CAPITAL UNIVERSITY OF SCIENCE AND TECHNOLOGY, ISLAMABAD

Handling Class Imbalance Data using Class Label Prediction

by

Asifa Kanwal

A thesis submitted in partial fulfillment for the degree of Master of Science

in the

Faculty of Computing

Department of Computer Science

2020

My dissertation work is devoted to My Family, My Teachers and My Friends. I have a special feeling of gratitude for My beloved parents, brothers. Special thanks to my supervisor whose uncountable confidence enabled me to reach this milestone.

CERTIFICATE OF APPROVAL

Handling Class Imbalance Data using Class Label Prediction

by

Asifa Kanwal

(MCS181049)

THESIS EXAMINING COMMITTEE

S. No.	Examiner	Name	Organization
(a)	External Examiner	Dr. Waseem Shehzad	FAST, Islamabad
(b)	Internal Examiner	Dr. Shahid Malik	CUST, Islamabad
(c)	Supervisor	Dr. Nayyer Masood	CUST, Islamabad

Dr. Nayyer Masood
Thesis Supervisor
November, 2020

Dr. Nayyer Masood
Head
Dept. of Computer Science
November, 2020

Dr. Muhammad Abdul Qadir
Dean
Faculty of Computing
November, 2020

Author's Declaration

I, **Asifa Kanwal** hereby state that my MS thesis titled "**Handling Class Imbalance Data using Class Label Prediction**" is my own work and has not been submitted previously by me for taking any degree from Capital University of Science and Technology, Islamabad or anywhere else in the country/abroad.

At any time if my statement is found to be incorrect even after my graduation, the University has the right to withdraw my MS Degree.

(Asifa kanwal)
Registration No: MCS181049

Plagiarism Undertaking

I solemnly declare that research work presented in this thesis titled "**Handling Class Imbalance Data using Class Label Prediction**" is solely my research work with no significant contribution from any other person. Small contribution/help wherever taken has been dully acknowledged and that complete thesis has been written by me.

I understand the zero tolerance policy of the HEC and Capital University of Science and Technology towards plagiarism. Therefore, I as an author of the above titled thesis declare that no portion of my thesis has been plagiarized and any material used as reference is properly referred/cited.

I undertake that if I am found guilty of any formal plagiarism in the above titled thesis even after award of MS Degree, the University reserves the right to withdraw/revoke my MS degree and that HEC and the University have the right to publish my name on the HEC/University website on which names of students are placed who submitted plagiarized work.

(Asifa Kanwal)
Registration No: MCS181049

Acknowledgements

"Say, He is Allah, [Who is] One. Allah, the Eternal Refuge. He neither begets nor is born. Nor is there to Him any equivalent." Al-Quran [112:1-4]. I would like to say Alhamdulillah, for blessing me, with strength to finish this work. My heartfelt thanks to my parents for their love, prayers and everything what I need. A special thanks to my teachers, brothers for their support. Thanks to my friends, with whom this journey becomes easy to me. Last but not the least special thanks to my esteemed supervisor Dr. Nayyer Masood for the support, for his assistance, inspiration and guidance in the field of research. I have learnt not only research under his supervision, but also his attitude towards others. May Allah shower his countless blessing upon all of you, JazakAllah.

(Asifa Kanwal)

Registration No: MCS181049

Abstract

Most of the machine learning algorithms perform best with a dataset having almost equal number of instances for each class label. A dataset with an inappropriate ratio of class labels is considered to have class-imbalance problem. Learning through class-imbalanced data creates an unreal impression of the prediction model. There exist such techniques to attain equal class distribution by creating synthetic data that are more flexible than algorithmic level moderations. These methods, named over-samplers, change the dataset into a balanced data set that enables any classification algorithm to produce more realistic results. For this purpose, numerous techniques have been suggested; the SMOTE (Synthetic Minority Oversampling Technique) being the most popular. The k-Mean SMOTE is a recent variant of SMOTE but it produces needless/noisy data. We have proposed 2 different approaches named as Minority Class Clustering SMOTE (MCC-SMOTE) and Oversee SMOTE to address the issues of k-Mean SMOTE. MCC-SMOTE reduces the noise generation by, firstly, generating the clusters of minority class then using these clusters to generate synthetic data. In Oversee SMOTE, first we apply under-sampling on majority class to balance the data, then we create synthetic instances and get the probabilities for each class label for these new instances. The synthetic instances where probability for the majority class is higher are ignored whereas the ones which get higher probabilities for the minority class are added to the dataset. This process is repeated until the data becomes balanced. The results are compared in terms of G-mean, F1-measure, accuracy, AUC and TPR/Recall. The results of extensive experiments that were conducted on 23 datasets show that proposed approaches outperformed the baseline technique.

Contents

List of Figures

List of Tables

Abbreviations

IR	Imbalance Ratio
IRT	Imbalance Ratio Threshold
KNN	K- Nearest Neighbor
LR	Logistic Regression
MCC	Minority Class Clustering
NB	Naïve Bayes
SMOTE	Synthetic Minority Oversampling Technique
TPR	True Positive Rate

Chapter 1

Introduction

Involvement of excessive amount of data in multifarious disciplines causes generation of huge amounts of information. Various fields including medicines, educational institutions, warehouses etc. contain bulk of information in different formats. When it comes to handling such information, data science in deemed as an emerging field. In order to discover different patterns, Analytics utilizes this data to obtain significant information using conventional machine learning algorithms. However, these conventional machine learning algorithms can only be applied if the data is cleaned and balanced. Therefore, data must be cleaned appropriately, so that accurate results could be obtained. In data science, the process of converting data into useful form involves several steps. First of all, the raw data is processed and pre-processing is applied thereafter. The processed raw data gets converted into structured format but still the data is in uncleansed format when passed further for pre-processing. This could be due to the fact that it may have dirty data because of duplicates, missing values, absurd outliers or inconsistent values that could mislead the results. Different strategies are employed by the data scientists to clean this data. After cleaning, next step is to learn a model by using some machine learning algorithm. Once the model is trained, the outcomes can be observed in the form of evaluation metrics or graphs [1]. Correctness of data plays a pivotal role in obtaining the accurate results. The main and essential steps leading to correctness of data is data pre-processing. Data preprocessing transforms the raw data into a clean set of data. In other words, it is obtained in raw format whenever the data is collected from different sources, which is not feasible for analysis. There are several techniques for data preprocessing. Integration of data, resampling of data, transformation of data, cleaning of data and reduction of data. The real world data is mostly in the form of noise, incoherent, and incomplete. The main issue arises when the data is missing. The existing pre-processing methods focus on cleaning the data via filling in the missing values, smooth out noise while identifying outliers, and correct data inconsistencies. Data preprocessing also involves integration of data — the combination of data from numerous data sources. Careful incorporation while integrating data into the resulting data set can help to reduce and eliminate redundancies and inconsistencies. Moreover, Tre liability and efficiency can also be improved through data mining process that follows the said procedure. Data reduction methods can be implemented in order to obtain a reduced representation of the data set. The reduction of data should be done in such a manner that quality of data does not get compromised. That is, mining should be more effective on the reduced data set while generating the same (or nearly the same) analytical results. Transformation of data (e.g., standardization) can be implemented where data is weighted to drop within a smaller range such as 0.0 to 1.0 [2]. Instead of the above discussed issues, now a days, the major issue faced during data classification is class-imbalance problem. Frequently, Class imbalance is an important issue in different real-world information sets, where one class (i.e. Minority class) has a tiny amount of data points and the other class (i.e. Majority class) has a big amount of data points as shown in Figure 1.1. The reason behind imbalanced data is that some applications produce a dataset of skewed nature. Class disequilibrium impairs the prediction performance of supervised learning. Most other techniques seek to optimize the precision of classification, a metric which is skewed against the dominant class. A classification algorithm can obtain optimal classification precision even though it does not forecast accurately an instance of a single minority class. For example, in credit card fraud detection dataset, very few classes are fraud and most of the classes are not fraud or rare medical diagnoses [2][3]. Let us consider an example

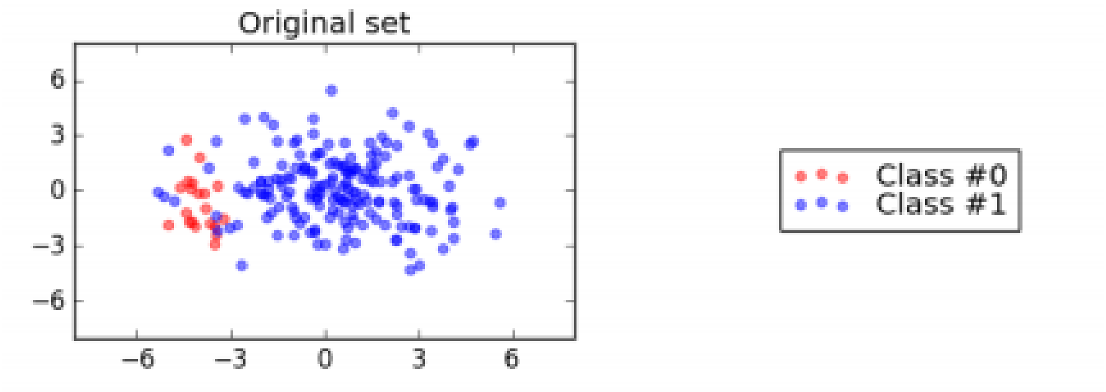

Figure 1.1: Class Imbalance Representation

of employ working in a leading tech company. The company has assigned him a task to train a model for fraud detection. The fraud transaction is relatively rare; so he trained the model and obtained around 95% accuracy. He feels good and presents his model in front of company's CEO and shareholders. When they give inputs to his trained model, the model predicts "Not a Fraud Transaction" every time. It is obviously a challenge, since many algorithms for machine learning are programmed to improve overall performance. Suppose the case of an imbalanced dataset with a 5:75 class imbalance. In this problem, their exist 5 minority (Fraud) samples and 75 majority (Not Fraud) samples. Now consider that the model is already trained and training set is also imbalance. After testing these 80 samples let the confusion matric presented in Table 1.1. If we calculate the accuracy from above confusion matric it gives 95% accuracy while almost all minority samples predicted wrongly. Which means classifier's trained model becomes bias towards majority class.

The underlying expectation in certain classifiers is the consistency of the costs of misclassified instances which is seldom a function of real-world issues. Unbalanced databases usually give misleading results of the positive class as dominant class is correlated with a higher expense than conversely. An illustration of this is the

Table 1.1: Confusion Metrics

	Class 1	**Class 0**
Class 1	1	4
Class 0	0	75

selling of repositories, where postage costs to a non-respondent are much smaller than the missed income of not postage to a respondent[4].

To tackle the above-mentioned deceptive data tactics, three groups are already defined. External (data level) approaches, Internal (algorithm level) approaches and their hybrid form [5]. But we focused on data level approach. Data level approach is in which first of all balance the dataset through resampling then a classifier is applied to do classification [5]. There exist two techniques of resampling, one is oversampling (to handle minority class by creating artificial instances) and other is undersampling (to handle majority class by eliminating instances). Under-sampling is accomplished by eliminating meaningless data points, either by selecting randomly or by using some heuristic rules. Like, some forms of undersampling are simplified nearest neighbor rule and one-sided selection [6]. Undersampling, though, is dangerous since potentially valuable data may remove. The other option which is possibly more effective is over-sampling [3][4]. Therefor we used oversampling technique to resample the datasets. In introduced methodologies instance generating step is done by using SMOTE (Synthetic Minority Oversampling Technique). SMOTE is a well-known and simple technique. Due to its simplicity and good results most of the researchers used it in their methodologies [5][7][8].

1.1 Data Balancing Approaches

There are three groups of data balancing approaches that are already defined such as external approaches, internal approaches and hybrid approaches which is detailed explained in coming sections.

(i) External approaches which is also named as data level approaches.

(ii) Internal approaches which is also named as algorithm level approaches.

(iii) Hybrid form of data level approaches and algorithm level approaches are mostly combination of internal and external approaches.

1.1.1 External Approaches

Firstly, the instances of positive class and negative class brings equal in external methods (data level) using resampling methods, after that the standard learning techniques are implemented so that efficiency of the classification algorithms is not skewed against the dominant class. Through filtering, i.e. deleting objects of dominant class or oversampling, i.e. by increasing the positive class objects, the re-sampling of the data sets is achieved. In this type of approaches classifiers not changed i.e. without changing the logic of classification algorithm, as re-sampling of imbalance data is performed prior to classification, calling such techniques as data pre-processing methods [5].

1.1.1.1 What is Re-sampling

As we discussed above in introduction section that some real word applications produces data of skewed nature in which one class dominant the other one. Standard classifiers are designed to implement on balanced dataset so if data is not balanced the results of classification will be biased. Reason of the biased results is that classifier ignore the minority class instances as it is noisy data and train the model on majority class samples. To tackle this issue of unbalancing re-sampling is done in which imbalance data becomes balance by using different strategies. Mainly there is two strategies one is oversampling and other is undersampling but a third strategy namely hybrid sampling is also used which is less common. Figure 1.2 describes the concept of re-sampling of data.

1.1.1.2 What is Oversampling

Oversampling is a data level approach which is carried out on training dataset by generating synthetic instances of positive class so that number of positive and negative class instances becomes equal as shown in Figure 1.3 original dataset and Figure 1.4 balanced dataset through oversampling. Class 0 is majority class represented with blue dots and class 1 is minority class represented with orange

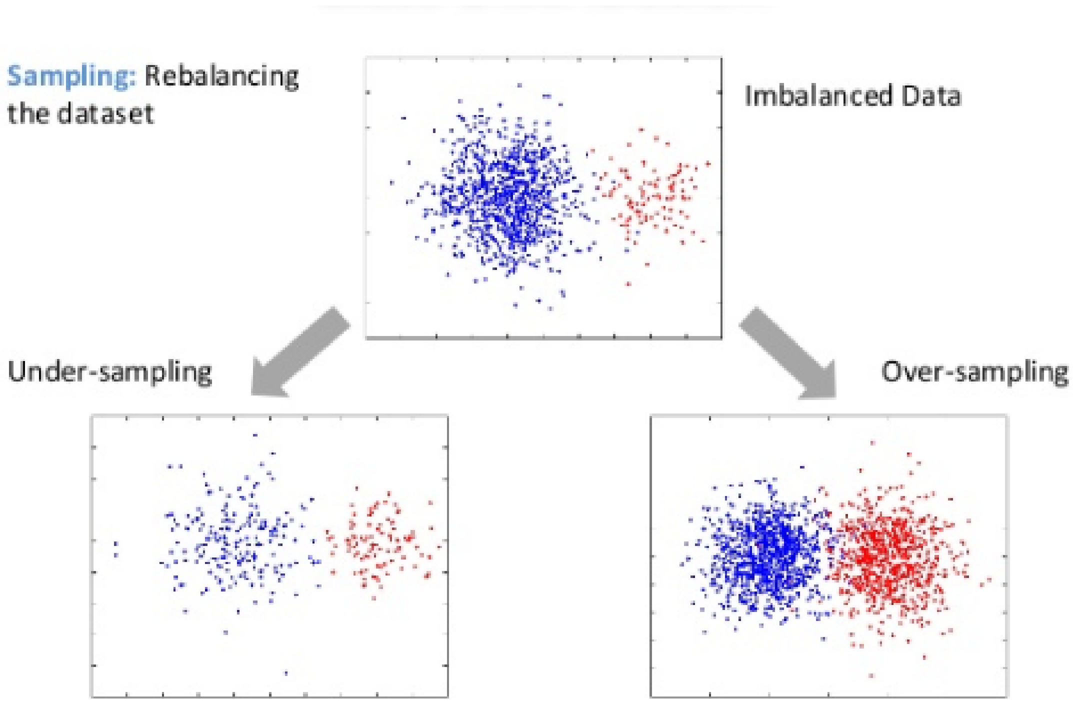

Figure 1.2: Resampling Concept

color dots. Mainly, there exist three common oversampling techniques named as Random Oversampling, SMOTE, adaptive synthetic sampling approach, or ADASYN. Till now many researchers have proposed various modifications in these existing techniques.

