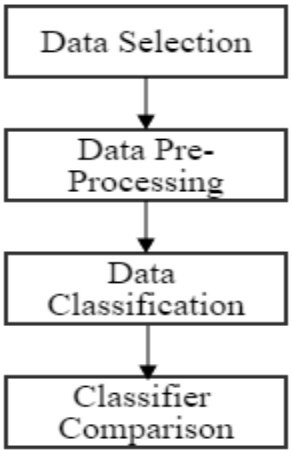


Figure 3.1 Phases of SFS

The research begins by selection of the imbalanced datasets. An imbalanced dataset is in which classes are not represented uniformly. These kinds of datasets are composed of typically two categories: Majority class and Minority class.

Below Figure 3.2 shows overall flow of SFS framework.

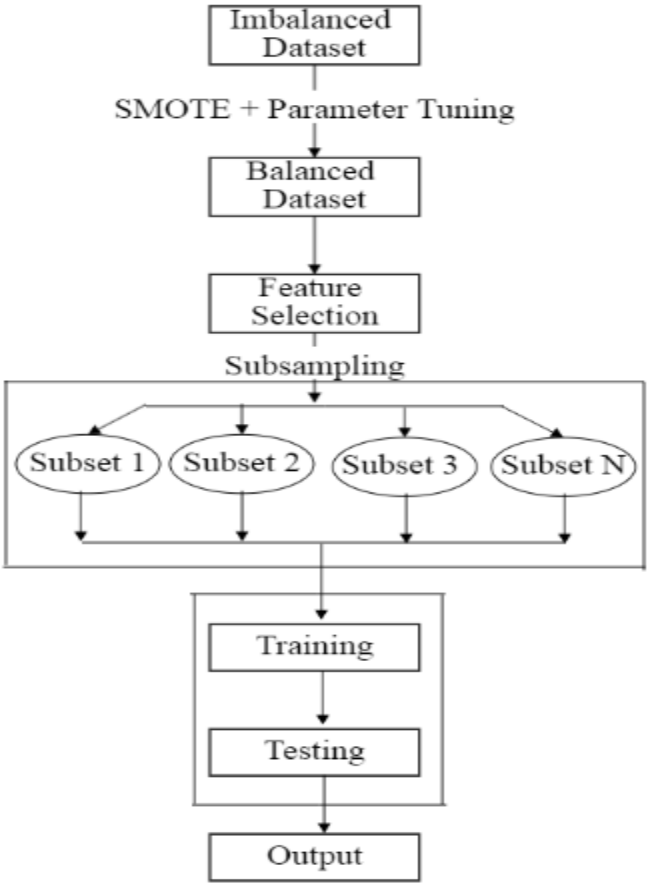


Figure 3.2 Block Diagram of SFS

Concept of parameter tuning along with SMOTE is used for the imbalanced dataset in order to bring forth a balanced dataset. We have applied the four binary classification algorithms for testing and training, namely:

a) Naïve Bayes
b) Random Forest
c) AdaBoost
d) SVM

These classification algorithms offer better performance outcomes for the measurement parameters such as Accuracy, Sensitivity, Precision, F-measure, AUC and they are calculated using eq.1, eq.2, eq. 3, eq. 4, eq. 5 respectively.

Once the dataset is balanced, important relevant features are selected from the large balanced dataset. This allows better learning performance, better model interpretability and lower computational cost. Feature selection returns important features of the original ones, according to certain relevance evaluation criterion.

In our work feature selection technique is applied to create the subsets of top features from selected features. After applying feature selection on original balanced data set, we got important features on which we further applied feature selection technique to produce different subsets of top features from selected features according to their rank.

The basic method is to train different classifiers like Naïve Bayes, Random Forest, AdaBoost, and SVM on multiple subsets of the features and then combine their production. After this, the resulted output is utilized for training and testing by different classification algorithms separately. The yield of all the four algorithms is compared and analyzed and our result outperforms other state -of –the- art techniques.

Chapter 4: Implementation Details

This chapter tells about the implementation performed during the research. Implementation details include installation of software, implementation of classification algorithms and implementation of proposed framework named Subset Feature Selection and also include snapshots of entire implementation.

4.1 SFS Implementation

4.1.1 R and R Studio Installation

a. First we downloaded R3.2.4 and installed it in our system.

b. After R installation we need to install R Studio.

4.1.2 SMOTE Implementation

For SMOTE implementation we did all the implementation with the help of R language. R is installed in the system and Rattle is used to make project where coding is done. For SMOTE implementation firstly we add fscaret library. After adding library we include DMwR package and perform read and write operation on Drug and Breast cancer imbalanced datasets. Through SMOTE firstly we convert imbalanced dataset into balanced dataset using parameter tuning. This conversion takes place using following function which includes various parameters like formula, dataset, k, Percentage of oversampling and percentages of under sampling.

```
dmSmote<-SMOTE(formula, data=dataset,k=5,
                perc.over =100,perc.under=90)
dmSmote
```

4.1.3 Feature Selection Implementation

For selection of features and subset of features from balanced datasets we did all the implementation with the help of R language. R is installed in the system and Rattle is used to make project where coding is done.

For selection of feature and subset of feature selection, firstly we add fscaret library. After adding library we include randomForest, ada, NB, and SVM packages, after that we performed read and write operation on balanced datasets. We have selected important features using following function which include various parameter like trainDataset, testDataset, MissData, preprocessData.

```
myFS<-fscaret(trainDataset, testDataset,missData="meanCol",
preprocessData=TRUE,Used.funcClassPred="ada", with.labels=TRUE,
no.cores=1)
```

4.1.4 Random Forest Implementation

We did all the implementation with the help of R language. For Random Forest implementation firstly we add various R packages and library like rpart, caret and fscaret. After adding rpart, caret and fscaret packages, we install randomForest package. After that we performed read and write operation on imbalanced and balanced datasets and measure performance of the random forest in terms of confusion metrics. Random forest can be implemented using following function which includes formula, trainDataset and method.

```
install.packages('randomForest')
library(randomForest)
model    <- randomForest(formula,trainDataset,method="randomForest")
```

After creating model we measure the performance of created model using confusion matrix. Confusion matrix can be created using following HMeasure and confusionMatrix functions.

```
EvaluationsParameters <- HMeasure(actual,Predicted)$metrics
EvaluationsParameters
ConfusionMatrix <-confusionMatrix(Predicted,actual)
ConfusionMatrix
```

4.1.5 SVM Implementation

We did all the implementation with the help of R language. For SVM implementation firstly we add various R packages and library like rpart, caret and fscaret. After adding rpart, caret and fscaret packages, we install SVM package and SVM library respectively.

After that we performed read and write operation on imbalanced and balanced datasets and measure performance of the SVM in terms of confusion metrics.SVM can be implemented using following function which includes formula, trainDataset and method.

```
install.packages("e1071")
library("e1071")
model    <- svm(formula,trainDataset,method="svm")
```

After creating model we measure the performance of created model using confusion matrix. Confusion matrix can be created using following HMeasure and confusionMatrix functions.

```
EvaluationsParameters <- HMeasure(actual,Predicted)$metrics
EvaluationsParameters
ConfusionMatrix <-confusionMatrix(Predicted,actual)
ConfusionMatrix
```

4.1.6 AdaBoost Implementation

We did all the implementation with the help of R language. For AdaBoost implementation firstly we add various R packages and library like rpart, caret and fscaret. After adding rpart, caret and fscaret packages, we install ada package and ada library respectively. After that we performed read and write operation on imbalanced and balanced datasets and measure performance of the adaboost in terms of confusion metrics. AdaBoost can be implemented using following function which includes formula, trainDataset.

```
install.packages("ada")
library("ada")
model <- ada(formula, trainDataset)
```

After creating model we measure the performance of created model using confusion matrix. Confusion matrix can be created using following HMeasure and confusionMatrix functions.

```
EvaluationsParameters <- HMeasure(actual,Predicted)$metrics
EvaluationsParameters
ConfusionMatrix <-confusionMatrix(Predicted,actual)
ConfusionMatrix
```

4.1.7 Naïve Bayes Implementation

We did all the implementation with the help of R language. For Naïve Bayes implementation firstly we add various R packages and library like rpart and fscaret.

and hmeasure. After adding rpart, caret and fscaret packages, we install NB package and NB library respectively. After that we performed read and write operation on imbalanced and balanced datasets and measure performance of the Naïve Bayes in terms of confusion metrics. Naïve Bayes can be implemented using following function which includes formula, trainDataset and method.

```
install.packages("NB")
library("NB")
model    <- naiveBayes(formula,trainDataset,method="naiveBayes")
```

After creating model we measure the performance of created model using confusion matrix. Confusion matrix can be created using following HMeasure and confusionMatrix functions.

```
EvaluationsParameters <- HMeasure(actual,Predicted)$metrics
EvaluationsParameters
ConfusionMatrix <-confusionMatrix(Predicted,actual)
ConfusionMatrix
```

Chapter 5: Experimental Results

The code for the project has been done in R language, and the simulations have been done on Windows 64 bit machine. To evaluate our subset feature selection method, we choose two dataset and apply 4 important machine learning algorithms before and after implementation of our proposed subset feature selection framework. A summary of datasets presented in Table 5.1.

Table 5.1 Summary of Datasets Used in Experiments

Dataset Name	Samples	Features
Drug	2000	100
Breast Cancer	700	59

An experimental result of proposed framework is presented below in tables and figures. Each table contains outcome of 4 classifiers along with the value of Accuracy, Sensitivity, AUC, Precision, and F-measure. The suggested framework is applied to Drug and Breast Cancer data sets. As already explained that we have applied the classification algorithms on imbalanced dataset. These algorithms are as follows: Naïve Bayes, Random Forest, AdaBoost, and SVM. We have compared these algorithms on the basis of their Accuracy, AUC, F-measure, Sensitivity and Precision. The experimental results for imbalanced Drug dataset are presented below in the Table 5.2 and Figure 5.1.