1.1.1.3 What is Undersampling

Undersampling is also a data level technique which is used to balance the training dataset. This type of approaches are carried out by eliminating the instances from the majority class to bring its number of instances equals to the minority class instances. There exist many undersampling techniques like Random Undersampling, Cluster based Undersampling, Tomek links and Undersampling with ensemble learning. The process of elimination of majority class instances is based on some well-defined rules. Although researchers used this type of approaches but most of the researches discourages undersampling due the risk of important data lose. As you can see in Figure 1.4 the original dataset means the data before undersampling contains a lot of majority class instances or 0 labeled instances which

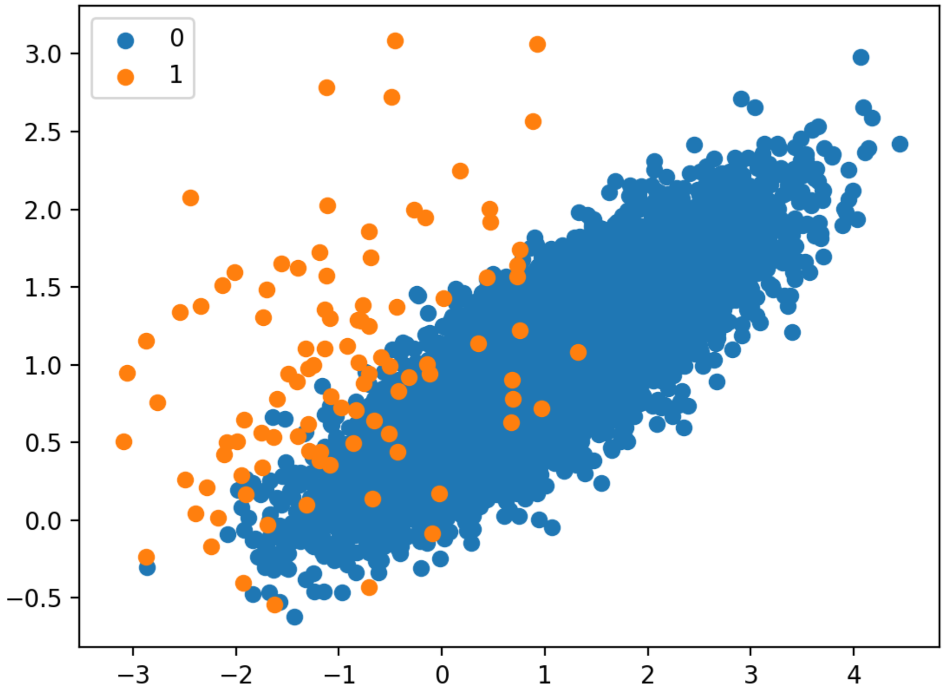

FIGURE 1.3: Original Dataset before Oversampling

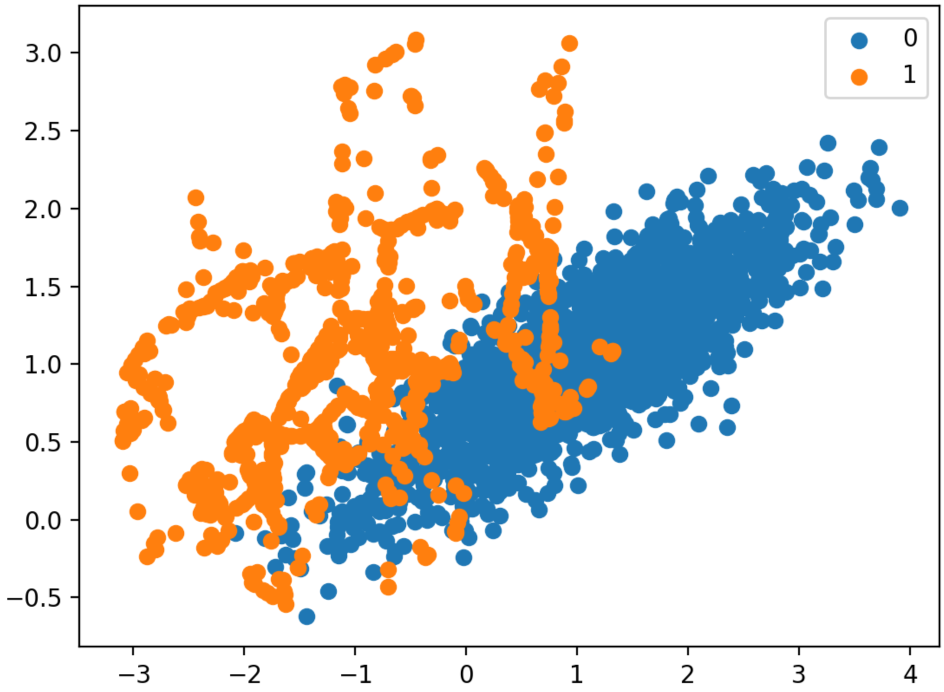

FIGURE 1.4: Balanced Dataset after Oversampling

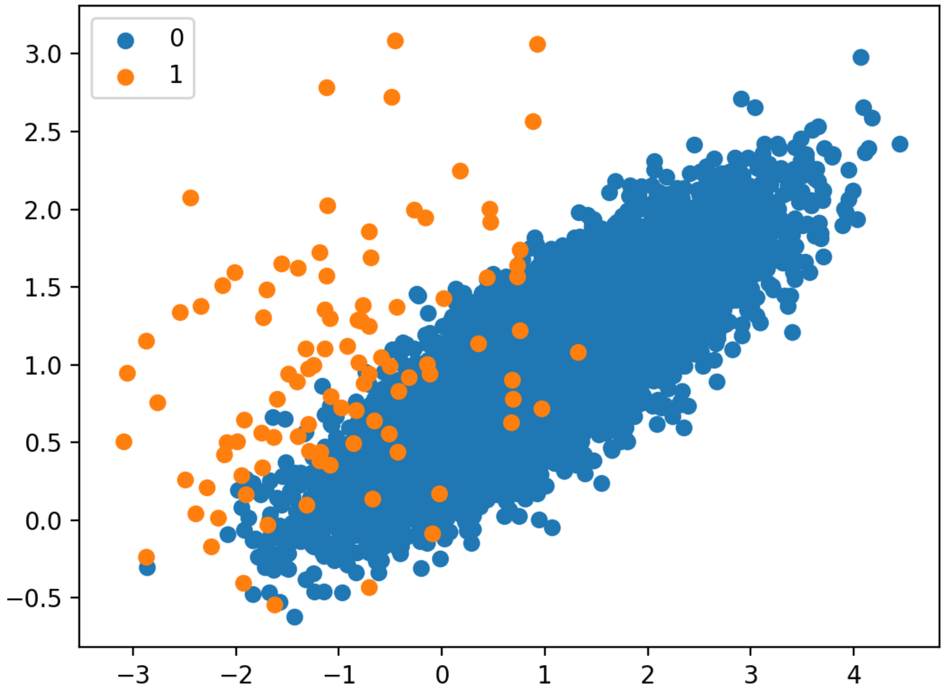

FIGURE 1.5: Original Dataset before Undersampling

is represented by blue dots while very few minority class instances or labeled 1 instances represented by orange colored dots.

After performing the undersampling a huge amount of instances eliminated from the majority class. Figure 1.5 represents the dataset after undersampling. No doubt data is balanced after undersampling but it increases the risk of potential data lose therefor most of the researchers used oversampling because it gives the robust results even in the presence of noise [9].

1.1.1.4 What is Hybrid Sampling

Hybrid sampling is basically the combination of Oversampling and Undersampling techniques. Although is gives the better solution but it's a time taking process due to the involvement of both Oversampling and Undersampling. Although is gives the better solution but it's a time taking process due to the involvement of both Oversampling and Undersampling.

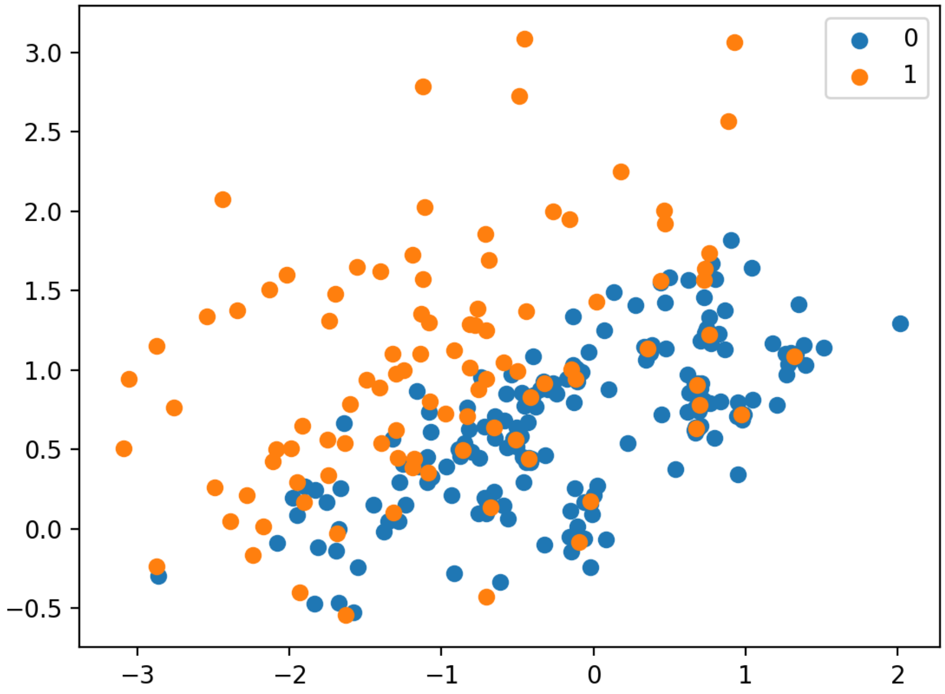

FIGURE 1.6: Balanced Dataset after Undersampling

1.1.2 Internal Approaches

Scientists introduced new supervised learning techniques in algorithmic level approaches or enhanced current systems to fix the problem of data unbalancing, with no change being made on the real data means data remains unbalanced. Internal approaches are again classified into two sub-categories one of them is Cost-sensitive techniques and the other is ensemble techniques. Scientists introduced new supervised learning in internal methods (algorithm level) or enhanced current systems to fix the problem of class imbalance, with no change being made on the original dataset. Several of these techniques sub-classes are Cost-sensitive algorithms and ensemble methods.Several of these techniques sub-classes are Cost-sensitive algorithms and ensemble methods. Several of these techniques sub-classes are Cost-sensitive algorithms and ensemble methods. Cost-sensitive model allocates various weights to each group of objects, i.e. various misclassification costs to decrease both algorithm level and data level solutions to the total costs. Secondly, the Ensemble approach is based on synchronized learning from several classification algorithms and is often utilized to improve poor learning methods to achieve success against powerful learning methods [5].

1.1.3 Hybrid Approaches

This type of approaches amalgamate the both internal and external approaches with the ensemble techniques and make one algorithm to present a best solution to tackle the unequal instances issue of a dataset.

1.2 Problem Statement

Oversampling based on Class imbalance ratio may not produce clusters with high proportion of minority objects. Such oversampling may results synthetic objects that resembles more to majority class objects which means that it produces noisy data. More over resampling within a cluster reduces the probability to generate distinct instances.

1.3 Research Questions

After finding of research gap mentioned in previous section, this thesis has formulated some of the following research questions:

- RQ1: How to avoid the impact of majority class on new generated synthetic data?
- RQ2: What would the comparison result of synthetic data be generated using classifier?
- RQ3: What would the comparison result of synthetic data be generated using clustering and classification?

1.4 Purpose

The goal of this work is to balance the data set by generating artificial instances for minority class in such a way that generation of noisy and bias data could be evaded in order to achieve equal and accurate instances for both the classes. This will further lead to accurate formation of the classification model.

1.5 Scope

Data mining is a significant sub-domain of data science. Pre-processing is a crucial step, which involves data cleaning, handling missing values, anomalies and duplications removal from the data. The outcomes of this study will assist the data scientists' community by providing them an accurate method of handling class imbalance problem.

1.6 Significance of the Solution

There is an involvement of data in almost various research studies. Research community harnesses different types of data sets to address the focused issue. Across several real-world applications, the nature of the problem sometimes indicates a considerable skew in the class distribution of a binary or multi-class classification. Therefore, the outcomes of this study will beneficiate the community to a great extent by providing them an accurate method of balancing the instances of their data sets.

1.7 Definitions, Acronyms and Abbreviations

- Synthetic Minority Oversampling Technique (SMOTE).
- Naïve Bayes (NB).

- K- Nearest Neighbor (KNN).
- Logistic Regression (LR).
- Imbalance Ratio (IR).
- Imbalance Ratio Threshold (IRT).
- Multi-Label Classification (MLC).

Chapter 2

Literature Review

Important research has been performed on addressing the issues of unbalanced learning. Most of such studies are focused on sampling. In recent years, sampling-based approaches have been shown to be very efficient, as it enables the classifier to function similarly to regular classification. In this section, I provided with imbalanced data problem a brief review of previous work and its deficiency.

In [10] Yun. et al. presented a technique named as Automatic Neighborhood size Determination (AND) that limits the amount of the SMOTE neighborhood to preserve the indigenous data distribution and allows SMOTE to achieve its highest efficiency. They claimed that Current methods use k as the standard parameter, the number of nearest neighbors. However, the most effective k value is based on the given data set, there are no guidelines for the definition of k. In addition, the current SMOTE and its derivatives exhibit low output if tones, small sub-clusters or complicated patterns are present in the datasets. They introduced AND-SMOTE to address those limitations. First they determined the suitable number of the neighborhood for each positive instance, then using AND-SMOTE create minority instances according to this size. AUC-ROC and AUC-PR were used to assess the accuracy of the classifier. Their research studies have shown that the suggested technique has performed better than SMOTE, ADASYN or Borderline-SMOTE.

Torres. et al. [6] presented a technique to introduce a new variants of smote called deterministic smote. Commonly the researcher needs to use SMOTE more than one time to pick the finest balanced data set generated, because smote created random synthetic objects. Their proposed methodology creates synthetic objects in deterministic way that produces good results as random methods but this techniques no needs to be applied several times. Their proposed technique based on 3 steps. First of all they computed number of artificial instances have to be created around each minority instance. Secondly from number of objects computed in step 1 they computed how many instances will be generated between each positive object and its k-nearest neighbor. At last uniformly synthetic objects are created between each minority instance and its k-nearest neighbor. Datasets from KEEL repository has been used for experiments. They use F-measure and AUC performance matrices and their results shows as good results as random methods.

Pruengkarn et al.[11] present a composite technique by integrating Complementary Fuzzy Support Vector Machine (CMTFSVM) and Synthetic Minority Oversampling Technique (SMOTE). They focused on imbalanced classification problem. CMTFSVM is used for undersampling and SMOTE is used for oversampling to balance the negative and positive classes of data. Basically their proposed technique consist of two sub techniques in primary approach dataset is undersampled by CMTFSVM and oversampled by SMOTE. In second approach dataset is oversampled by SMOTE then both classes undersampled by CMTFSVM to create new balanced dataset. Then these methodologies are classified by set of classification algorithms. Matrices used to compare results are G-mean and Area Under the receiver operating characteristic Curve (AUC). The presented approaches are compared with simple compliantly technique and SMOTE and are evaluated by comparing the results of three main classifiers NN, SVM, FSVM. Four datasets from KEEL, UCI and a real world dataset are used for evaluation. The performance on real world dataset are best with 0.9589 of G-mean and 0.9598 of AUC.

Lin et al.[3] proposed a technique based on two undersampling approaches to resolve class imbalance problem. They used clustering technique to make the equal count of positive and negative instances. In first approach they set cluster's amount equal to the minority class instances then by k-mean method find the centroids of each cluster and replace whole majority class instances with these centroids. In second approach they select nearest neighbor instance of each cluster's centroid and replace the whole majority class with these selected nearest neighbor instance. They used 44 small scale datasets and 2 large scale datasets. These large scale datasets was from Knowledge Discovery and Data Mining Cup. For comparison five data balancing techniques have been used named as Underbagging4, Underbagging24, Rusboost1, SMOTEbagging4 and Underbagging1. For the evaluation of new proposed technique C4.5, K-NN, SVM, Naïve Bayse (NB) and Multilayer Perceptron (MLP) classifiers have been used. Their experiments shows that their approach performed better than other and average accuracy increases. No doubt their approach performed very well but they didn't focus on features selection.

Kang et al.[5] presented a technique in which they combine noise filtering with undersampling. They focused on minority class for noise filtering and under-sampled majority class. First of all a small subset from minority and majority class is selected where majority class sample is greater than minority class and combine it. Finally they used the KNN filter to search the nearest neighbors of positive instances and to classify them into three categories of pretty effective instance: all KNN are positive instances. Comparatively useful example: KNN includes both positive and negative instances. Noisy example: all KNNs are negative objects. Then they remove the noisy one where all KNN are majority data points. After removing the noise from minority class they take a subset of random instances equally from negative and positive class. Then combine to perform classification. They used different undersampling techniques to combining with KNN filter. Although their results proved that their proposed approach got good results but still there is a limitation that they choose random sample from majority class that may contain noise because their proposed approach only focused on minority class noise filtering.

Jiannan Wang et al. [7] proposed Active Clean technique for continuous and iterative cleaning in statistical modeling issues. Through this technique they handled dirty data by removing outliers, merging terms and standardize attributes semantically. Initially they train a model on dirty data to make a dirty model. In second step a small subset of dirty data is randomly selected for cleaning. After cleaning it the updater module update the model so that the model could move towards the expected clean model. These steps performed iteratively to get the desired model. They compared their approach with Naïve-Mix, Naïve Sampling, Active Learning and Oracle. They used two metrics to compare these approaches one is distance between trained model and true model and second is prediction correctness of the model. They used datasets of UCI Adult, UCI EEG, MNIST, IMDB, and Dollars for Docs their results showed that their model had given accuracy up-to 2.5x for the same number of instances of cleaned data. They have given results in detail but their work is limited to only statistical modeling issues.

In [12] Lee et al. proposed an extension of SMOTE. It's true that SMOTE is a most popular oversampling technique but they claimed that because SMOTE chooses nearest neighbor randomly so if wrong nearest neighbor will be chosen it may causes over fitting or under fitting problem. To avoid the former problem their technique reject wrongly chosen nearest neighbor. The proposed method generates synthetic data by taking their location into account. If the synthetic data generated is behaved as noise data, it will be rejected by seeing its rejection level. Level of rejection is specify by the number of minority objects with its nearest 5 neighbors. If the rejection level value is greater than or equal to 3, they determine that the synthetic data is generated appropriately. Otherwise they reject the synthetic data being produced. Precision, recall and G-mean used as performance matrices. They used 8 datasets from UCI repository. G-mean value increased for all datasets.

In [8] José A. et al. presented their idea by highlighting the noisy and border examples problem in imbalance dataset produced by over sampling techniques. To overcome this problem their presented technique introduced a new version of SMOTE by enhancing SMOTE through a new element, an iterative ensemble based noise filter called Iterative-Partitioning Filter (IPF). Their technique consist of 2 steps, in first step synthetic data is generated by using smote then in step 2 IPF is applied to remove noisy and borderline examples. Through several iterations, IPF reduces noisy instances before a stop condition is met. The iterative process ends when throughout the iterations the amount of noise level among the instances is reached around a percentage p of the size of the original training dataset for a number of consecutive iterations (k). Datasets from the KEEL server were used. The AUC test values have shown that their approach works significantly better when working with unbalanced data sets having noisy and inconsistent instances with both virtual and real-world datasets.

Pruengkarn et al. [13] proposed technique Complementary Fuzzy Support Vector Machine (CMTFSVM) for handling outliers and noise problems in field of classification. In this approach first of all they used Complementary technique to the falsity data then classify the both truth and falsity data using Complementary Neural Network (CMTNN) and CMTFSVM. In CMTFSVM exponentially decaying membership function and radial basis function kernel has been used. After that these falsity and truth datasets are compared and new training dataset is generated. At the end new dataset is classified by FSVM and NN. The results are compared of both classifier and showed that CMTFSVM technique gives better result than CMTNN. They used the datasets from UCI and Keel. The adverse point is that the number of examples (instances) are so small to draw a generic conclusion.