Table 5.2 Imbalanced Drug Dataset

Model Name	Accuracy	Sensitivity	AUC	Precision	F
Naïve Bayes	.498	.497	.526	.015	.295
RandomForest	.952	.986	.503	NAN	NAN
AdaBoost	.970	.992	.504	NAN	NAN
SVM	.976	1	.5	.02	.045

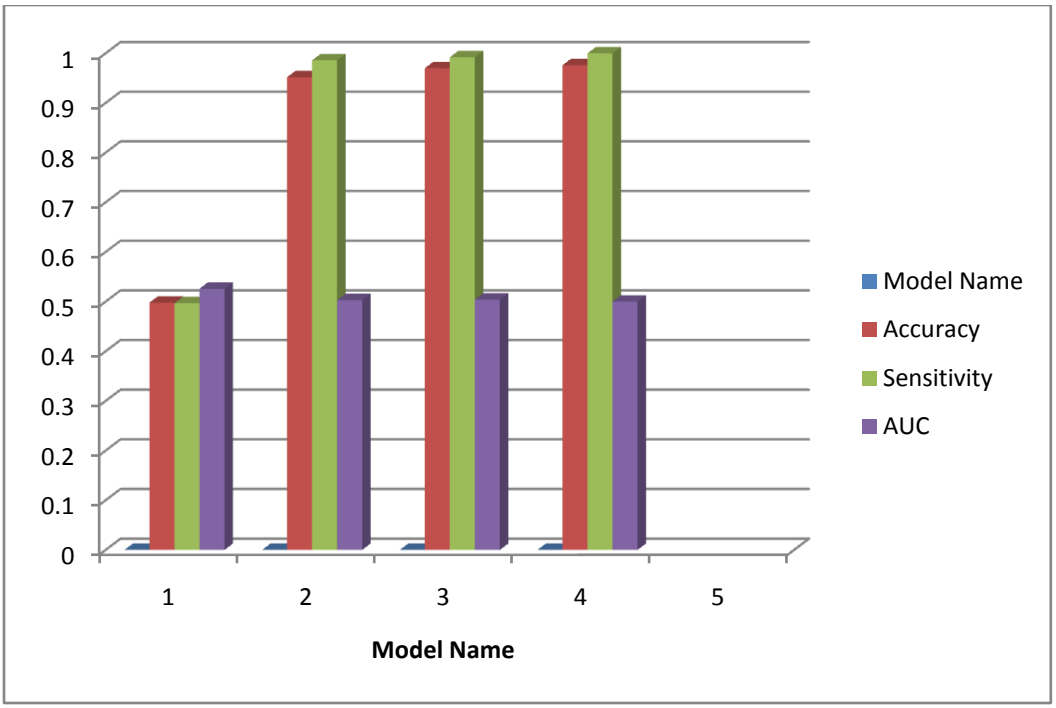


Figure 5.1 Imbalanced Drug Dataset

Now to balance the Drug dataset we have applied the SMOTE algorithm along with parameter tuning. After applying the SMOTE algorithm, Drug dataset is balanced. Now again we have applied the classification algorithms on balanced Drug dataset. After applying classification algorithms on balanced Drug dataset results are improved as compared to results of imbalance Drug dataset. The experimental results are presented below in Table 5.3 and Figure 5.2.

Table 5.3 Balanced Drug Dataset

Model Name	Accuracy	Sensitivity	AUC	Precision	F
Naïve Bayes	.614	.394	.609	.509	.674
RandomForest	.965	.938	.965	.472	.641
AdaBoost	.959	.915	.957	.517	.682
SVM	.862	.750	.861	.501	.667

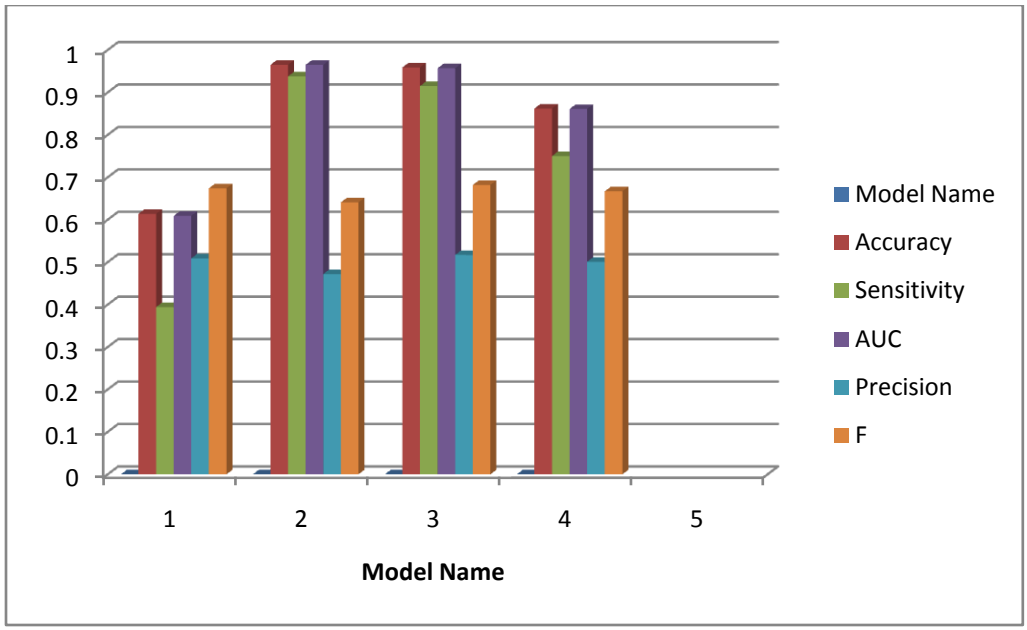


Figure 5.2 Balanced Drug Dataset

After balancing the Drug dataset we have applied the feature selection technique on balanced Drug dataset. Feature selection technique selects the important features from balanced Drug dataset and after getting the important features we have made the subset of top feature from the important features on the basis of their rank. After that we have applied classification algorithms on each subset of features. When we completed this, we have seen the tremendous change in the results as compared to the previous results of balanced and imbalanced Drug dataset. The experimental results are presented below in the form of Table 5.4 and Figure 5.3.

Table 5.4 Subset Feature Selection on Balanced Drug Dataset

Model Name	Accuracy	Sensitivity	AUC	Precision	F
NaïveBayes	.637	.463	.634	.509	.675
RandomForest	.973	.962	.972	.509	.675
AdaBoost	.959	.943	.959	.487	.655
SVM	.834	.817	.833	.533	.695

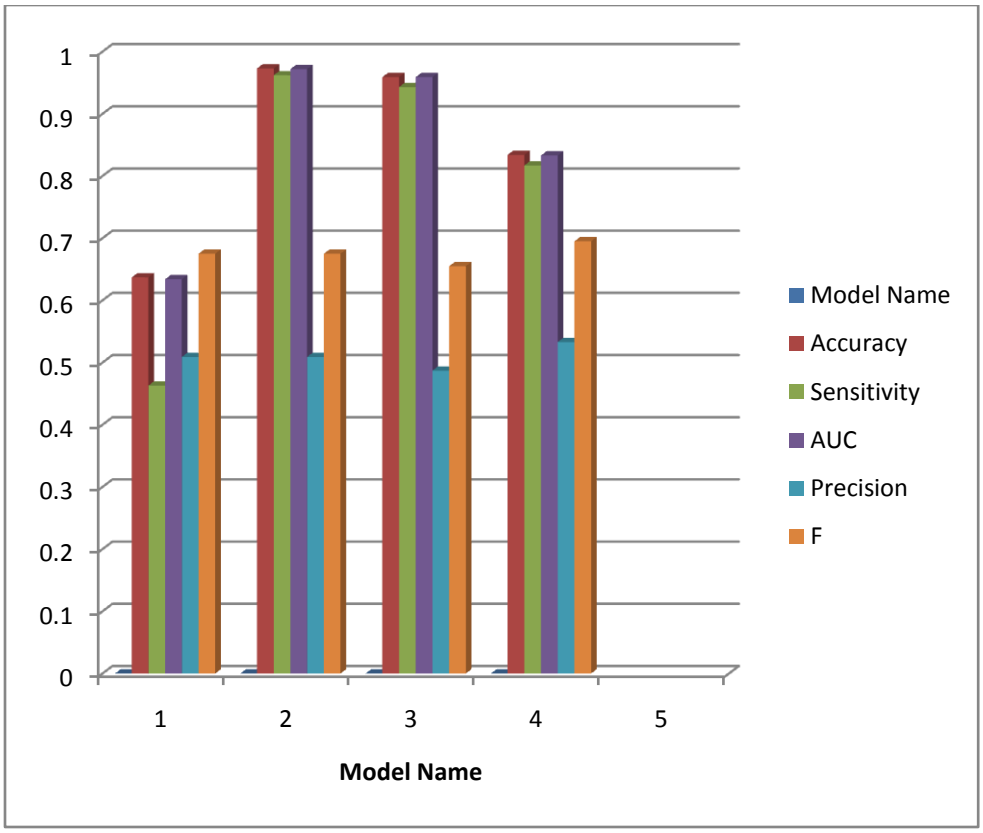


Figure 5.3 Subset Feature Selection on Balanced Drug Dataset

We have applied the classification algorithms on balanced Breast Cancer dataset. After applying classification algorithms on balanced Breast Cancer dataset, the experimental results are presented below in Table 5.5 and Figure 5.4.

Table 5.5 Balanced Breast cancer Dataset

Model Name	Accuracy	Sensitivity	AUC	Precision	F
Naïve Bayes	.938	.939	.938	.656	.792
RandomForest	.962	.985	.961	.633	.775
AdaBoost	.943	.865	.925	.647	.786
SVM	.962	.956	.960	.676	.806