In [14] Dina Elreedy et al. perform the first SMOTE mathematical survey. An assessment of the distribution characteristics of the synthetic samples produced by SMOTE is presented in this work. This aims to analyze the standard of the data in terms of how well the data produced emulates the actual underlying distribution. In our research, we are testing how the oversampled patterns vary. It is important to focus on the core components of the distribution, because the divisions of classes directly affect the shape of the boundary classification. They used SMOTE and its 2 modified versions for this survey. They used G-mean metric to perform analysis. Through experiments they found that SMOTE is an effective approach that produces additional minority class examples to try to meet the majority class's dataset, in order to counter the existing disparity. SMOTE is also a standard approach which can be used to classify a small data for standard balanced classification problems.

Piri et al. in [15] proposed a minority oversampling technique. They claimed that positive class sampling is most common and effective solutions in the learning of unequal dataset. Their study proposed a new synthetic informative minority oversampling (SIMO) algorithm leveraging support vector machine (SVM). According to their methodology they performed two steps. Firstly, they applied SVM to the native unequal dataset. Secondly, after the 1st step positive instances near to the SVM judgment boundary over-sampled. Moreover, they modified SIMO in another way and named it weighted SIMO (W-SIMO). W-SIMO differs from SIMO to the extent of oversampling of the insightful positive instances. In W-SIMO, insightful misclassified positive instances are over-sampled with a greater extent contrasted to the truly predicted most important positive objects. In this manner greater emphasis is placed on cases of unjustly classified positive data points. They implemented these techniques to the 15 freely available standard imbalanced datasets and measured their results relative to current methods in the field of imbalanced class learning. The evaluation revealed that in all datasets our approaches had the good result over all other methodologies.

A. Rivera et al. [16] introduces a priory synthetic over-sampling technique by modifications in SLOUPS and OUPS. They uses propensity scores as SMOTE uses Euclidean distance measure. In this work they claimed that by using propensity scores extra feature spatial details can help increase sensitivity scores. Using PSM, the researcher will effectively sample the most similar participants from different groups and thus reduce the impact of imbalanced groups when evaluating treatment results. The modified SLOUPS based techniques resulting in the highest average sensitivity and G-mean measures overall and performed well with SVM based learners. However after modification to OUPS performance is not as effective as the other methods did performed.

In [17] X. Zhang et al. proposed a technique k Rare-class Nearest Neighbor or KRNN, by changing the KNN induction bias directly. Generally they make two major contributions in this research. Primarily contrary to current re-sampling and cost-sensitive learning approaches, which are regional strategies targeted at inter-class imbalances, we suggest native techniques to specifically change KNN's induction bias to counter the inter-class discrepancy of positive data sparsity. Secondly they suggested an instinctive, easy approach to estimating positively-dependent future class probability dependent on positive distribution in native areas, which does not necessitate future learning. To test the KRNN efficiency, they performed comprehensive studies on thirty real-world and feigned datasets. Our results indicated that KRNN substantially enhanced KNN for uncommon class classification, and also re-sampling and cost-sensitive learning techniques were performed with generality-oriented base learners. In [18] researchers used random forest classifier to remove the noisy instances before added them to the minority class. But it works properly if the trained data is not skewed or imbalance otherwise the resulted instance wrongly classified due to biasness.

In [19] author introduced a class based global weighting scheme (Global Imbalance Handling Scheme or GIHS) to reduce the class imbalance negative effects. To tackle the formerly described problem the specified class weighted kNN classifier can prove to be effective. Class-specific weights may be used to amplify increases in the number of representatives from the positve classes in a test point neighborhood while reducing numbers from the majority classes in order to account for their excess. The authors introduced the way of assessing the classes is by using a regional weighting system. This system defines a set of class weights that is same for all check points. Since the number of delegates from each class would preferably be equal. By performing extensive experiments on different datasets they showed that their results are better than competitors.

Xie, Z. et al. in [20] proposed a synthetic minority oversampling method based on local densities in low-dimensional space named as MOT2LD. Present synthetic oversampling suffers from the curse of dimension because they largely depend on the Euclidean distance. In this work they present another way of oversampling. Their proposed methodology depends on two phases. Firstly MOT2LD maps each training sample into a low-dimensional space, and renders the low-dimensional portrayals clustered. Secondly it allocates weight as the sum of two quantities to each minority sample: local minority density and local majority count, suggesting that is produces significant samples. For certain positive instance's clusters the synthetic minority class samples are produced. MOT2LD was tested on 15 sets of real-world results. The experimental results showed that with regard to the G-mean and F-measure, our system outperforms several other existing approaches like SMOTE, Borderline-SMOTE, ADASYN and MWMOTE.

Owing to the relative nature of misclassification of unusual cases, conventional approaches which are skewed against the dominant class are ineffective. Lim, P., et al. in [21] proposed a new, cluster-based oversampling ensemble system, which is used to generate synthetic data points on the bases of clusters with an evolutionary algorithm (EA) to construct a band. The approach presented to generate synthetic data is based on modren ideas for defining oversampling regions using clusters. So first of all they accurately define data space by making clusters then in second phase they generate synthetic data using two strategies one is generating new instances using cluster centers and the other one is using random data points within the cluster. The novel use of EA performs a double objective of optimizing data generation process parameters while generating numerous examples using the characteristics of EAs, reducing overall computational costs.

Siriseriwan, W. et al.[22], in their work introduces a new adaptive algorithm called Adaptive neighbor Synthetic Minority Oversampling Technique (ANS). It adjust dynamically the number of neighbors needed for oversampling through numerous positive regions. The aim of this research is to set measure for selecting parameter K value. In addition, this paper also deals with how to manage a positive instance surrounded by negative instances in order to use a positive instance for enhancing accuracy. This optimistic example is known as an outcaste minority. Based on our experiments in UCI and PROMISE datasets, data sets produced from this methodology have improved a classification's accuracy. And statistically the Wilcoxon signed-rank test can validate the improvement. However, this requires a classifier to learn very particular cases, not their general property, which therefore leads to an over fitting problem.

In [23] Liang, Y. et al. This paper proposes a changed (MSMOTE) approach to learning from Imbalanced data sets dependent on SMOTE. MSMOTE considers not just the allocation of minority party samples but also the reduction of noise samples by adaptive consultation. MSMOTE and AdaBoost together are added to many strongly and mildly imbalance data sets. The Experimental findings indicate the accuracy of predictions of MSMOTE for the minority class is greater than SMOTEBoost and F-values are rising, too.

For an imbalance dataset it is difficult to construct accurate classifiers for forecasting class participation as the classifier appears to be skewed against the over-represented or dominant class as a result. Rivera, W.A. et al. in [24] proposed a new methodology to combat this problem. The main contribution or newness of this proposed methodology is to remove noise from the positive class before applying oversampling technique. Experiments are performed through a wide spectrum of data sets, learners, and methods of sampling. The findings of this new technique show progress over the comparative methods for the Sensitivity and G-mean scales.

In [25] Gosain, A. et al. presented a technique to tackle imbalance problem of real world datasets. Their technique is basically presented a slightly modification in SMOTE named as Farthest SMOTE or FSMOTE. According to this technique new instances are created with the line that joins the positive instances and its k afar neighbors of positive class. A data point is randomly selected from the k-farthest neighbor of positive class instances and then by using the selected farthest neighbor new instances are generated. The suggested FSMOTE approach raises the region of judgment from which the minority samples are found near to the boundary. By using the two standard classifiers, namely Naïve Bayes and SVM, on seven publicly accessible datasets we have contrasted the efficiency of our proposed FSMOTE methodology with oversampling approaches, namely SMOTE, ADASYN, borderline SMOTE, and safe-level SMOTE. The experimental findings indicate that our approach works better than the current techniques. Different evaluation measurements such as overall efficiency, sensitivity, precision, F-measure, geometric mean (G-mean) and receiver operating curve (ROC) area, i.e. area below curve (AUC) value, can be observed this.

The k nearest neighbor (kNN) algorithm classifies an unknown instance in the training instance space into the most frequent class of its k closest neighbors. For unequal class distribution where minority training samples are uncommon, a new instance is often outnumbered by negative instances in its neighborhood and unfairly categorized into the negative majority class. To solve this problem Zhang, X. et al. [26] proposed a new nearest neighbor algorithm called Positive-based Nearest Neighbor (PNN). In this proposed method a new instance and parameter k is given and the probability of query instance for positive class and frequency of positive instances in the neighborhood of new generated instance decides either this query instance belongs to the positive class or not. If frequency of positive instances around the query instance is very low then we increase the neighborhood for classification decision. Extensive experiments on imbalanced datasets in the real world show that PNN performs well for imbalanced classification. PNN often outperforms recent unbalanced classification algorithms based on kNN, while significantly reducing their additional cost of computing. In [27] Sharma, S. et al. claimed that existing oversampling techniques only

focused on minority class to generate new instances which is not fruitful if imbalance ratio is very high. Their proposed technique gives the solution to tackle this problem by using the information inherited from the majority class too. To fulfill their idea they used Mahalanbois distances to generate new data. This generated data would have the same Mahalanbois distance from majority instances as the identified minority instances. The evaluation of their technique over the 26 benchmark datasets gives the good performance improvement over the existing standard techniques. However their proposed idea is mostly useful only in extreme imbalance ratio because if imbalance ratio is low then minority class information is enough to generate new synthetic data.

A technique proposed by Douzas at al [4] focused to improve classification by balancing the imbalance data. This research provides a straightforward and efficacious strategy of over-sampling based on clustering of k-means and SMOTE (synthetic minority oversampling technique), which prevents to generate noisy instances and effectually reduces the effect of imbalance between and within groups.

K-means SMOTE is basically included 3 steps: first is to make clusters of data, second is to filter these clusters and third is to over-sample filtered clusters. In the clustering step, using k-means clustering, the whole dataset is grouped into k clusters. The filtering step selects oversampling clusters that retain those with a large amount of samples in the minority class. Then it provides the number of generated instances to be produced, allocating more instances to clusters where positive samples are sparsely distributed. At the end, SMOTE is implemented to get the wanted ratio of positive and negative instances in each selected cluster in the oversampling stage. Figure 2.1 shows the steps of this technique 12 imbalanced dataset from UCI and 19 from Keel repository have been used to evaluate proposed methodology on the basis of g-mean, F1-score and AUPRC. For comparisons different oversampling techniques were used like random oversampling, SMOTE, borderline-SMOTE1, borderline-SMOTE2, K-mean SMOTE and no oversampling (imbalance dataset). Their results shows that they achieved higher results as compared to other techniques but they used all datasets on small scale.Their results shows that they achieved higher results as compared to other techniques but they used all datasets on small scale.Their results shows that they achieved higher results as compared to other techniques but they used all datasets on small scale.

Conclusively, a lot of recent research studies have been carried out focusing on the issue of imbalanced data sampling. The undersampling based approaches of handling imbalanced data are not recommended due to potential data loss [3][4]. On contrary, oversampling based approaches are preferred over undersampling approaches. Some of the oversampling based approaches have used clustering techniques prior to random sampling or SMOTE sampling. A recent approach [3] uses k-mean SMOTE in which clusters for oversampling are selected on the basis of IR, then using SMOTE technique, synthetic data is generated to bring the minority class equal to majority class. We argue that such a clear representation of the clustering presented in [3] cannot be obtained based on imbalanced data. Such clustering may include clusters with high proportion of majority class such oversampling may produces new objects that resemble to majority class having class label of minority class (noisy instances).

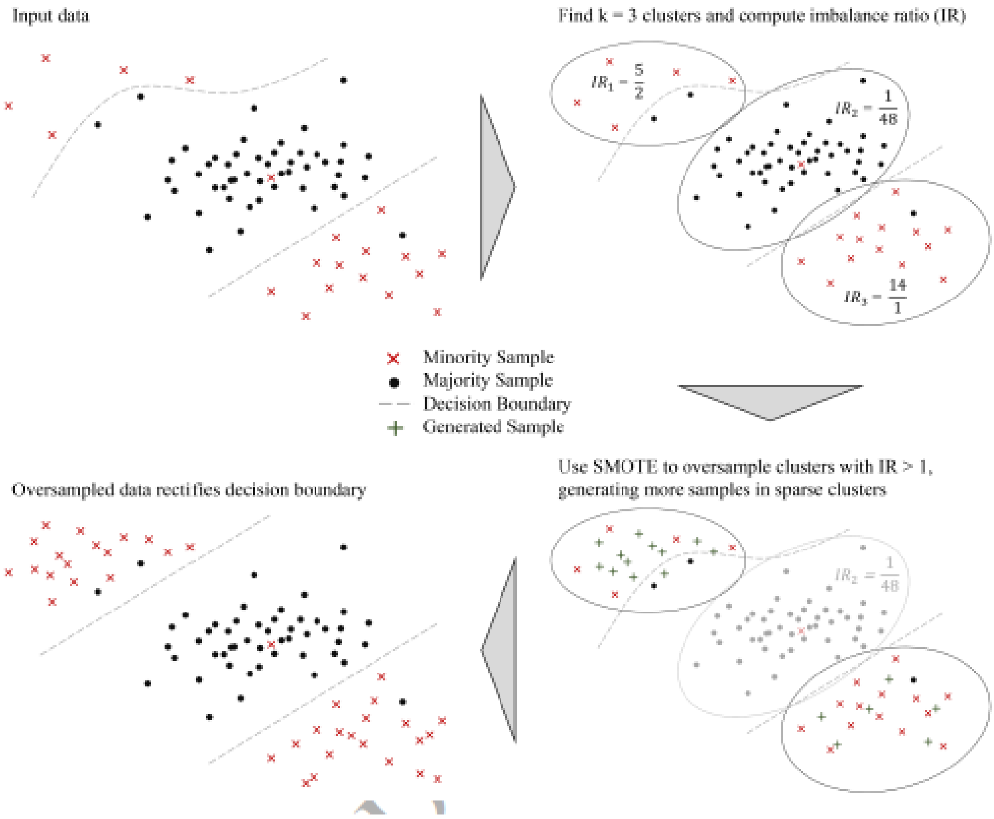
Input data
Find k = 3 clusters and compute imbalance ratio (IR)
$IR_1 = \frac{5}{2}$
$IR_2 = \frac{1}{48}$
$IR_3 = \frac{14}{1}$
Minority Sample
Majority Sample
Decision Boundary
Generated Sample
Use SMOTE to oversample clusters with IR > 1, generating more samples in sparse clusters
Oversampled data rectifies decision boundary
$IR_2 = \frac{1}{48}$

Figure 2.1: Oversampling Steps of k-means SMOTE

2.1 Evaluation Criteria

After detailed study of literature view of past papers related to sampling techniques our findings are based on some parameters. To explain these findings we will use parameters like datasets, algorithm or technique, results and limitations. In dataset we talk about type of dataset and quantity of dataset. Algorithm or technique parameter give us the information that which techniques were used to balance the dataset and after balancing which algorithm is used for classification. In results we talk about the implemented technique's results on the base of different evaluation measures like G-mean, F-measure and accuracy. In limitations we talk about the major drawbacks of previously implemented techniques that we observed during detailed literature view.

2.2 Evaluation Measures

Though binary classification and state-of-art evaluation measures have been generally agreed in order to define that how the classification algorithms performed, most of them are not ideal in context of data imbalance classification since the performance of the majority class is over-represented. In unbalanced research, the key objective is to enhance the classification of positive instances while preserving a fair output as for the rest of majority class. According to the study of previous research papers we selected 3 evaluation measures G-mean, F1-measur and accuracy to build and compare our classification results. Implementation of the TP, TN, FP and FN notations for the number of true positive, true negative, false positive and false negative samples; So P=TP+FN, N=TN+FP, the steps chosen are set as follows.

2.2.1 G-mean

The geometric mean value is defined as the geometric mean of sensitivity and specificity. The 2 elemnts may be considered as being accurate per class. The g-mean presented value in [0, 1], all metrics are aggregated into a single measure, giving equal weight to each class. Sensitivity is the other name of recall or true positive rate that gives the explanation about accuracy of prediction for the instances of minority class. This gives solution to the query that how many objects of positive groups were correctly listed as such? For the majority class the specificity addresses the same question [4].

$$G - Mean = \sqrt{(sensitivity * specificity)} = \sqrt{((\frac{TP}{P}) * \frac{TN}{N})}$$

2.2.2 F1-Score

The (often weighted) harmonic mean of precision and recall is the F1-score, or F-measure. In other words, the measure assesses both the completeness and accuracy of positive predictions [4].

$$F1 = \frac{(1+\alpha) * (Sensitivity * Precision)}{Sensitivity + \alpha * Precision} = \frac{(1+\alpha) * (\frac{TP}{P} * \frac{TP}{PP})}{\frac{TP}{P} + \alpha * \frac{TP}{PP}}$$

2.2.3 Accuracy

Accuracy is the most popular metric for classification problems. For imbalanced samples these measures display a bias against the dominant party. For example, in a data set where only 1 percent of instances are positive, an ingenuous classification algorithm which forecasts all samples as majority instance will attain 99 percent accuracy. Although such high performance indicates an effectual classification algorithm, the formula conceals the fact that no single positive object has been accurately forecasted.

$$Accuracy = \frac{TP + TN}{P + N}$$

2.2.4 Area Under the Curve

To a large degree, the choice of metric depends on the target that consumer is attempting to accomplish. In some practical tasks, one particular aspect of classification may be more important than another (for example, in medical diagnosis, false negatives are far more crucial than false positive ones). But no such priority should be put in order to establish a general rating of over samplers.But no such priority should be put in order to establish a general rating of over samplers.

2.3 Critical Review of Oversampling Techniques

The brief overview of most related technique to our work is described in Table 2.1 below along with their methodology, results, limitations, publication year and dataset information.