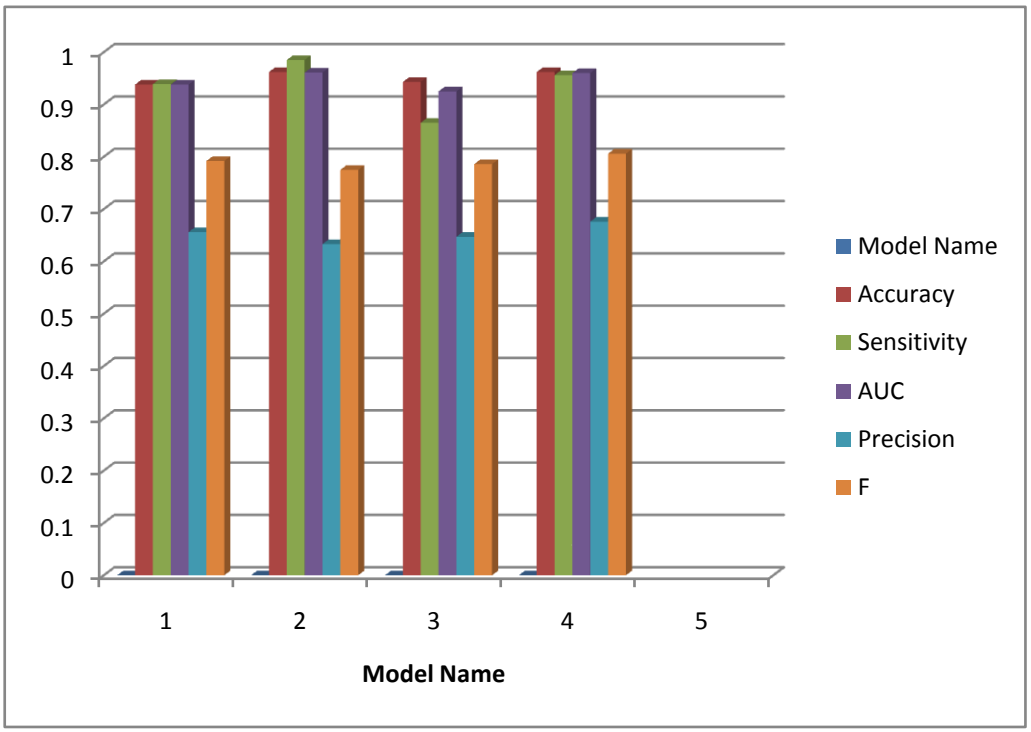


Figure 5.4 Balanced Breast Cancer

We have applied the feature selection technique on balanced Breast Cancer dataset. Feature selection technique selects the important features from balanced Breast Cancer dataset and after getting the important features we have made the subset of top feature from the important features on the basis of their rank. After that we have applied classification algorithms on each subset of features. When we completed this, we have seen the tremendous change in the results as compared to the previous results of balanced Breast Cancer dataset. The experimental results are presented below in the form of Table 5.6 and Figure 5.5.

Table 5.6 Subset Feature Selection on Balanced Breast Cancer Dataset

Model Name	Accuracy	Sensitivity	AUC	Precision	F
Naïve Bayes	.962	.986	.967	.657	.793
Random Forest	.971	.986	.975	.666	.799
AdaBoost	.967	.971	.968	.671	.803
SVM	.971	.944	.965	.663	.797

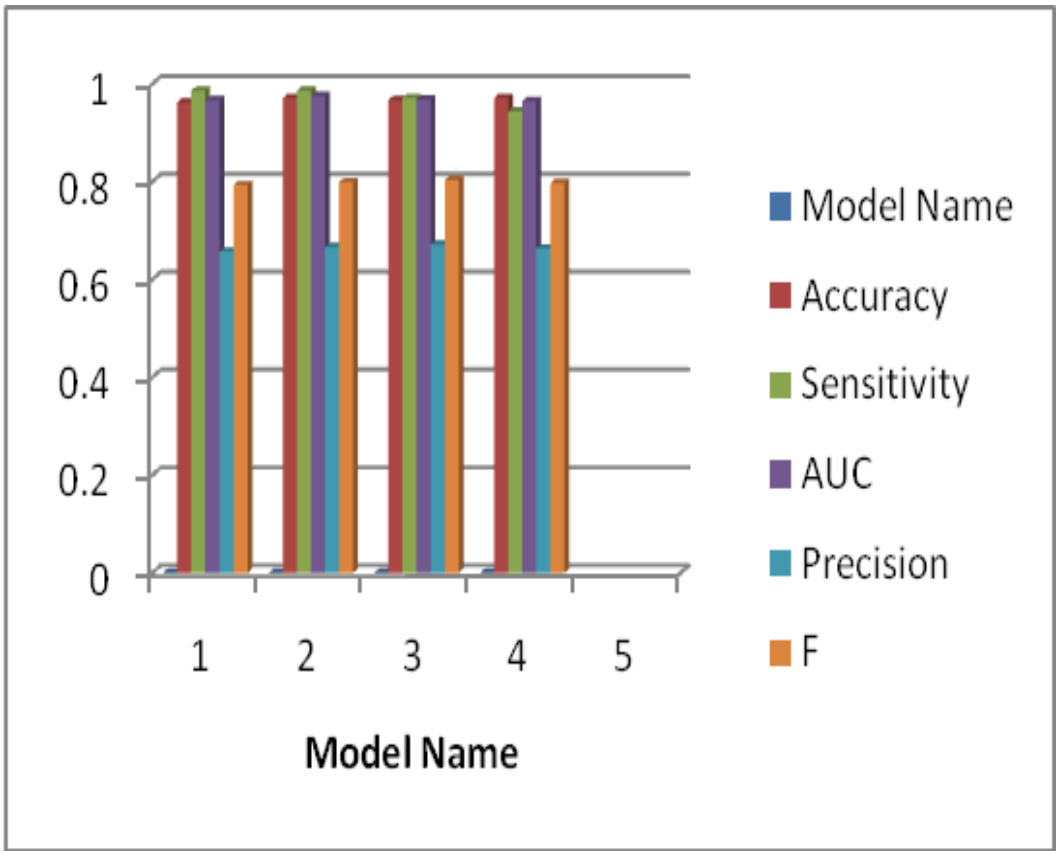


Figure 5.5 Subset Feature Selection on Balanced Breast Cancer Dataset

Chapter 6: Conclusion and Future Work

6.1 Conclusion

Class imbalance is a major problem in real world datasets. Numbers of techniques like cost sensitive learning techniques, recognition based techniques, and sampling based techniques, exist to handle data imbalance problem. However, these techniques suffer from data loss and over fitting because they change the original distribution of data. Surmount the data imbalance problem in this thesis; a new framework is proposed, called Subset Feature Selection (SFS). It is employed as a pre-processing and feature selection step to train Naïve Bayes, Random Forest, AdaBoost, and SVM on imbalanced dataset and balanced dataset. The proposed SFS framework firstly converts imbalanced datasets into balanced datasets with the help of SMOTE algorithm. In SMOTE algorithm datasets is balanced by 3 parameters such as over-sampling, under-sampling and K, where K is instances of nearest neighbour's. After that we have applied classification algorithms on balanced datasets and with the help of various R packages and libraries we select the important features. Further we applied classification algorithms on selected features. When compared with experimental results derived using SMOTE along with parameter tuning and feature selection it is concluded that the performance improvement of subset feature selection with Naïve Bayes, Random Forest is better than Adaboost and SVM. It can be concluded that classification of data can be improved significantly to key out the rare events from the datasets by applying subset feature selection. Finally it is concluded that proposed framework give better performance results than other techniques.

6.2 Future Work

Future work will involve conducting additional empirical studies with imbalanced datasets. More comparative study using other sampling approaches and rankers will also be considered in the future.

References

[1] D. Annarita, M. Rosalia, “Parallel selective sampling method for imbalanced and large data classification”, ELSEVIER, Pattern Recognition Letters 62, pp. 61–67, 2015.

[2] S. Zhongbin, S. Qinbao, Z. Xiaoyan, S. Heli, Z. Yuming, “A Novel ensemble method for classifying imbalanced data”, ELSEVIER, Pattern Recognition 48, pp. 1623–1637, 2015.

[3] B. Cigdem, F. Robert, “Classifying imbalanced data sets using similarity based hierarchical decomposition”, ELSEVIER Pattern Recognition 48, pp. 1653–1672, 2015.

[4] Z. Shu, S. Samira, M. Malck, “An empirical analysis of imbalanced data classification”, Computer and Information Science, pp. 151-162, 2015.

[5] P. L. Son, B. Abdesselam, N. H. Giang, “Learning pattern classification tasks with imbalanced data sets”, ELSEVIER, Pattern recognition, pp. 193-208, 2009.

[6] L. yang, Y. Xiaohui, H.X. Jimmy, A. Aijun, “Combining integrated sampling with SVM ensembles for learning from imbalanced datasets”, ELSEVIER, Information Processing and Management 47, pp. 617–631, 2011.

[7] N. Yok-Yen, C. Siu-Yeung, “An unsupervised self-organizing learning with support vector ranking for imbalanced datasets”, ELSEVIER, Expert Systems with Applications 37, pp. 8303–8312, 2010.

[8] B. Rok, L. Lara, “Joint use of over- and under-sampling techniques and cross validation for the development and assessment of prediction models”, BMC Bioinformatics, pp. 1-10, 2015.

[9] B. Marcelo, F. F. E. Nelson, S. L. Beatriz, L. D. Pires, “A KNN under-sampling approach for data balancing”, Journal of Intelligent Learning Systems and Applications, PP. 104-116, 2015.

[10] C. A. David, N. V. Chawla, “Start globally, optimize locally, predict globally on improving performance on imbalanced data”, Eighth IEEE International Conference on Data Mining, pp. 143-152, 2008.

[11] R. M. Mostafizur, D. N. Davis, “Addressing the class imbalance problem in medical datasets”, International Journal of Machine Learning and Computing, pp. 224-228, 2013.

[12] W. Juanjuan, X. Mantao, W. Hui, Z. Jiwu, “Classification of imbalanced data by using the SMOTE algorithm and locally linear embedding”, IEEE International Conference on Data Mining, 2006.

[13] L. Xu-Ying, W. Jianxin, Z. Zhi-Hua, “Exploratory under-sampling for class-imbalance learning”, IEEE Transactions on Systems, Man and Cybernetics, pp. 1-14, 2008.

[14] N. Mehdi, M. B. Amir, V. Touraj, “A hybrid feature selection method to improve performance of a group of classification algorithms”, International Journal of Computer Applications, pp. 28-35, 2013.

[15] L. Lin, R. Guy, S. Meiling, C. Shuching, “Effective feature space reduction with imbalanced data for semantic concept detection”, International Journal of Computer Applications, pp. 48-55, 2011.

[16] C. Xuewen, W. Michael, “FAST: A ROC-based feature selection metric for small samples and imbalanced data classification problems” ACM, pp.124-132, 2008.

[17] H.Haibo, A.G.Edwardo, “Learning from imbalanced data”, IEEE Transactions on Knowledge and Data Engineering, pp. 1263-1284, 2009.

[18] V. C. Nitesh, “C4.5, “An imbalanced data sets: investigating the effect of sampling method, probabilistic estimate, and decision tree structure”, Workshop on Learning from Imbalanced Datasets Washington DC, 2003.

[19] Z. Xueying, S. Qinbao, W. Guangtao, Z. Kaiyuan, H. Liang, J. Xiaolin, “A dissimilarity-based imbalance data classification algorithm”, SPRINGER Application Intelligence, pp. 544–565, 2015.

[20] M. K. Taghi, V. H. Hulse, N. Amri, “Comparing boosting and bagging techniques with noisy and imbalanced data”, IEEE Transactions Systems, Man, Cybernetics A Systems Humans, pp. 552-568, 2011.

[21] Y. L. Tian, “EasyEnsemble and feature Selection for imbalance data sets”, In Proceedings IEEE International Joint Conference on Bioinformatics, Systems Biology and Intelligent Computing, pp. 517-520, 2009.