TABLE 2.1: Critical Analysis Table of Literature

Ref	Technique	Dataset	Results	Limitation
[4] 2014	Two undersampling techniques by clustering majority Approach 1: replace all clusters with centroids Approach 2: replace all clusters with nearest neighbor of centroid	44 small scale datasets and 2 large scale datasets	Average accuracy of approach 1= 0.85, Average accuracy of approach 2= 0.86	Under-sampling is risky as potential data points could be loss
[5] 2018	Method of over-sampling based on clustering of k-means and SMOTE, Made up of three Steps: 1) Clustering 2) Filtering 3) Over-Sampling	12 imbalanced datasets from UCI and 19 from keel repository	F-Measure is 0.85 Maximum average G-mean is 0.85 accuracy is 0.86	Ambiguous clustering Ambiguous l tuple
[6] 2016	Presented a technique in which they combine noise filtering with undersampling, Focused on minority class for noise filtering and under-sampled majority class	Datasets from UCI and KEEL repository	G-Mean= 0.612 F-measure= 0.029e AUC=0.665	Selection of random sample from majority class may contain noise because their proposed approach only focused on minority class noise filtering Only Work for lower IR

Ref	Technique	Dataset	Results	Limitation
[17] 2015	Supervised-SMOTE uses random forest by default and removes noise before sample generation	104 Datasets from UCI and KEEL repository	Average F-measure of all datasets= 0.6723	Works properly only if the training data of the random forest is balanced, Imbalance ratio should be low
[7] 2016	Deterministic oversampling, Synthetic objects are created uniformly	33 datasets from keel	Average F Measure is 0.64	Synthetic data will be uniform
[24] 2019	Synthetic oversampling, with the majority class	26 datasets from UCL	Average G Means is 0.79	It works only if IR is very high it may produces noisy instances
[19] 2015	Adaptive neighbor Synthetic minority Oversampling Technique under 1NN outcost handling Adjust dynamically the number of neighbors needed for oversampling through minority regions	13 datasets from UCL Promise repository	Average F measure is 0.62	Requires a classifier to learn very particular cases Leads to over fitting problem

After the above critical analysis of past papers in imbalance dataset domain we come up with our best knowledge that most of the researchers done their work by using oversampling techniques. Oversampling technique's main focus is to produce new artificial data instances on the basis of knowledge provided by present minority instances without losing and effecting any information of the original. A research conducted in 2017 reveal that oversampling also work robustly in noisy environment. In their research they used SMOTE (oversampling) and RUS (undersampling) techniques. They conducted their survey on the synthetic dataset containing 70% noise [28]. Moreover we also find out that many researchers used SMOTE in different ways to balance the data. But most of the techniques gives best results for low imbalance ratio and some of them generate noisy data. I also used SMOTE in my work but with the main focus on improving the results for low imbalance ration and tried to generate data that have lesser possibility to effect by negative class and closely related to positive class.

Chapter 3

Research Methodology

The main aim of every form of resampling is enhancing the performance of the classification. We can say that, a resampling method is effective if the synthetic data that it generated increases a given classifier's predictive efficiency. Consequently, the potency of an oversampling process can be evaluated implicitly by testing a classification algorithm learned on oversampled records. This surrogate metric, i.e. the productivity of the classification algorithm, is only useful when contrasted with the results produces by the same classification method learned on the imbalance dataset. It is then possible to classify numerous oversampling strategies by assessing the achievement of the classification model as regards to each redesigned training set generated by the re-samplers.

The error in making predictions for previously observed data is a general problem in the application of the classifiers. Classification algorithms can do well in predicting instances used during training, but its performance goes down when trained model used to classify new instances. The former described problem also named as over fitting problem. It has been noted that oversampling techniques invigorated over-fitting that's why we should do evaluation carefully to avoid overfitting [4].

Generally, we do is to split data into two parts one is training data and the other one is test data. But there is a problem in this approach that the splitted training set could miss potential instances and classifier misses to learn on them that would badly effect accuracy and efficiency. So to tackle this problem one most common approach used by researchers is K-fold cross validation. In cross validation data is splitted K times one after another which means that every time instances in training set will be change in this way classifier get chance to learn almost all instances. At the end average of these K iterations is the final result[4][28]. The value of K in cross validation could be different for different researchers but we used 5-fold cross validation as our base paper used [4].

3.1 Evaluation Measures

As we described in chapter 2 that binary classification and state-of-art evaluation measures have been generally agreed in order to define the performance of the classifiers, all of them are not ideal to imbalanced situations since the performance of the majority class is over-represented. In unbalanced research, the key purpose is to enhance the classification of (positive) minority instances while preserving a fair output as for the rest of majority class. According to the study of base research paper [4] we have selected 2 evaluation measures G-mean and F1-measur to compare results. Additionally, we also predict accuracy, AUC and TPR of base methodology and our methodology to build and compare our classification results.

3.2 Oversampling Technique

Generally, there exist many oversampling techniques and almost all of them are modified many times. But to the best of our knowledge SMOTE is the famous due to its simplicity and good performance. Moreover, the base paper selected as a benchmark also used SMOTE to generate instances. For this reason we used SMOTE. Although one of our technique is based on our base paper methodology K-means SMOTE with a prominent change but parameters for K-means and SMOTE we have used is same. For SMOTE K-means Knn will be 3, 5 and 20 and for K-means K will be 2, 20, 50 and 100.

3.2.1 What is SMOTE

A wide variety of techniques to oversample a set of data that is used in a standard binary classification. All of them, SMOTE: Oversampling Synthetic Minority Technique proposed in 2002 by Chawla et al. [29], is used by most of the researchers due to its simplicity and good results. Moreover, over-fitting problem which comes during the random oversampling is avoided by SMOTE. It does not create the copy of existing minority instances instead of replicating it creates synthetic data. The process of generating artificial samples is done by choosing a random positive class instance and one of its nearest neighbor positive class instance in linear fashion. Mainly, the process of generating synthetic data SMOTE consists of 3 steps.

(i) Selecting random instance ‘a’ of positive class

(ii) Selecting randomly a positive class instance ‘b’ from its k-nearest neighbor

(iii) An artificial instance ‘x’ is created between these two selected instances ‘a’ and ‘b’ by using the following formula.

$$x = a + w * (b - a)$$

Here, w is a random weight between 0 and 1. The process of generating artificial instance is diagrammatically illustrated in Figure 3.1.

3.3 Classifiers

Several separate classification algorithms are used for the assessment of the various oversampling approaches to make certain that the produced results can be extended and are not limited to the use of a single classification algorithm. Therefor to select a classifier is an important task of research. We use K-Nearest Neighbor (K-NN) and Logistic Regression (LR) as our base paper used these two classifiers. Logistic regression (LR) is a simplistic linear classifier that is used by research communities to rank binary data. Model fitting is an issue of optimization that

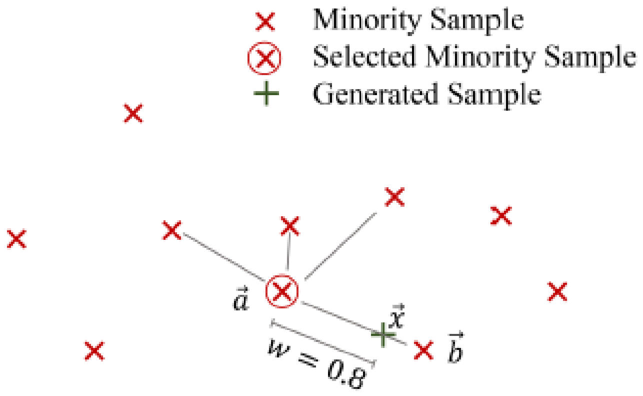

FIGURE 3.1: Process of SMOTE to Generate Artificial Instance with its 4 Nearest Neighbors

can be fixed with simple optimization techniques that do not require parameters of the model to be specify [4][30].

One other classifier called k-nearest neighbors (KNN) allocates an occurrence to the entity of which most of its closest neighbors belong. K parameter of the model decides that how many instances are used as a neighbor [4][31][5, 28].

In one of our approach we use Naïve Bayes (NB) to calculate the probability of new generated instances. Naïve Bayes is a simplest approach used to calculate probabilities and also gives the highest results of precision, recall and F1-score for minority class when applied its trained model on instances generated by SMOTE [30].

3.4 Dataset

To asses our approaches on fair bases we used same datasets which is used by K-means SMOTE [4]. These are 10 datasets from UCI machine learning repository and 12 dataset from KEEL repository. Originally these datasets were not contain class label in binary form so firstly they change class label in binary form by using one-versus-approach. All changed datasets are publically available on https://github.com/AlgoWit/publications. Moreover, imbalance ratio is increased by removing some minority samples from all datasets one by one so that on high imbalance ratio results could be seen [4].

3.5 Experimental Framework

To perform the experiment we used all defined measures, datasets, classification algorithms and oversampling techniques. We used 5-foldes cross validation for each dataset. For every measure the final result we got is the average of these 5 values that is calculated during each iteration of 5-foldes.

3.6 Proposed Methodology

In this thesis we have proposed two approaches one is K-means Minority Class Clustering and other is named as Oversee SMOTE for oversampling. One is based on our base paper [4] methodology with a minor change in its clustering step and the other is proposed without clustering and used Naïve Bayes classifier probability to predict that either new generated data point belongs to minority class or majority class. Now the detailed steps of both strategies are given blow one by one.

3.6.1 K-means Minority Class Clustering

As it is already described that this methodology is built on K-means clustering [4] so it also included 3 steps. First is to make clusters of minority class instances, second is to filter these clusters and third is to over-sample filtered clusters. In the clustering step, using k-means clustering, the whole dataset is grouped into k clusters. The filtering step selects oversampling clusters that retain those with a large amount of samples in the minority class. Then it provides the number of generated instances to be produced, allocating more instances to clusters where positive samples are sparsely distributed. At the end, SMOTE is implemented to get the wanted ratio of positive and negative instances in each selected cluster in the oversampling stage.

K-means is a well-liked iterative algorithm which is used to find natural locating classes in dataset that can be expressed in a Euclidean space. It tends to work by repeating two commands in iterative manner: initially, allocate each sample to the closest cluster centers of the k cluster. Second, change the cluster centers location such that they are aligned within their allocated instances. Figure 3.3 illustrate above explained methodology.

3.6.1.1 Algorithm Steps

(i) Separate the minority and majority class.

(ii) Clustering of minority class using k-mean clustering by finding an appropriate k.

(iii) Assign weights to each cluster.

  (a) For each cluster c, calculate the Euclidean distance matrix

  (b) Calculate the average distance within each cluster by adding all non-diagonal values of the distance matrix, after that divide this average distance by the count of non-diagonal cells.

  (c) Calculate density by using

$$Density(C) = \frac{minoritycount(C)}{avg.minoritydistance(C)^m}.$$

  (d) Calculate sparsity by using

$$Sparsity(C) = \frac{1}{Density(C)}.$$

  (e) The sampling weight of each cluster is calculated by

$$Sampling\_weight(C) = \frac{Sparsity(C)}{\sum Sparsity(C)}.$$

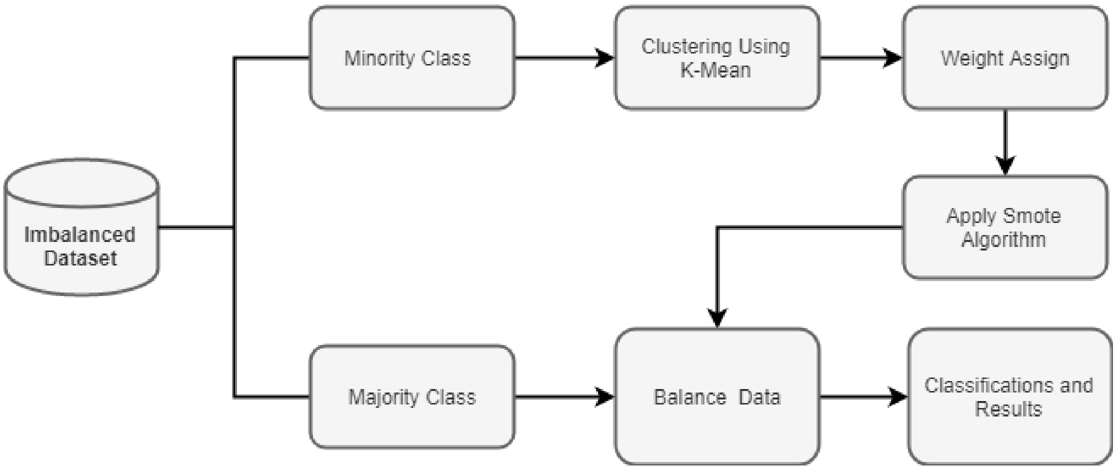

FIGURE 3.2: Minority Class Clustering SMOTE Methodology Diagram

(iv) Apply SMOTE to generate instances equal to (sampling weight (c) X n) samples, where n is the overall number of samples to be generated.

3.6.2 Oversee SMOTE

This proposed approach is distinct from others because it involved a classifier in the process of resampling. It does not use clustering to handle class imbalance problem. Instead of clustering synthetic data will be produced and added to the minority class on the basis of probability obtained using Naïve Bayes classifier. Naïve Bayes is a simplest approach used to calculate probabilities and also gives the highest results of precision, recall and F1-score for positive instances when implemented its trained model on instances generated by SMOTE [30]. In both approaches, SMOTE will be used to create new instances. Figure 3.3 illustrate the steps of this methodology.

3.6.2.1 Algorithm Steps

(i) Select majority class randomly equal to minority class and combine both classes.

(ii) Apply Naïve Bayes to train it on the selected data in step one.

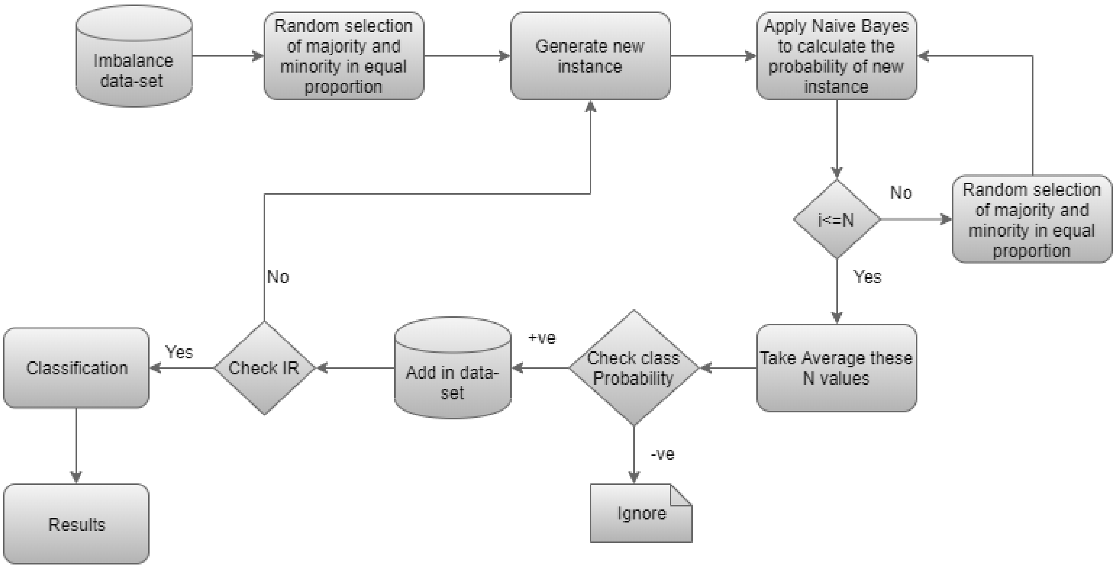
Imbalance data-set
Random selection of majority and minority in equal proportion
Generate new instance
Apply Naive Bayes to calculate the probability of new instance
i<=N
No
Random selection of majority and minority in equal proportion
Yes
Take Average these N values
Check class Probability
+ve
-ve
Ignore
Add in data-set
Check IR
No
Yes
Classification
Results

FIGURE 3.3: Oversee SMOTE Methodology Diagram

(iii) Generate new instance and give it to train model and save its probability.

(iv) Generate new instance and give it to train model and save its probability.

(v) Repeat above 3 steps N times.

(vi) Take average of theses N values and check probability status.

- (a) If it belongs to the positive class specified range add this instance to minority class.
- (b) Else ignore it.

(vii) Repeat above 5 steps for every instance to get the minority data point equal to the majority class.

(viii) At the end give this balance dataset to classifier to get results.

Generated instances' probability is check by defined upper and lower threshold. This strategy gives us a fair minority instances because only those instances will be added which have probability close to the threshold defined for the minority class so majority class effect will reduce on the synthetic data. The threshold range we defined for minority class is 0.5-1. Moreover, we defined that probability calculation step will iterate 5 times for each generated instance and in each iteration Naïve Bayes model will train on new selected data. This concept is useful in reducing the biasness of training data and every instance get chance to train model including new generated instances. This fact increases the chance that the new generated data is fairly belongs to the minority class.

Chapter 4

Results and Evaluation

In the previous chapter we have explained the in-depth details of the proposed methodology. This chapter presents the details about the results that have been obtained by applying the proposed methodologies. Comparative results of some datasets are shown in table below.

Table 4.1 shows the results of F1-measure. A significant improvement of our proposed approaches on most of the datasets can be seen throughout the results. Table 4.2 shows the results of G-mean. It can be observe that no doubt in few datasets K-means SMOTE performed very well and in some datasets it gave same results as our proposed approaches but overall MCC SMOTE and Oversee SMOTE outperformed it. Table 4.3 shows the results of accuracy. It can be observe that no doubt in few datasets K-means SMOTE performed very well and in few datasets it gave same results as our proposed approaches but overall MCC SMOTE and Oversee SMOTE outperformed it.

Now we discuss the above results in detail with their graphical representation.