[22] L. Victoria, F. Alberto, G. Salvador, P. Vasile, H. Francisco, “An insight into classification with imbalanced data: Empirical results and current trends on using data intrinsic characteristics”, ELSEVIER Journal of Information Science 250, pp. 113-141, 2013.

[23] L. Cleofas, R. M. Valdovinos , V. García, R. Alejo, “Use of ensemble based on GA for imbalance problem”, In Proceedings IEEE Congress on Evolutionary Computation, pp. 2254-2261, 2004.

[24] F. Georage, “An extensive empirical study of feature selection metrics for text classification”, Journal of Machine Learning Research 3, pp. 1289-1305, 2003.

[25] G. Isabelle, E. Andre, “An introduction to variable and feature selection”, Journal of Machine Learning Research 3, pp. 1157-1182, 2003.

[26] G. Kehan, K. Taghi, W. Randall, “Combining feature selection and ensemble learning for software quality estimation”, Twenty-Seventh International Florida Artificial Intelligence Research Society Conference, 2014.

[27] Y. Lei, L. Huan, “Efficient feature selection via analysis of relevance and redundancy”, Journal of Machine Learning Research 5, pp. 1205–1224, 2004.

[28] D. Jesse, G. Mark, “The relationship between precision-recall and ROC curves”, Neural Information Processing Systems 15 (NIPS), 2003.

[29] T. Asha, S. Natarajan, K.N.B. Murthy, “A data mining approach to the diagnosis of tuberculosis by cascading clustering and classification”, Journal of Computing, 2011.

[30] N.V. Chawla, K.W. Bowyer, L.O. Hall, W.P. Kegelmeyer, “SMOTE: synthetic minority over-sampling technique”, Journal of Artificial intelligence Research 16, pp. 341–378, 2002.

[31] L. Huan, M. Hiroshi, S. Rudy, Z. Zheng, “Feature selection an ever evolving frontier in data mining”, Fourth Workshop on Feature Selection in Data Mining, pp. 4-13, 2010.

[32] V. Nikulin, G. J. McLachlan, S. K. Ng, “Ensemble approach for the classification of imbalanced data”, Advance Artificial Intelligence, Springer, pp. 291–300, 2009.

[33] J. Huang, C. X. Ling, “Using AUC and accuracy in evaluating learning algorithms”, IEEE Transactions Knowledge Data Engineering 17, pp. 299–310, 2005.

[34] S. Wang, X. Yao, “Diversity analysis on imbalanced datasets by using ensemble models”, IEEE Symposium on Computational Intelligence and Data Mining, pp. 324–331, 2009.

[35] H. Guo, H. Viktor, “Learning from imbalanced datasets with boosting and data generation”, ACM SIGKDD Explor.Newsl.6, pp. 30–39, 2004.

[36] R. Ranawana, V. Palade, “Optimized precision: a new measure for classifier performance evaluation”, IEEE Congress on Computational Intelligence, Canada, pp. 2254–2261, 2006.

[37] G. M. Weiss, “The impact of small disjuncts on classifier learning”, Jouranl of Information Systems, SPRINGER, pp. 193–226, 2010.

[38] V. Garcia, J. S. Sanchez, R. A. Mollineda, “On the effectiveness of preprocessing methods when dealing with different levels of class imbalance”, ELSEVIER Knowledge-Based Systems 25, pp. 13–21, 2012.

[39] K. Secuk, Z. Gokmen, G. Dincer, “Drug/nondrug classification using Support Vector Machines with various feature selection strategies”, ELSEVIER Computer Methods and Programs in Bio-Medicine, pp.51–60, 2014.

[40] I. H. Witten, E. Frank, “Data Mining: Practical machine learning tools and techniques, 2nd ed.”, Morgan Kaufmann Publishers, San Francisco, 2005.

[41] R. Kohavi, G. H. John, “Wrappers for feature subset selection”, Artificial Intelligence 97, pp. 273–324, 1997.

[42] S. Aixin, L. Ee-Peng, L. Ying, “On strategies for imbalanced text classification using SVM: A comparative study”, ELSEVIER Decision Support Systems 48, pp. 191–201, 2009.

[43] Z. Zheng, X. Wu, R. Srihari, "Feature selection for text categorization on imbalanced data", ACM SIGKDD Explorations Newsletter 6 (1), pp. 80–89, 2004.

[44] A. Sun, E.-P. Lim, B. Benatallah, M. Hassan, "FISA: Feature-based instance selection for imbalanced text classification", Proceeding of PAKDD, pp. 250–254, 2006.

[45] D. Fragoudis, D. Meretakis, S. Likothanassis, "Integrating feature and instance selection for text classification", Proceeding of ACM SIGKDD, pp. 501–506, 2002.

[46] K. Priyanka, N. Abhigyan, C. Radha, "Identification of human drug targets using machine-learning algorithms", ELSEVIER Computers in Biology and Medicine 56, pp. 175–181, 2015.

[47] M. Hossin, M. N. Sulaiman, "A review on evaluation metrics for data classification evaluations", International Journal of Data Mining & Knowledge Management Process (IJDKP), 2015.

[48] M. Shaza, E. Abd, A. Ajith, "A review of class imbalance problem", Journal of Network and Innovative Computing, pp. 332-340, 2013.