4.1 Results of MCC SMOTE

As we already discussed previously that we will build our results in terms of G-mean, F1-measure and accuracy. We run our base paper's technique K-means

Table 4.1: Results of F1-measure with k-means SMOTE, MCC SMOTE and Oversee SMOTE

Dataset	**IR**	**Classifier**	**K-means SMOTE**	**MCC SMOTE**	**Oversee SMOTE**
Breast Tissue	(70:36)	KNN	0.67	**0.76**	**0.72**
		LR	0.77	**0.84**	0.75
Cleveland	(13:160)	KNN	0.8	**0.9**	**0.82**
		LR	0.92	**0.95**	0.85
Dermatology	(20:338)	KNN	0.93	**0.99**	**1**
		LR	1	1	1
Ecoli	(52:284)	KNN	0.92	**0.96**	0.92
		LR	0.85	**0.93**	**0.92**
Eucalyptus	(98:544)	KNN	0.79	**0.84**	**0.82**
		LR	0.82	**0.91**	**0.84**
Glass	(70:144)	KNN	0.77	**0.82**	**0.84**
		LR	0.77	**0.81**	**0.82**
Haberman	(81:225)	KNN	0.76	0.76	**0.89**
		LR	0.7	**0.79**	**0.89**
Heart	(120:150)	KNN	0.64	**0.68**	**0.8**
		LR	0.79	**0.84**	**0.8**
Iris	(50:100)	KNN	1	1	1
		LR	1	1	1
Led	(37:406)	KNN	0.79	**0.92**	**0.95**
		LR	0.89	**0.92**	**0.95**
Libras	(24:336)	KNN	0.96	**0.98**	0.95
		LR	0.94	**0.98**	**0.96**
Liver	(145:200)	KNN	0.67	**0.72**	**0.78**
		LR	0.61	**0.67**	**0.78**
New Thyroid 1	(35:180)	KNN	0.92	**0.99**	**0.93**
		LR	0.97	**0.99**	0.94
New Thyroid 2	(35:180)	KNN	0.98	0.98	**1**
		LR	0.97	**0.99**	**1**
Page Blocks 1	(28:444)	KNN	0.95	0.94	**0.96**
		LR	0.97	0.97	0.96
Pima	(268:500)	KNN	0.77	**0.79**	**0.8**
		LR	0.79	**0.84**	**0.89**
Vehicle	(199:647)	KNN	0.93	**0.96**	**0.96**
		LR	0.96	**0.98**	**0.96**
Vowel	(90:898)	KNN	0.89	**0.97**	**0.93**
		LR	0.95	**0.98**	0.93
Wine	(71:107)	KNN	0.68	**0.77**	**0.89**
		LR	0.93	**0.95**	**0.96**
Yeast 1	(429:1055)	KNN	0.79	**0.81**	**0.8**
		LR	0.79	0.78	**0.92**

Table 4.2: Results of G-Means with k-means SMOTE, MCC SMOTE and Oversee SMOTE

Dataset	**IR**	**Classifier**	**K-means SMOTE**	**MCC SMOTE**	**Oversee SMOTE**
Breast Tissue	(70:36)	KNN	0.7	**0.75**	**0.77**
		LR	0.77	**0.86**	0.76
Cleveland	(13:160)	KNN	0.78	**0.81**	**0.8**
		LR	0.93	**0.97**	0.84
Dermatology	(20:338)	KNN	0.98	**0.93**	**1**
		LR	1	1	1
Ecoli	(52:284)	KNN	0.92	**0.99**	**0.94**
		LR	0.86	**0.93**	**0.94**
Eucalyptus	(98:544)	KNN	0.78	**0.96**	**0.83**
		LR	0.82	**0.91**	**0.84**
Glass	(70:144)	KNN	0.76	**0.94**	**0.85**
		LR	0.7	**0.8**	**0.84**
Haberman	(81:225)	KNN	0.77	**0.98**	**0.84**
		LR	0.74	**0.81**	**0.84**
Heart	(120:150)	KNN	0.65	**0.84**	**0.81**
		LR	0.8	**0.85**	**0.81**
Iris	(50:100)	KNN	1	1	1
		LR	1	1	1
Led	(37:406)	KNN	0.8	**0.9**	**0.94**
		LR	0.89	**0.92**	**0.94**
Libras	(24:336)	KNN	0.96	0.8	0.95
		LR	0.94	**0.98**	**0.95**
Liver	(145:200)	KNN	0.69	**0.83**	**0.82**
		LR	0.63	**0.68**	**0.79**
New Thyroid 1	(35:180)	KNN	0.9	0.85	**0.94**
		LR	0.97	**0.99**	0.94
New Thyroid 2	(35:180)	KNN	0.98	0.93	**1**
		LR	0.98	0.98	**1**
Page Blocks 1	(28:444)	KNN	0.96	0.82	**0.97**
		LR	0.97	**0.98**	0.97
Pima	(268:500)	KNN	0.76	**0.79**	**0.81**
		LR	0.8	**0.83**	**0.9**
Vehicle	(199:647)	KNN	0.91	0.79	**0.97**
		LR	0.9	**0.98**	**0.97**
Vowel	(90:898)	KNN	0.96	0.77	0.95
		LR	0.95	**0.98**	0.95
Wine	(71:107)	KNN	0.69	**0.84**	**0.9**
		LR	0.9	**0.95**	**0.96**
Yeast 1	(429:1055)	KNN	0.8	**0.82**	**0.82**
		LR	0.81	0.79	**0.94**

Table 4.3: Results of Accuracy with k-means SMOTE, MCC SMOTE and Oversee SMOTE

Dataset	IR	Classifier	K-means SMOTE	MCC SMOTE	Oversee SMOTE
Breast Tissue	(70:36)	KNN	0.69	**0.77**	**0.77**
		LR	0.79	**0.86**	0.78
Cleveland	(13:160)	KNN	0.79	**0.8**	**0.8**
		LR	0.93	**0.97**	0.84
Dermatology	(20:338)	KNN	0.98	**1**	**1**
		LR	1	1	1
Ecoli	(52:284)	KNN	0.92	**0.93**	**0.93**
		LR	0.86	**0.93**	**0.94**
Eucalyptus	(98:544)	KNN	0.78	**0.84**	**0.84**
		LR	0.82	**0.91**	**0.84**
Glass	(70:144)	KNN	0.76	**0.85**	**0.85**
		LR	0.74	**0.8**	**0.84**
Haberman	(81:225)	KNN	0.77	**0.85**	**0.85**
		LR	0.74	**0.82**	**0.85**
Heart	(120:150)	KNN	0.66	**0.81**	**0.81**
		LR	0.8	**0.85**	**0.84**
Iris	(50:100)	KNN	1	1	1
		LR	1	1	1
Led	(37:406)	KNN	0.82	**0.95**	**0.95**
		LR	0.89	**0.92**	**0.95**
Libras	(24:336)	KNN	0.96	0.95	0.95
		LR	0.95	**0.98**	0.95
Liver	(145:200)	KNN	0.69	**0.82**	**0.82**
		LR	0.63	**0.68**	**0.83**
New Thyroid 1	(35:180)	KNN	0.89	**0.91**	**0.91**
		LR	0.97	**0.99**	0.91
New Thyroid 2	(35:180)	KNN	0.98	**1**	**1**
		LR	0.98	**0.99**	**1**
Page Blocks 1	(28:444)	KNN	0.96	0.96	**0.97**
		LR	0.97	**0.98**	0.97
Pima	(268:500)	KNN	0.76	**0.81**	**0.81**
		LR	0.8	**0.83**	**0.93**
Vehicle	(199:647)	KNN	0.93	**0.96**	**0.96**
		LR	0.96	**0.98**	0.96
Vowel	(90:898)	KNN	0.93	**0.95**	**0.95**
		LR	0.94	**0.98**	0.94
Wine	(71:107)	KNN	0.7	**0.93**	**0.93**
		LR	0.93	**0.95**	**0.97**
Yeast 1	(429:1055)	KNN	0.75	**0.79**	**0.79**
		LR	0.81	0.79	**0.94**

SMOTE clustering [4] and our proposed approach MCC SMOTE on the same data using same hyper-parameters. We run both approaches 3 or 4 times to get the best results and to sure that either results change or not [4].

4.1.1 Results using KNN Classifier

We observe that using KNN classifier our proposed approach improves results in terms of each defined measure for all datasets except 2 dataset. For 'Iris' dataset the behavior of both strategies remained same and for 'page blocks 1' dataset there is no improvement. The maximum improvement that we observed is 0.14 for g-mean, 0.12 for F1-measure and 0.15 for accuracy measure. They said that on 6 datasets their approach showed no improvement but in our case the proposed approach is outperformed on all datasets except 2 datasets. We computed average for all measures of all datasets and we got 0.89 F1-measure, 0.9 G-mean and 0.89 accuracy. Figure 4.1 shows the comparison of average results using F1-measure, G-mean and accuracy of K-mean SMOTE and MCC SMOTE techniques.

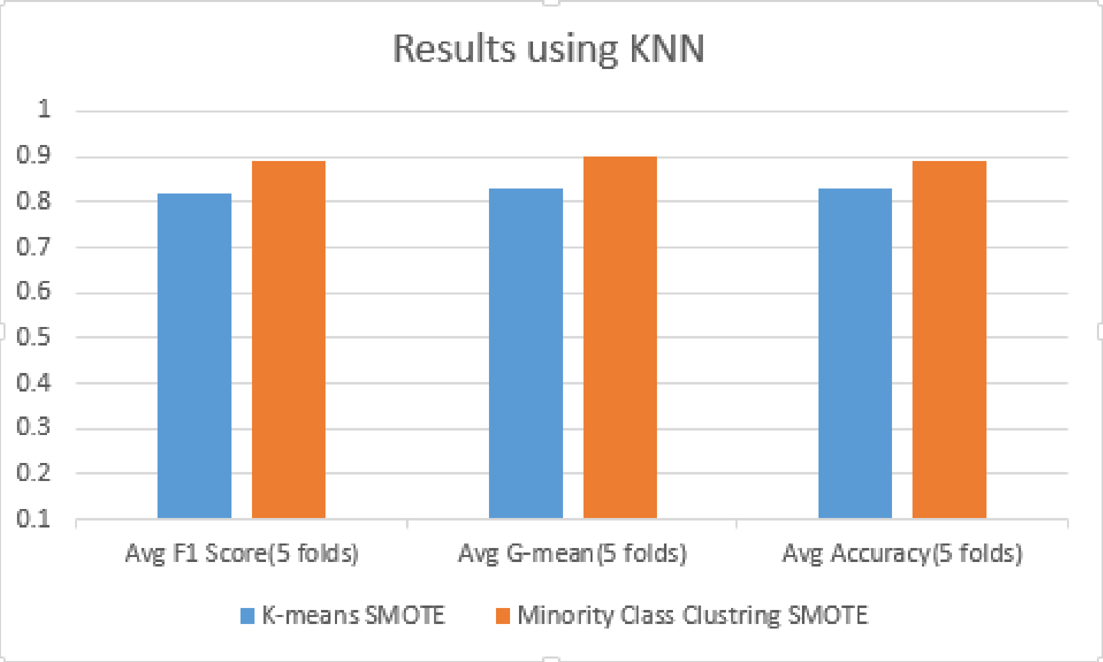

FIGURE 4.1: Average Results using KNN Classifier

4.1.2 Results using LR Classifier

If we talk about the results using LR then we observed that biggest gain that we achieved is 0.2, 0.19 and 0.2 for F1-measure, G-mean and accuracy respectively. But for 3 datasets it shows no improvements. The average result that was shown is same i.e. 90% for all measures. Figure 4.2 shows the comparison of average results using F1-measure, G-mean and accuracy of K-mean SMOTE and Minority Class clustering SMOTE technique.

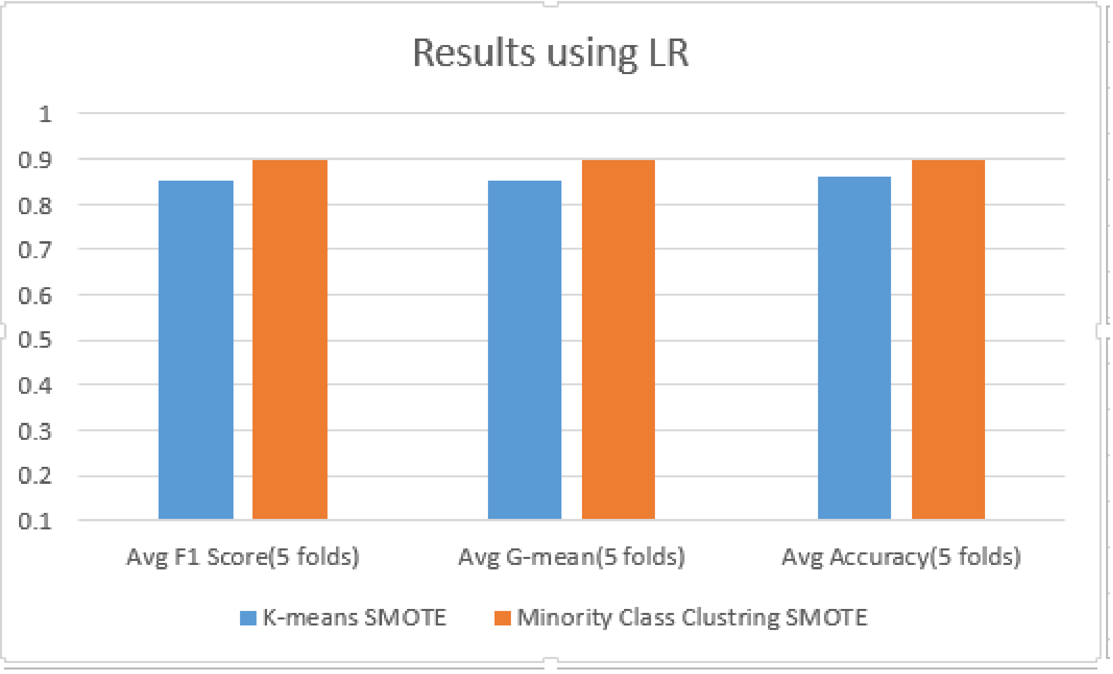

FIGURE 4.2: Average Results using LR Classifier

Moreover, it also observed for both strategies that LR showed better results over KNN.

4.2 Results of Oversee SMOTE

Basically we proposed this approach to observe that what will be the results if we use a classification algorithm instead of making clusters in resampling process.

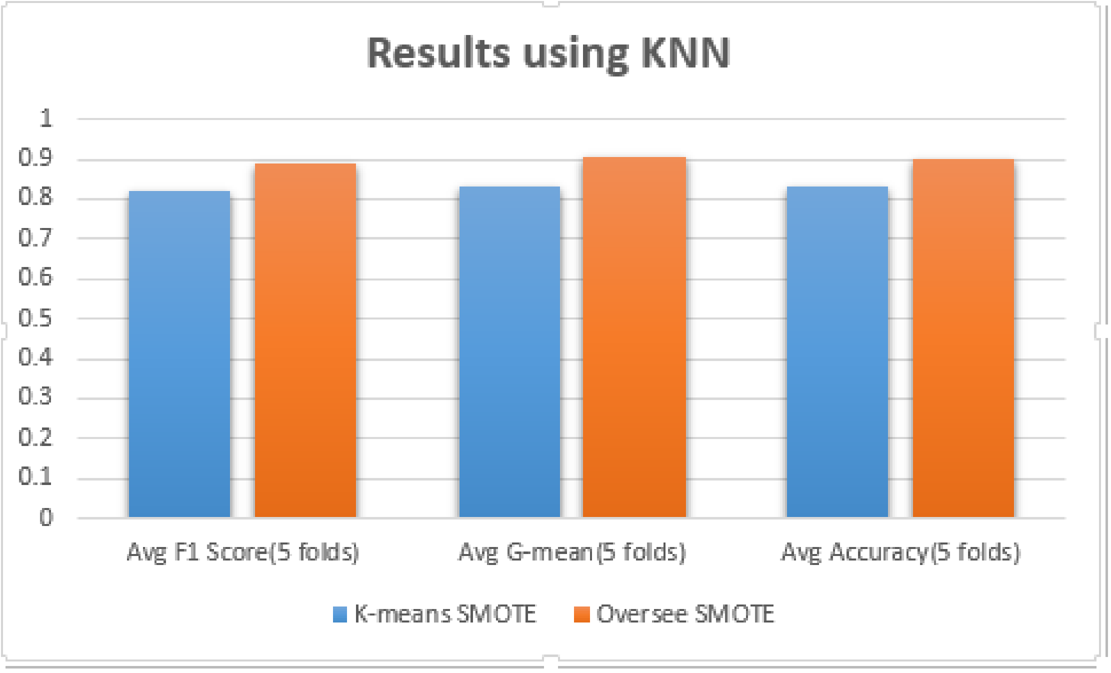

FIGURE 4.3: Average Results using KNN Classifier

After getting the results we come to know that overall this approach is also outperformed then base paper approach as well as our other proposed approach too. However in case of classifiers here also LR performs better than KNN.

4.2.1 Results using KNN Classifier

Using KNN classifier Oversee SMOTE out performed for all datasets but the results for Iris dataset is same as we discussed in MCC SMOTE approach's results. However, for two dataset it outperformed the K-means SMOTE and achieved 100% results in term of all measures. Figure 4.3 shows the comparison of average results using F1-measure, G-mean and accuracy of K-mean SMOTE and Minority Class clustering SMOTE technique.

4.2.2 Results using LR Classifier

In case of LR this approach also outperformed for all datasets and achieved 90% F1-measure, 90% G-mean and 91% accuracy. This outperformance shows that

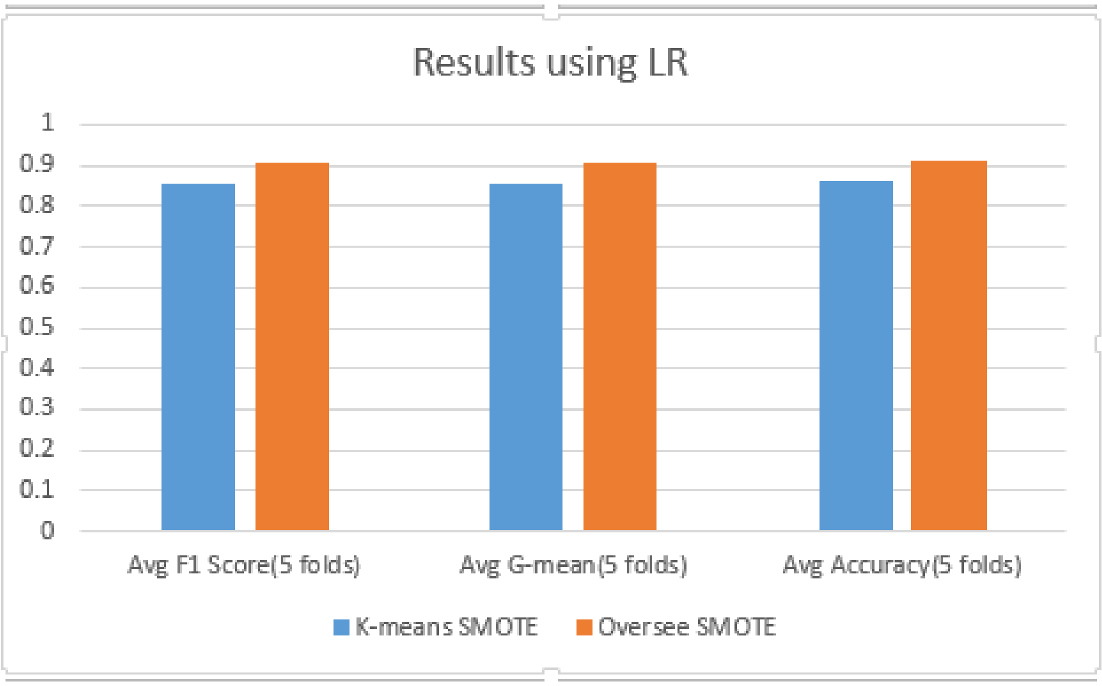

FIGURE 4.4: Average Results using LR Classifier

the generated instance are less noisy than the generated instance using clustering technique. However, the one thing that is observed during the run of Oversee SMOTE the algorithm is cost sensitive in term of time because of iterative step of probability calculation.However, the one thing that is observed during the run of Oversee SMOTE the algorithm is cost sensitive in term of time because of iterative step of probability calculation. However, the one thing that is observed during the run of Oversee SMOTE the algorithm is cost sensitive in term of time because of iterative step of probability calculation. Figure 4.4 shows the comparison of average results using F1-measure, G-mean and accuracy of K-mean SMOTE and Minority Class clustering SMOTE technique.

4.3 Results of Oversee SMOTE and MCC SMOTE

We also compared Oversee SMOTE results with our other proposed approach MCC SMOTE and observed that Oversee SMOTE outperformed then MCC SMOTE in terms of accuracy by 0.01. Figure 4.5 and 4.6 shows the graphical representation of results in bar graph.

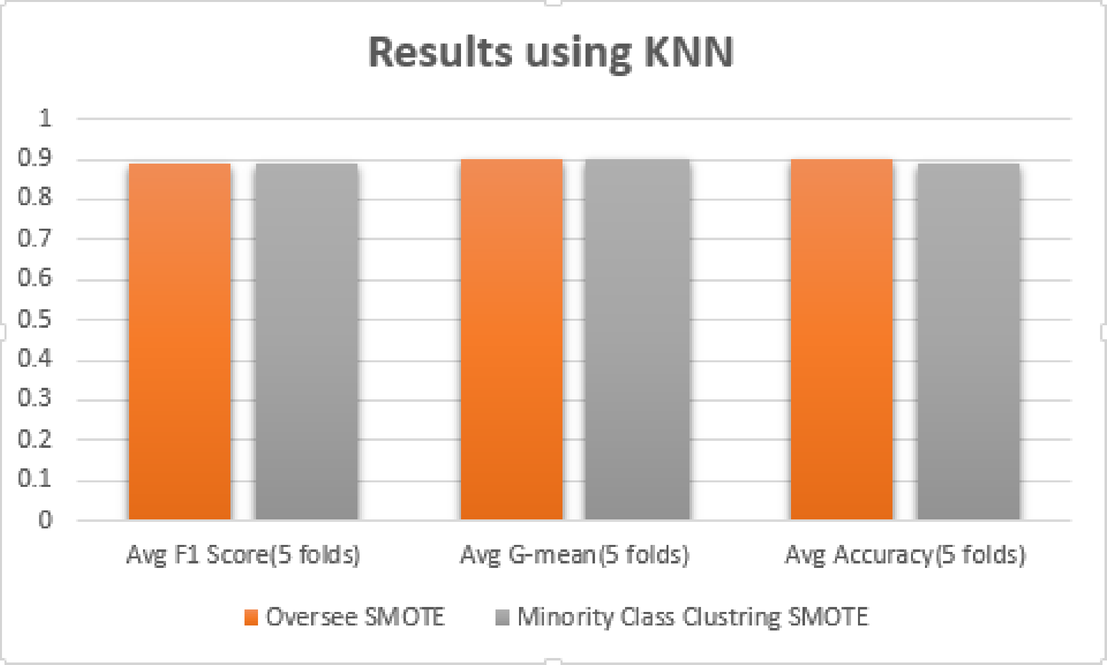

FIGURE 4.5: Comparison's Results of MCC SMOTE and Oversee SMOTE

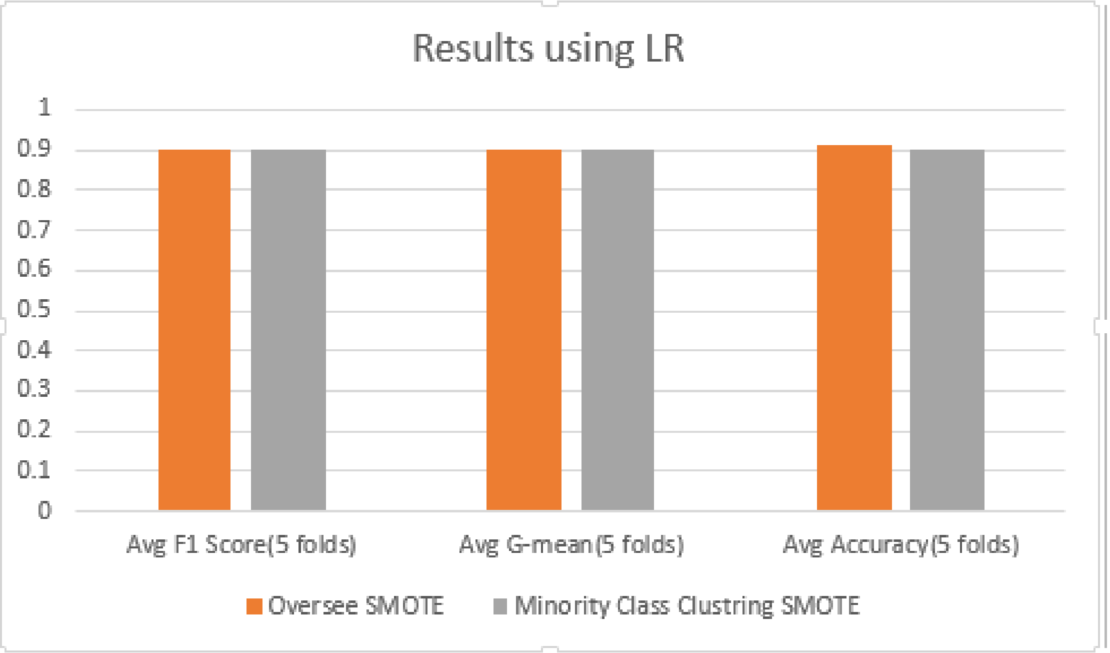

FIGURE 4.6: Comparison's Results of MCC SMOTE and Oversee SMOTE

4.4 Results in Terms of Area Under the Curve

Additionally we also compare the results in terms of Area Under the curve or AUC. When we compare our proposed approaches with base line approach in terms of AUC both approaches MCC SMOTE and Oversee SMOTE gave outperformed

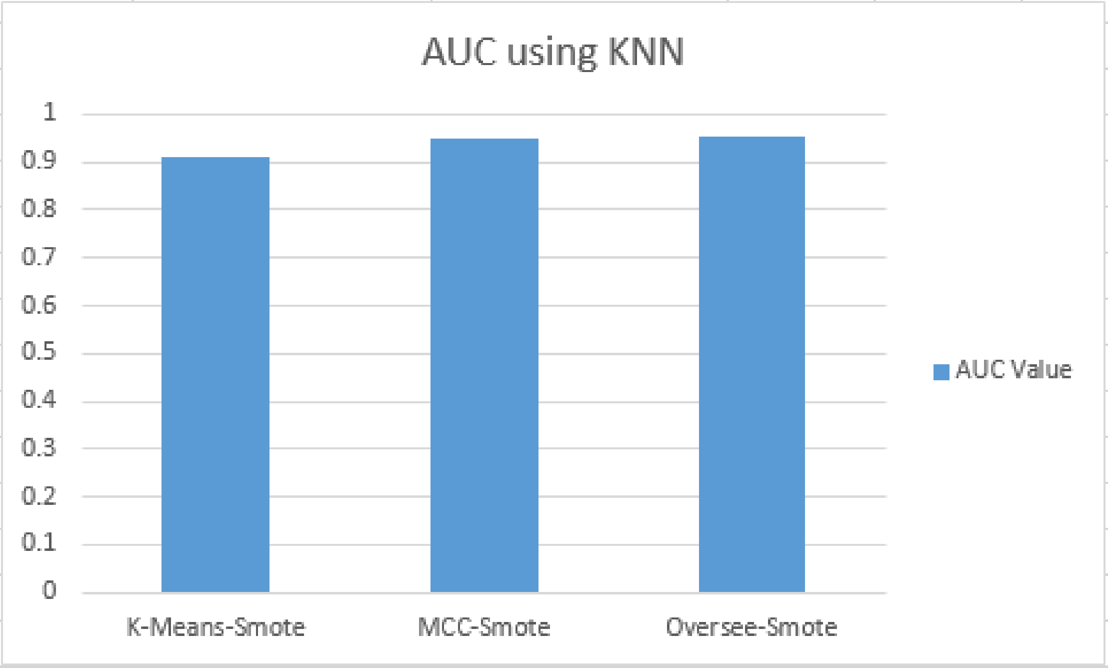

FIGURE 4.7: Average Results of AUC

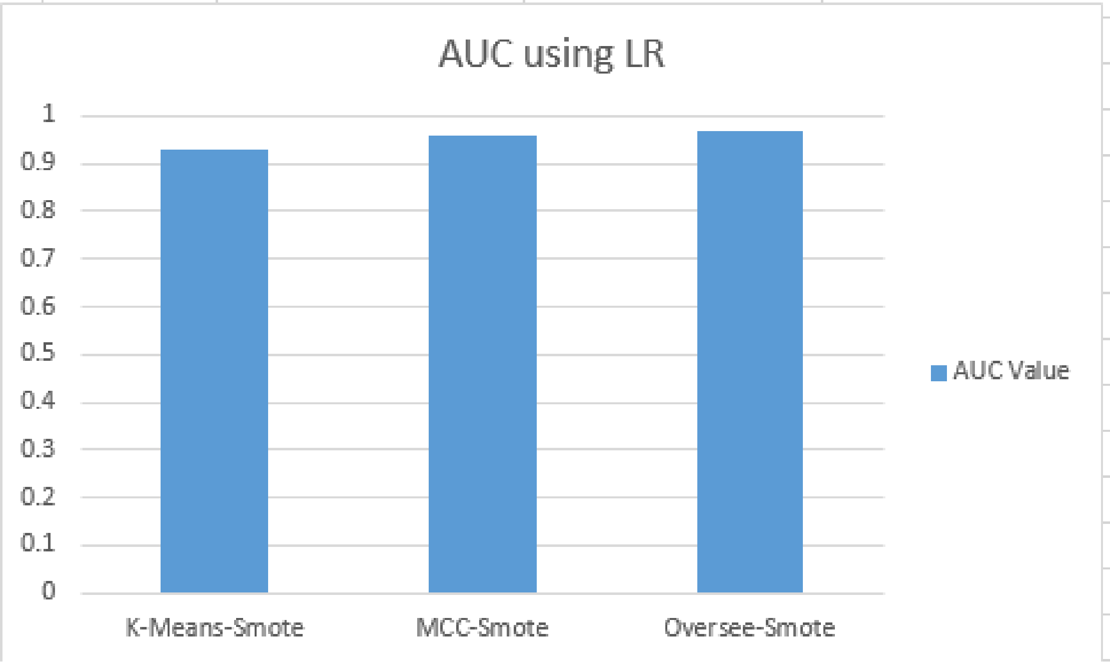

FIGURE 4.8: Average Results of AUC

results using KNN and LR classifiers. Comparative results are shown in table 4.4. Figure 4.7 and 4.8 represents the results of average AUC of all datasets. As you can see our both approaches outperform k-Means Smote. it means our both approaches better predict positive class as postive and negative class as negative.

Table 4.4: AUC values of K Means Smote, MCC Smote and Oversee Smote

Dataset	**IR**	**Classifier**	**K-means SMOTE**	**MCC SMOTE**	**Oversee SMOTE**
Breast Tissue	(70:36)	KNN	0.73	**0.77**	**0.78**
		LR	0.96	**0.99**	0.95
Cleveland	(13:160)	KNN	0.83	**0.93**	0.81
		LR	0.95	**0.97**	**0.99**
Dermatology	(20:338)	KNN	0.71	**0.94**	**1**
		LR	1	1	1
Ecoli	(52:284)	KNN	0.99	**1**	0.96
		LR	0.99	**1**	0.92
Eucalyptus	(98:544)	KNN	0.76	**0.88**	**0.83**
		LR	0.81	**0.89**	**0.87**
Glass	(70:144)	KNN	0.99	**1**	**1**
		LR	0.99	**1**	**1**
Haberman	(81:225)	KNN	0.53	**0.63**	**0.8**
		LR	0.43	**0.61**	**0.85**
Heart	(120:150)	KNN	0.61	**0.64**	**0.75**
		LR	0.79	**0.8**	**0.84**
Iris	(50:100)	KNN	0.98	**1**	**1**
		LR	1	1	1
Led	(37:406)	KNN	0.99	**1**	**1**
		LR	0.99	**1**	**1**
Libras	(24:336)	KNN	0.99	**1**	**1**
		LR	0.99	**1**	**1**
Liver	(145:200)	KNN	0.71	**0.74**	**0.87**
		LR	0.65	**0.68**	**0.9**
New Thyroid 1	(35:180)	KNN	0.92	**0.93**	0.92
		LR	0.97	0.91	0.95
New Thyroid 2	(35:180)	KNN	0.99	0.99	0.99
		LR	1	1	1
Page Blocks 1	(28:444)	KNN	0.96	**0.98**	**0.98**
		LR	0.81	**1**	**1**
Pima	(268:500)	KNN	0.75	**0.78**	**0.88**
		LR	0.81	**0.83**	**0.9**
Vehicle	(199:647)	KNN	0.9	**0.94**	**0.94**
		LR	0.92	**0.98**	**0.98**
Vowel	(90:898)	KNN	0.58	**0.89**	**0.93**
		LR	0.96	**0.98**	**0.94**
Wine	(71:107)	KNN	0.99	**0.99**	**0.99**
		LR	0.93	**0.96**	**0.96**
Yeast 1	(429:1055)	KNN	0.82	0.82	**0.9**
		LR	0.66	**0.69**	**0.92**

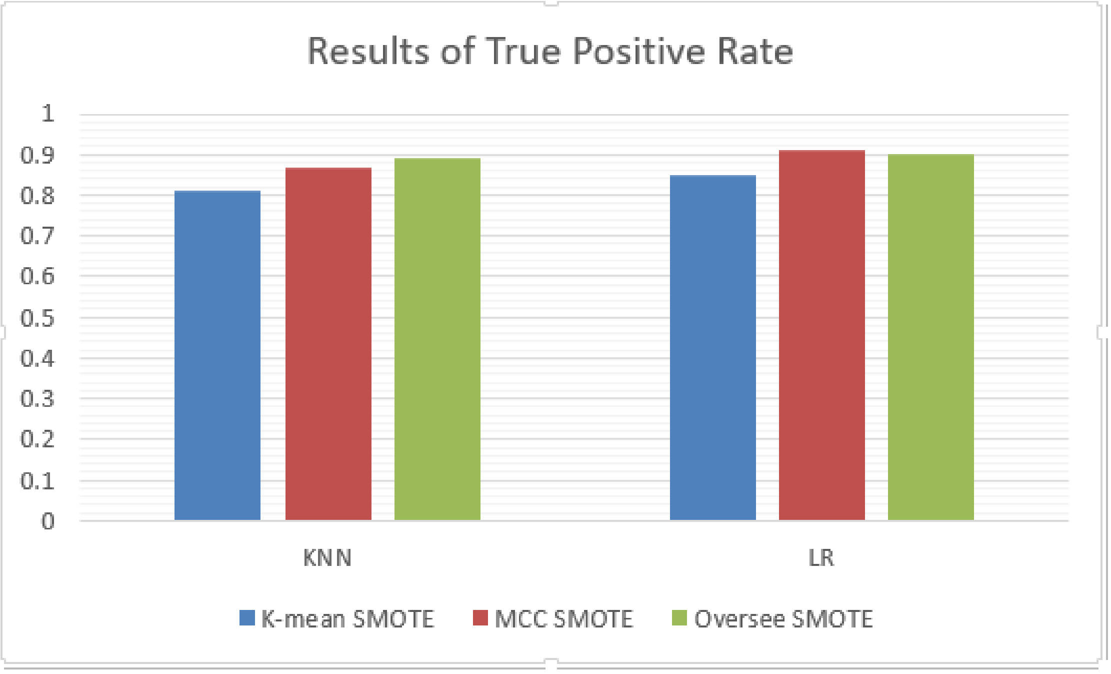

FIGURE 4.9: Average Results of True Positive Rate

4.5 Results in Terms of True positive Rate

True positive rate (TPR) or recall is one of the best measure to assess the results in case of imbalance dataset. The TPR ratio measures how many appropriate samples are picked, which indicates how well all the positive samples involved in our dataset can be predicted by our model. Our results show that both proposed approaches outperformed the base paper approach. Table 4.5 shows the individual TPR results on some of our datasets.