[49] K. U. Jaseena, M. D. Julie, "Issues, Challenges, and Solutions in Data Mining", NeTCoM, pp. 131–140, 2014.

[50] N. M. Maryam, V. Flavio, M. K. Taghi, "Deep learning applications and challenges in machine learning" SPRINGER, Journal of Big Data, pp. 1-21, 2015.

[51] A. A. Haya, R. M. Mohammad, "Handling class imbalance in direct marketing dataset using a hybrid data and algorithmic level solutions", SAI Computing Conference, pp. 13-15, 2016.

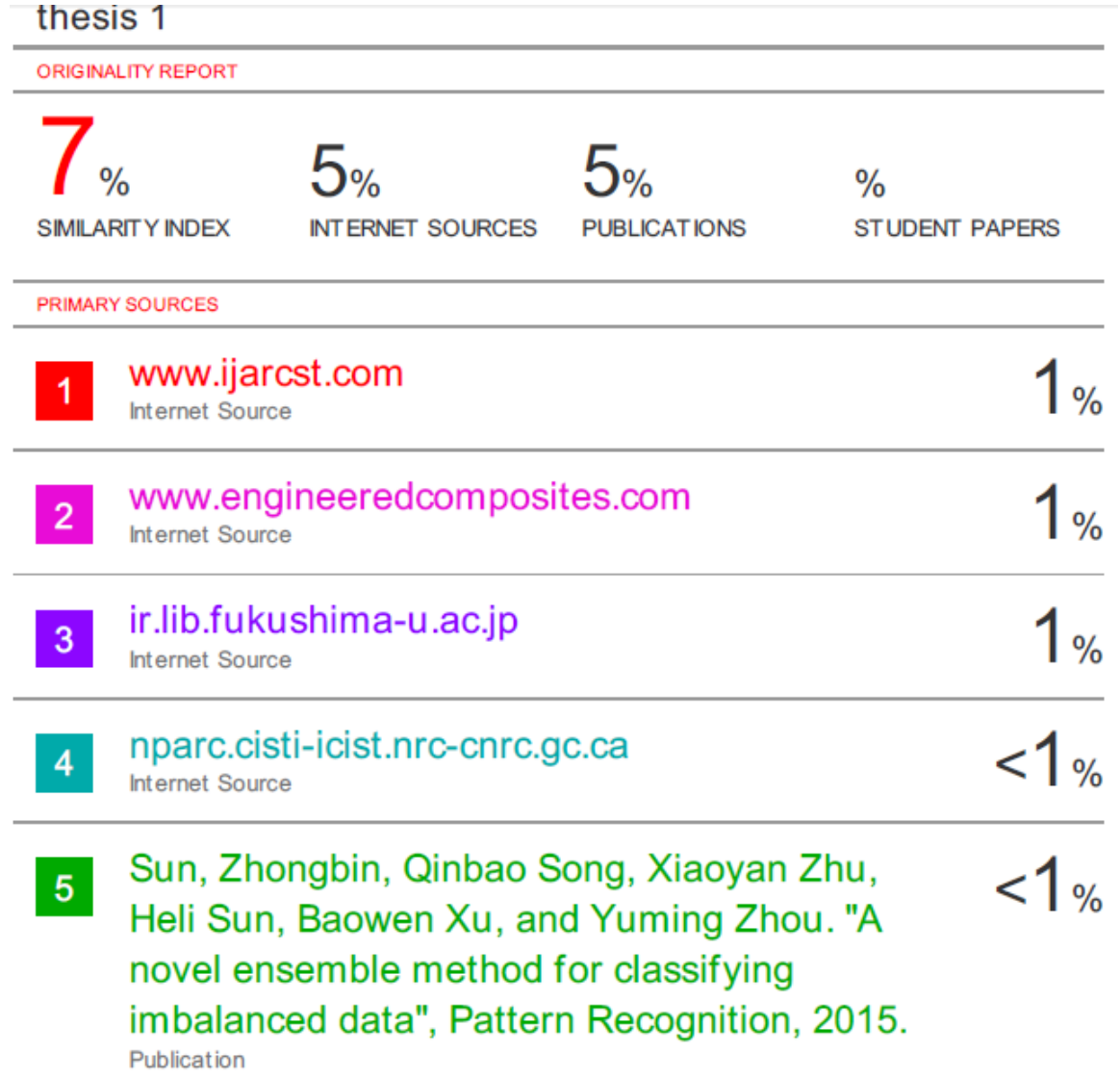
List of Publication

Pawan Lachheta and Seema Bawa, "Combining Synthetic Minority Oversampling Technique and Subset Feature Selection Technique for Class Imbalance Problem", in International Conference on Advances in Information Communication Technology & Computing" (ACM) (AICTC - 2016). [Accepted]

Video Link

https://youtu.be/RAzaH_4EGTs

Plagiarism Certificate



thesis 1

ORIGINALITY REPORT

7%	5%	5%	%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	www.ijarcst.com Internet Source	1%
2	www.engineeredcomposites.com Internet Source	1%
3	ir.lib.fukushima-u.ac.jp Internet Source	1%
4	nparc.cisti-icist.nrc-cnrc.gc.ca Internet Source	<1%
5	Sun, Zhongbin, Qinbao Song, Xiaoyan Zhu, Heli Sun, Baowen Xu, and Yuming Zhou. "A novel ensemble method for classifying imbalanced data", Pattern Recognition, 2015. Publication	<1%