4.6 Analysis Results

Besides the above comparisons Figure 4.10 to 4.17 shows analysis of some datasets that how K- means minority class clustering SMOTE and Oversee SMOTE have achieved good results. Here 0 represents majority class with red colored dots, 1 represents minority class with blue colored symbol and 2 represents new generated instances with green colored symbol. Figure 4.9 and 4.10 represents analysis of

Table 4.5: Results of True Positive Rate with k-means SMOTE, MCC SMOTE and Oversee SMOTE

Dataset	**IR**	**Classifier**	**K-means SMOTE**	**MCC SMOTE**	**Oversee SMOTE**
Breast Tissue	(70:36)	KNN	0.64	**0.74**	**0.76**
		LR	0.97	**0.98**	0.78
Cleveland	(13:160)	KNN	0.94	0.92	0.8
		LR	0.98	0.96	0.83
Dermatology	(20:338)	KNN	0.96	0.96	**1**
		LR	0.99	**1**	**1**
Ecoli	(52:284)	KNN	1	**1**	0.94
		LR	1	**1**	0.94
Eucalyptus	(98:544)	KNN	0.75	**0.84**	**0.87**
		LR	0.84	**0.91**	**0.85**
Glass	(70:144)	KNN	0.95	**1**	0.85
		LR	0.96	**1**	0.84
Haberman	(81:225)	KNN	0.66	**0.72**	**0.83**
		LR	0.56	**0.6**	**0.86**
Heart	(120:150)	KNN	0.47	**0.7**	**0.86**
		LR	0.8	**0.86**	**0.84**
Iris	(50:100)	KNN	1	1	1
		LR	1	1	1
Led	(37:406)	KNN	1	**1**	0.95
		LR	1	**1**	0.95
Libras	(24:336)	KNN	0.99	**1**	0.95
		LR	1	1	0.95
Liver	(145:200)	KNN	0.62	**0.69**	**0.84**
		LR	0.59	**0.68**	**0.83**
New Thyroid 1	(35:180)	KNN	0.85	**0.92**	**0.92**
		LR	0.88	**0.95**	**0.91**
New Thyroid 2	(35:180)	KNN	0.84	**0.9**	**1**
		LR	0.86	**0.93**	**1**
Page Blocks 1	(28:444)	KNN	0.9	**0.94**	**0.97**
		LR	0.9	**0.96**	**0.98**
Pima	(268:500)	KNN	0.91	**0.96**	0.88
		LR	0.92	**0.98**	0.92
Vehicle	(199:647)	KNN	0.65	**0.74**	**0.93**
		LR	0.94	**1**	**0.96**
Vowel	(90:898)	KNN	0.9	**0.97**	**0.95**
		LR	0.91	0.9	**0.95**
Wine	(71:107)	KNN	0.7	**0.88**	**0.93**
		LR	0.76	**0.9**	**0.95**
Yeast 1	(429:1055)	KNN	0.75	**0.8**	**0.8**
		LR	0.67	**1**	**0.94**

'Haberman' and 'Heart' dataset respectively using K- means minority class clustering SMOTE. It clearly can be seen that instance generated through our proposed approach are less noisy than the instances generated through baseline approach.

4.6.1 Analysis of K-means Minority Class Clustering SMOTE and K-means SMOTE

Figure 4.10 and 4.11 represents analysis of 'Haberman' dataset using K-means SMOTE [5] and K- means minority class clustering SMOTE respectively. It clearly can be seen that instance generated through our proposed approach are less noisy than the instances generated through baseline approach. In case of k-means SMOTE the randomly selected instances by the SMOTE are mostly lay in majority class therefor new generated instance also lay in majority class having class label of minority class. But if we observe the Figure 4.11 we can see that there is no such new instance that is created by selecting majority class sample. Another analysis is performed on 'Cleveland' dataset using K-mean SMOTE and Minority

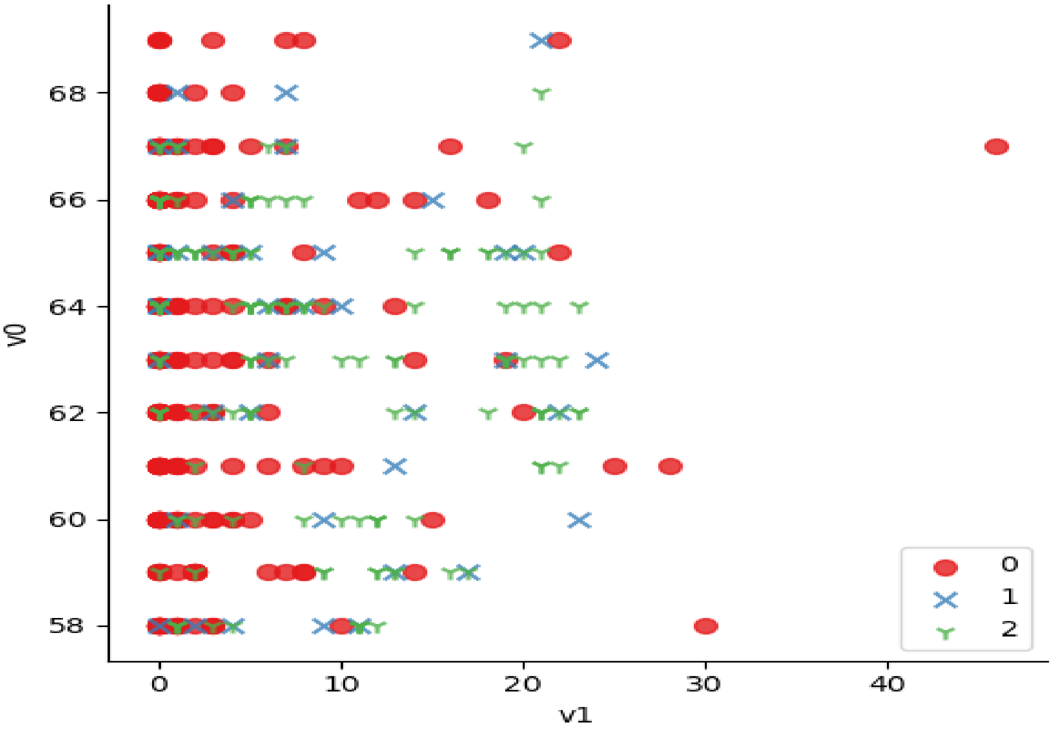

FIGURE 4.10: Instances Generated Through K-means SMOTE

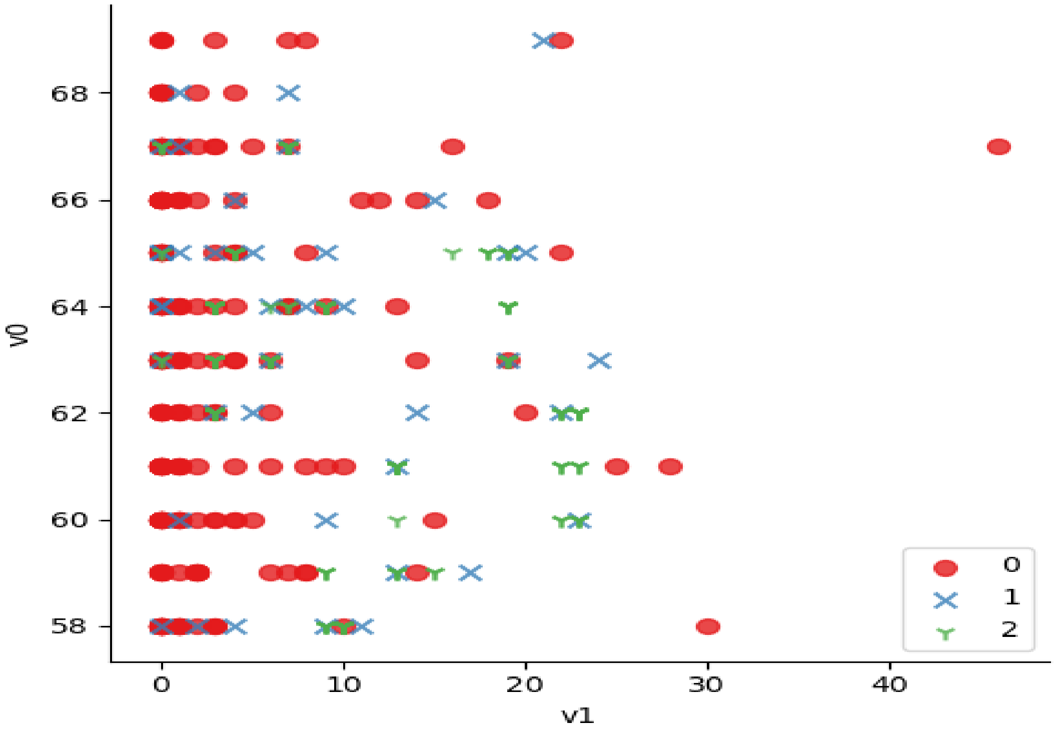

FIGURE 4.11: Instances Generated through MCC SMOTE

Class Clustering SMOTE respectively. In Figure 4.12 it can be clearly seen that instance generated through k-means SMOTE produced problem of noisy instances. There exist many new generated instances that are not clearly lay between two minority instances on a straight line. It means that when they make whole feature space clusters then there may exist such clusters in which valuable number of majority instances present. So when SMOTE chooses a nearest neighbor by using Euclidean Distance within the cluster to create new samples then it may selects a majority instance as its nearest neighbor. Such type of clusters produces noisy instances. It means that when they make whole feature space clusters then there may exist such clusters in which valuable number of majority instances present. So when SMOTE chooses a nearest neighbor by using Euclidean Distance within the cluster to create new samples.

Our proposed approach named MCC SMOTE tackle the above discussed problem shown in Figure 4.12 to a great extent. Figure 4.13 clearly shows that new generated samples are exactly created between 2 minority class instances and are less noisy than the instances generated through baseline approach.

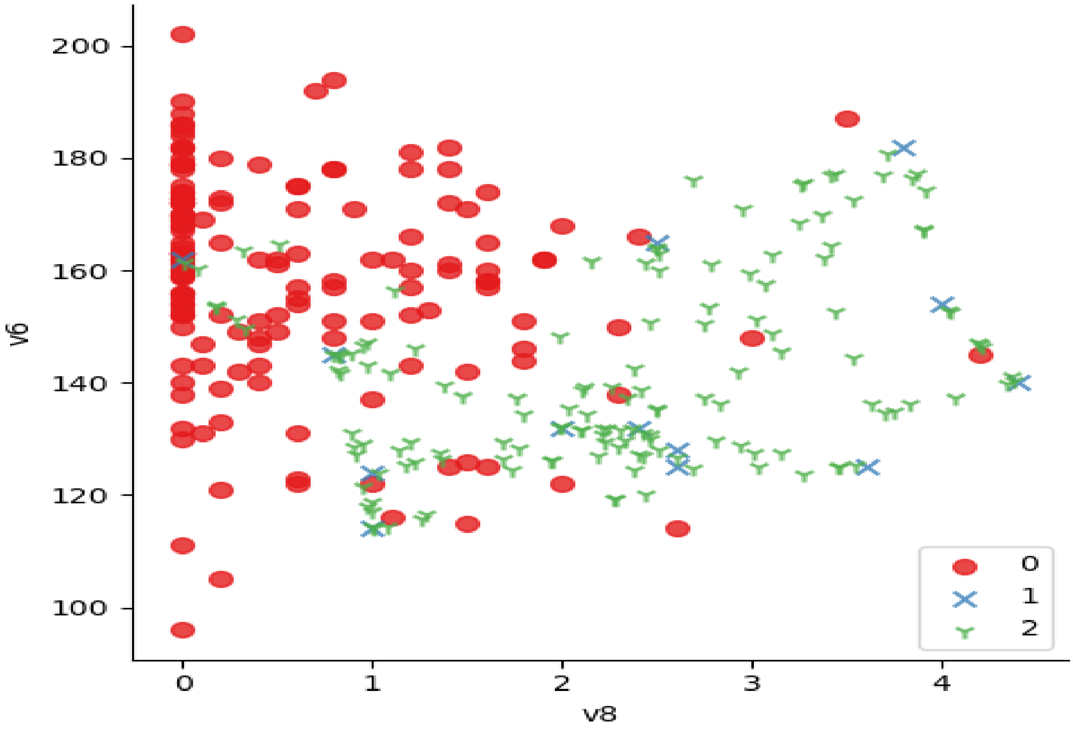

Figure 4.12: Instances Generated through K-means SMOTE

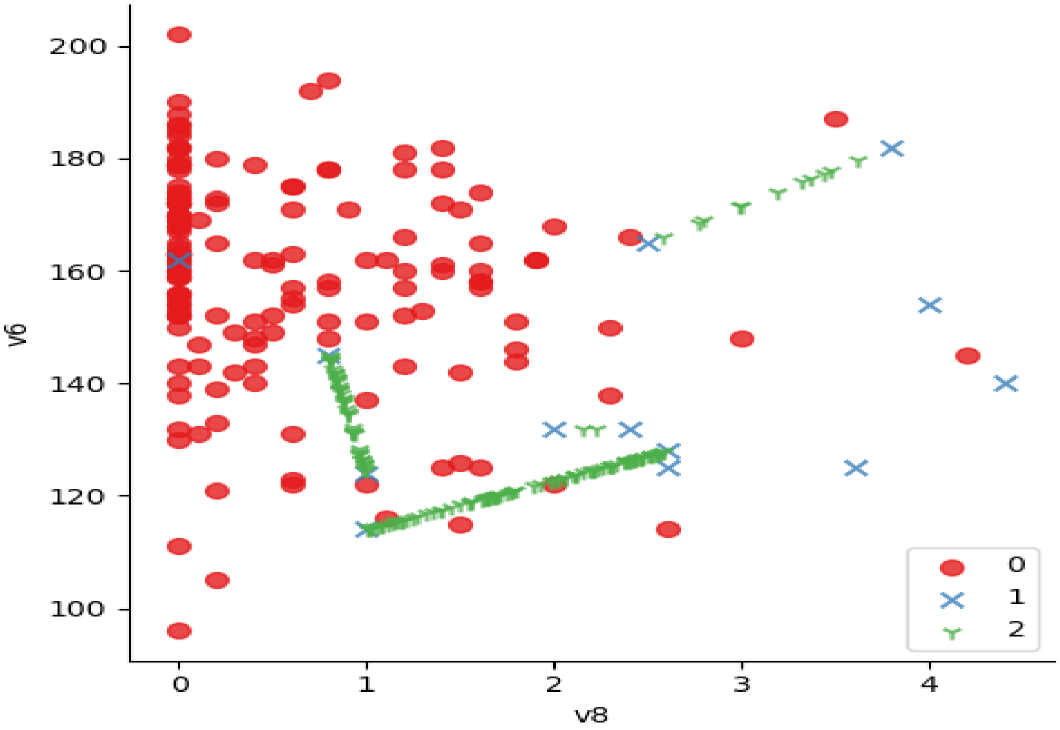

Figure 4.13: Instances Generated through MCC SMOTE

4.6.2 Analysis of Oversee SMOTE and K-means SMOTE

This section represents the performed analysis on dataset named 'Heart'. Figure 4.14 represents the data generated using k-means SMOTE and it can be observed that many new instances have been generated among the majority class samples. When we perform the same analysis on the same dataset but oversampled it by Oversee SMOTE it showed that our proposed approach genuinely outperformed the baseline technique. There is less such new generated instances that are created by selecting majority class instance as the nearest neighbor of randomly selected minority class sample by SMOTE. Another analysis is performed on dataset named 'Led'. Figure 4.16 represents the dataset view after oversampling using k-means SMOTE and it can be observed that many new instances have been generated among the majority class samples. The reason is the same as we discussed above that when clusters are made up with whole feature space there exist such clusters that have both majority and minority observations which put impact on new created samples.

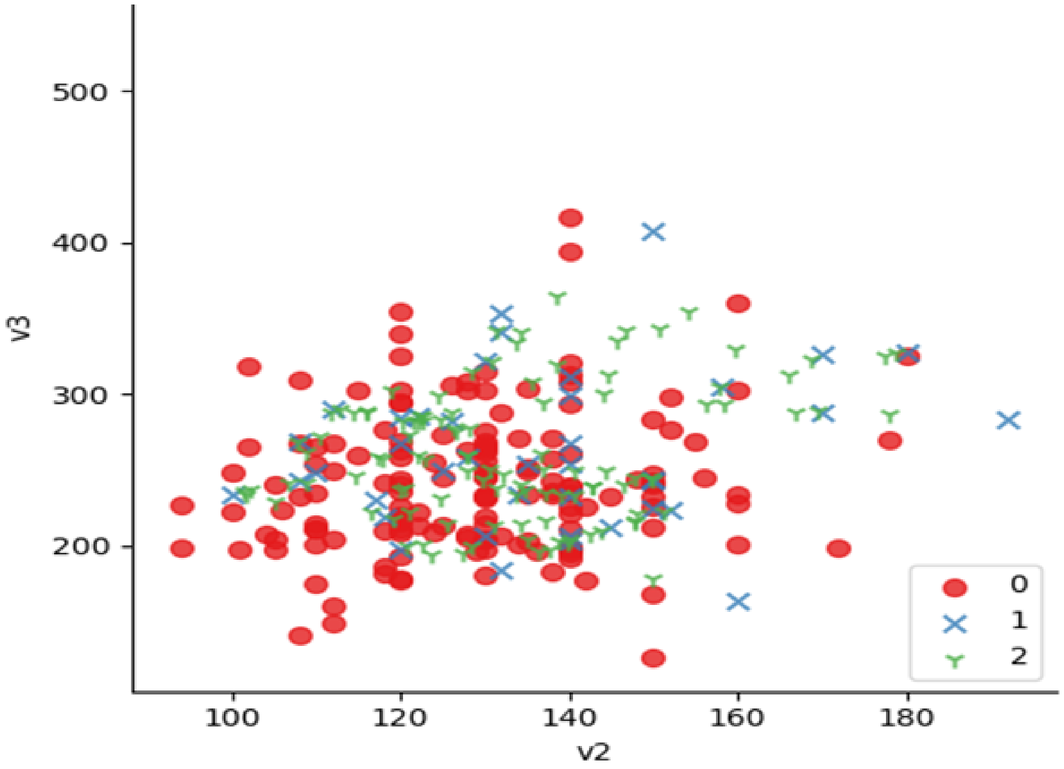

FIGURE 4.14: Instances Generated through K-means SMOTE

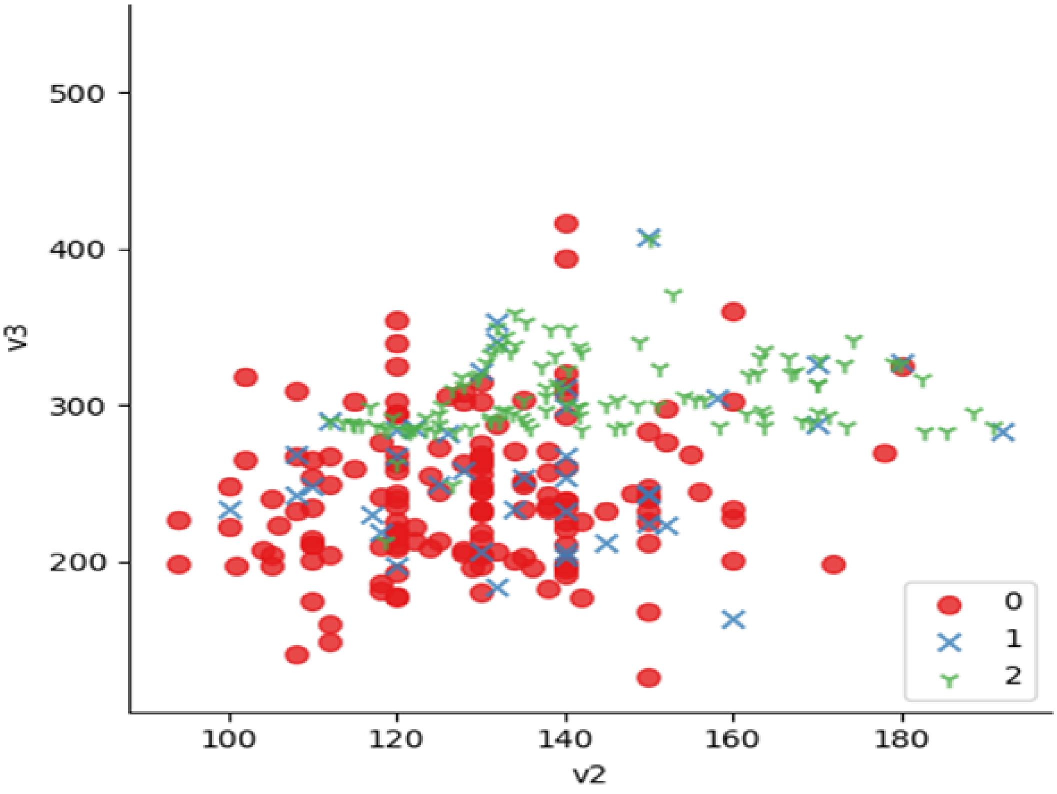

Figure 4.15: Instances Generated through K-means SMOTE

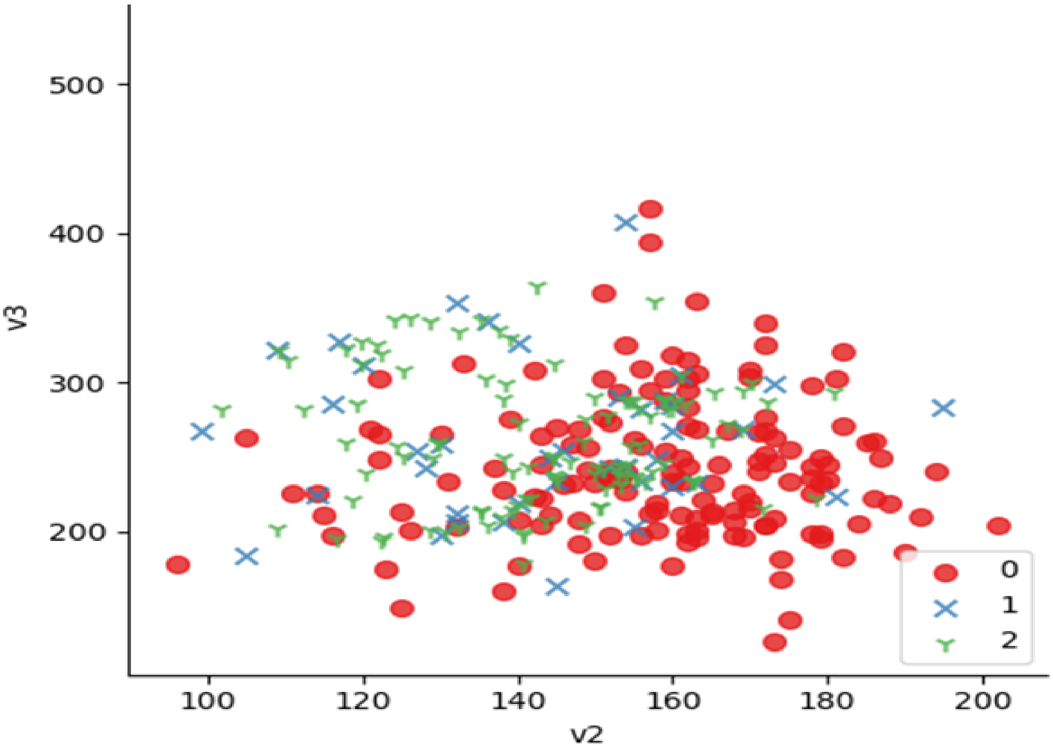

Figure 4.16: Instances Generated through K-means SMOTE

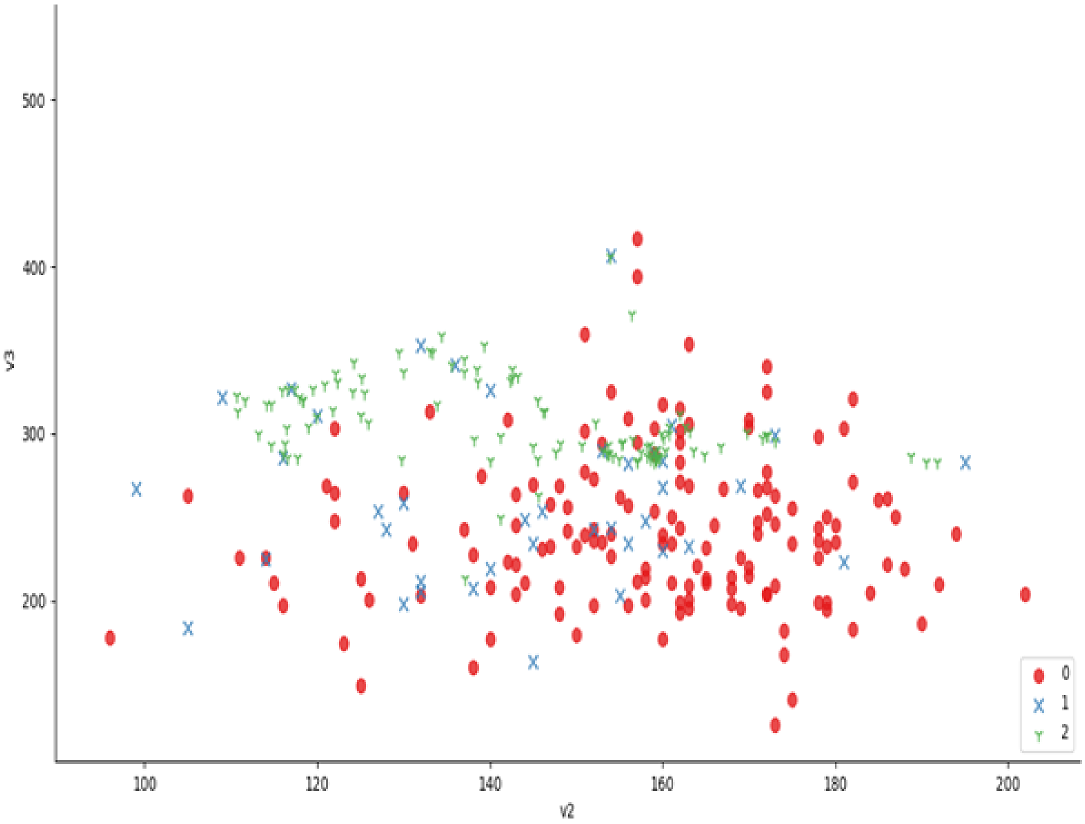

FIGURE 4.17: Instances Generated through Oversee SMOTE

Now if observe the analysis result of Oversee SMOTE 4.17 on 'Led' dataset we came to know that our proposed approach covers the above discussed problem which was shown in Figure 4.16. Because Oversee SMOTE firstly trained the model on equal amount of instances of both positive and negative class so when it gives the probability for a new generated artificial instance it will not biased to only one of the class. The generated sample is fairly categorized either it belongs to the minority class or majority class that is the main reason to reduces the number of false positive observations due to which noisy instances are reduced.

Chapter 5

Conclusion and Future Work

This section will provide the conclusion of our research work and limitations for the future work.

5.1 Conclusion

In many classifiers, unbalanced data presents a challenging task. In this situation to balance the data, resampling of the training data is the best direction to tackle this problem regardless of the classification algorithm. Besides this if the balanced data have duplicate positive instances than it stimulate the over-fitting problem and it reduce the performance of classifiers trained model for the obscured information. Another problem that researchers faced in field of imbalance data is that level of noise is increased after oversampling due to the high effect of majority class on new created objects. Additionally, most of the existing sampling techniques do not perform better in case of high imbalance ratio. To effectually assist the classification algorithm's performance the synthetic data should not contain noisy samples by avoiding to be generated in majority areas. Moreover, the duplication of data should be avoided by trying to select the different minority instances to generate the synthetic data. Our proposed methodologies cover the above discussed limitations. Our contribution indicated the following points.

1. Clustering of only minority class in Minority Class Clustering SMOTE makes the area safe for the generation of new instances reduce the noisy samples by achieving 90% accuracy.

2. Our other proposed technique Oversee SMOTE covers the above both limitation of noisy samples by using the 5 times iteratively calculated probability of a new generated instance.

3. Analysis shows that it also reduce the duplication to some extent because the new generated instance is included in minority class very carefully. This methodology achieved over all 91% accuracy.

Finally, this work find that MCC SMOTE and Oversee SMOTE is outperformed to the K-means clustering SMOTE.

5.2 Future Work:

Although our proposed approaches outperformed the baseline approach by achieving the quality results. However, we have identified some of potential directions for future research in this area which are described below:

1. The one thing that is observed during the run of Oversee SMOTE the algorithm is cost sensitive in term of time because of iterative step of probability calculation. So in future we can focus to make the change at algorithm level to reduce the run time.

2. The second thing is to focus on finding the most appropriate range of threshold value for minority class so that more optimal synthetic data could be generated.

3. The third thing is to find the optimal value of K for K Means method.

Bibliography

[1] C. O'Neil and R. Schutt, "Doing data science: Straight talk from the frontline," *O'Reilly Media*, vol. 5, no. 17, pp. 1–378, 2013.

[2] J. Han, J. Pei, and M. Kamber, "Data mining: concepts and techniques," *Elsevier*, vol. 96, no. 209, pp. 1–585, 2011.

[3] W.-C. Lin, C.-F. Tsai, Y.-H. Hu, and J. S. Jhang, "Clustering-based undersampling in class-imbalanced data," *Information Sciences*, vol. 409, no. 25, pp. 17–26, 2017.

[4] G. Douzas, F. Bacao, and F. Last, "Improving imbalanced learning through a heuristic oversampling method based on k-means and smote," *Information Sciences*, vol. 456, no. 3, pp. 1–20, 2018.

[5] Q. Kang, X. Chen, S. Li, and M. Zhou, "A noise-filtered under-sampling scheme for imbalanced classification," *IEEE transactions on cybernetics*, vol. 47, no. 12, pp. 4263–4274, 2016.

[6] F. R. Torres, J. A. Carrasco-Ochoa, and J. F. Martínez-Trinidad, "Smote-d a deterministic version of smote," *In Mexican Conference on Pattern Recognition*, vol. 20, no. 3, pp. 177–188, 2016.

[7] S. Krishnan, J. Wang, E. Wu, M. J. Franklin, and K. Goldberg, "Activeclean: Interactive data cleaning for statistical modeling," *Proceedings of the VLDB Endowment,*, vol. 9, no. 12, pp. 948–959, 2016.

[8] J. A. Sáez, J. Luengo, J. Stefanowski, and F. Herrera, "Smote–ipf: Addressing the noisy and borderline examples problem in imbalanced classification by a

re-sampling method with filtering," *Information Sciences*, vol. 291, no. 15, pp. 184–203, 2015.

[9] P. Kaur and A. Gosain, "Comparing the behavior of oversampling and undersampling approach of class imbalance learning by combining class imbalance problem with noise," *ICT Based Innovations*, vol. 152, no. 6, pp. 23–30, 2018.

[10] J. Yun, J. Ha, and J.-S. Lee, "Automatic determination of neighborhood size in smote," *Proceedings of the 10th International Conference on Ubiquitous Information Management and Communication*, vol. 7, no. 3, pp. 1–8, 2016.

[11] R. Pruengkarn, K. W. Wong, and C. C. Fung, "Imbalanced data classification using complementary fuzzy support vector machine techniques and smote," *2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, vol. 18, no. 3, pp. 978–983, 2017.

[12] J. Lee, N.-r. Kim, and J.-H. Lee, "An over-sampling technique with rejection for imbalanced class learning," *Proceedings of the 9th International Conference on Ubiquitous Information Management and Communication*, vol. 15, no. 11, pp. 1–6, 2015.

[13] R. Pruengkarn, K. W. Wong, and C. C. Fung, "Data cleaning using complementary fuzzy support vector machine technique," *International Conference on Neural Information Processing*, vol. 4, no. 8, pp. 160–167, 2016.

[14] D. Elreedy and A. F. Atiya, "A comprehensive analysis of synthetic minority oversampling technique (smote) for handling class imbalance," *Information Sciences*, vol. 505, no. 17, pp. 575–603, 2019.

[15] S. Piri, D. Delen, and T. Liu, "A synthetic informative minority over-sampling (simo) algorithm leveraging support vector machine to enhance learning from imbalanced datasets," *Decision Support Systems*, vol. 106, no. 5, pp. 15–29, 2018.

[16] W. A. Rivera and P. Xanthopoulos, “A priori synthetic over-sampling methods for increasing classification sensitivity in imbalanced data sets,” *Expert Systems with Applications*, vol. 66, no. 15, pp. 124–135, 2016.

[17] X. Zhang, Y. Li, R. Kotagiri, L. Wu, Z. Tari, and M. Cheriet, “Krnn: k rare-class nearest neighbour classification,” *Pattern Recognition*, vol. 62, no. 9, pp. 33–44, 2017.

[18] J. Hu, X. He, D.-J. Yu, X.-B. Yang, J.-Y. Yang, and H.-B. Shen, “A new supervised over-sampling algorithm with application to protein-nucleotide binding residue prediction.” *PloS one*, vol. 9, no. 9, pp. 33–47, 2014.

[19] S. S. Mullick, S. Datta, and S. Das, “Adaptive learning-based k-nearest neighbor classifiers with resilience to class imbalance,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 11, pp. 5713–5725, 2018.

[20] Z. Xie, L. Jiang, T. Ye, and X. Li, “A synthetic minority oversampling method based on local densities in low-dimensional space for imbalanced learning,” *International Conference on Database Systems for Advanced Applications*, vol. 4, no. 3, pp. 3–18, 2015.

[21] P. Lim, C. K. Goh, and K. C. Tan, “Evolutionary cluster-based synthetic oversampling ensemble (eco-ensemble) for imbalance learning,” *IEEE transactions on cybernetics*, vol. 47, no. 9, pp. 2850–2861, 2016.

[22] W. Siriseriwan and K. Sinapiromsaran, “Adaptive neighbor synthetic minority oversampling technique under 1nn outcast handling,” *Songklanakarin J. Sci. Technol*, vol. 39, no. 4, pp. 565–576, 2017.

[23] S. Hu, Y. Liang, L. Ma, and Y. He, “Msmote: Improving classification performance when training data is imbalanced,” *2009 second international workshop on computer science and engineering*, vol. 2, no. 3, pp. 13–17, 2009.

[24] W. A. Rivera, “Noise reduction a priori synthetic over-sampling for class imbalanced data sets,” *Information Sciences*, vol. 408, no. 9, pp. 146–161, 2017.

[25] A. Gosain and S. Sardana, "Farthest smote: a modified smote approach," *Computational Intelligence in Data Mining*, vol. 3, no. 9, pp. 309–320, 2019.

[26] X. Zhang and Y. Li, "A positive-biased nearest neighbour algorithm for imbalanced classification," *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, vol. 25, no. 123, pp. 293–304, 2013.

[27] S. Sharma, C. Bellinger, B. Krawczyk, O. Zaiane, and N. Japkowicz, "Synthetic oversampling with the majority class: A new perspective on handling extreme imbalance," *2018 IEEE International Conference on Data Mining (ICDM)*, vol. 4, no. 7, pp. 447–456, 2018.

[28] I. Nekooeimehr and S. K. Lai-Yuen, "Adaptive semi-unsupervised weighted oversampling (a-suwo) for imbalanced datasets," *Expert Systems with Applications*, vol. 46, no. 5, pp. 405–416, 2016.

[29] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "Smote: synthetic minority over-sampling technique," *Journal of artificial intelligence research*, vol. 16, no. 6, pp. 321–357, 2002.

[30] A. Verma, "Evaluation of classification algorithms with solutions to class imbalance problem on bank marketing dataset using weka," *International Research Journal of Engineering and Technology*, vol. 5, no. 13, pp. 54–60, 2019.

[31] A. Pertiwi, N. Bachtiar, R. Kusumaningrum, I. Waspada, and A. Wibowo, "Comparison of performance of k-nearest neighbor algorithm using smote and k-nearest neighbor algorithm without smote in diagnosis of diabetes disease in balanced data," *Journal of Physics: Conference Series*, vol. 1524, no. 1, pp. 12–48, 2019.