Data Level Approach for Multiclass Imbalance Financial Data

NURSEL SELVER RUZGAR[1], CLARE CHUA[2]
[1, 2] Ted Rogers School of Management
Ryerson University
350 Victoria Street, Toronto, ON M5B 2K3
CANADA
nruzgar@ryerson.ca

Abstract: - In the real world, the class imbalance problem is a common issue in which the classifier gives more importance to the majority class and less importance to the minority class. This paper examines the three resampling approaches (undersampling, oversampling and hybrid) with different points of view to solve the imbalance problems. Here, we aimed to show how the multiclass imbalance data can be classified with different resampling approaches and how the performance metrics of classification algorithms are influenced by the proposed resampling methods. For this purpose, one new undersampling and three hybrid resampling approaches were proposed. To test the impact of the proposed approaches on the performances of nine selected algorithms, financial data sets from two Canadian banks spanning 37 years were used and analysed by WEKA software. The results provide a clear picture on the overall impact of class imbalance on the classification data set and indicate that the proposed resampling methods with different definitions can also be used in class imbalance problems.

Key-Words: - Imbalance classification, Resampling, Oversampling, Undersampling, Hybrid, WEKA: Received: April 12, 2020. Revised: September 6, 2020. Accepted: October 9, 2020. Published: October 27, 2020.

1 Introduction

The class imbalance problems have a crucial role in different fields and have been taken interest of researchers especially in last two decades. Imbalance data set is a set that is classified into different ratio of the classes, majority, and minority classes. The class with large number of instances is called majority class and the other is called minority class [1,2]. In real world, imbalance data exist in many applications, such as detection of fraudulent phone calls [2,3], intrusion detection [4, 5], financial fraud detection [6], medical diagnosis [7], pediatric brain tumors from magnetic spectroscopy [8], text classification [9], credit assessment [10], fault diagnosis [11], and so on.

Imbalance in classes affects the classification performance of a classifier [12, 13]. Most of time, the minority classes are more important than those majority ones. Since the traditional methods are used under the assumption of equal class distributions and equal misclassification cost for each class, it is often difficult to obtain good performance [14]. It has been observed that class imbalance may cause a significant deterioration in the performance attainable by standard learners because these are often biased towards the majority class [15]. Traditional classification algorithms are designed to look for either bigger classes or classes with the similar size. These algorithms when used to identify smaller class from the data either fails to detect or gives erroneous results. As a result, the classifier can achieve a high degree of accuracy on the majority class, and by extension on the overall data set, all the while performing poorly on the minority set [16]. Traditional classifiers tend to be overwhelmed by the majority classes and ignore the minority ones, which is not acceptable in many real applications [17]. Thus, the accuracy metric would no longer be a proper evaluation measure and the derived classifiers may produce misleading information, especially with regards to the minority class [18].

Many approaches have been developed to address the class imbalance problem that are divided into four main approaches: Data level approach, algorithm level approach, Feature based approach and Hybrid (Data+ Algorithm) algorithms [19]. Data level algorithms convert the data into a balance data set by pre-processing with either oversampling or undersampling or hybrid sampling. In undersampling approach, the data balanced by decreasing the size of majority class by randomly removing the data observations from the majority class [19, 20] whereas in oversampling approach, data is balanced by increasing the size of minority class either by copying the existing data [19, 21, 22] or by using some other oversampling methods such as SMOTE and SMOOTEBoost. After balancing the data in either approach, the classification procedures are applied to classify the new data set. In Hybrid sampling approach the combination of oversampling and undersampling approaches are used to pre-process the data before classification [23]. In algorithm level approach, to modify the sensitivity of the algorithm towards the majority class, the internal structure of the traditional classification procedure is changed, or a new method is developed for class imbalance situation. [17, 24, 25]. Beside the Data level approach and Algorithm level Approach, the Feature based selection is another important method that can alleviate the class imbalance problems by working with the performance criteria and correlation. Highly co-related feature can result into a more accurate partitions [26]. In the Hybrid approach, a combination of algorithms or data level methods are used together with the ensemble approaches like bagging, boosting, random forest, etc., [27].

In this paper, the proposed data level approaches, undersampling, oversampling and hybrid, are tested with two imbalance financial data sets.

This paper is organized as follows: Section 2 reviews the resampling techniques, Section 3 briefly defines the classifiers which are used in this paper. Section 4 explains the data features, and methods, Section 5 presents analysis and the results. Finally, Section 6 concludes the paper.

2 Resampling Methods

To overcome the effect of imbalance on performance metrics, various methods have been developed. One of them is data level approach that attempts to balance the class distribution [16, 28]. Resampling modifies the data set to reduce the discrepancy among the sizes of classes [19]. This approach contains three resampling techniques like undersampling, oversampling, or hybrid (a combination of both) sampling.

2.1 Undersampling

The aim of undersampling is to randomly eliminate majority class instances until the number in this class equals that in the minority class [29, 30], or up to a threshold value that the data set can tolerate [29,31]. There are several undersampling approaches such as the condensed nearest neighbor rule(CNN), Wilson Edited Nearest Neighbour Rule (ENN), Neighbourhood Cleaning Rule (NCL), One-sided selection, Tomek link, and One-sided selection (OSS) [16,32]. These approaches help to find out border, noise, and redundant samples by certain rules and strategies, and selectively remove the majority class samples and retain the safe samples and small class samples as the training set of the classifier [33]. The only disadvantage of this method is that the effective information of majority class can be lost easily or may be the important information can be omitted from the majority class.

In this paper, a different approach other than the literature is proposed for undersampling. There are five steps for this process: In step 1, upload the majority class data to SPSS and select some number of samples with similar size of minority class, randomly. In step 2, check the normality requirements for each sample and majority classes. In step 3, check if the normality requirements are satisfied, compare the difference between majority class and run each sample with t test, if they are not satisfied, use nonparametric tests to check the samples are good enough to represent the majority class without losing any important features. In step 4, check if the samples selected in the third step are the good representatives of the majority class, which means each majority class does not lose the effective information, then select one of the samples as undersampled data set. In step 5, classify the new balanced data set by different classification algorithms in order to find out which classification algorithm best classify the data set by comparing the performance measures.

2.2 OverSampling

Oversampling is another common sampling approach used to deal with an imbalanced class problem [16]. The goal of this phase is to balance the highly imbalanced class distribution of the given problem by replicating instances of the minority class [34]. Various oversampling approaches are available including random oversampling, focused oversampling, and synthetic sampling [35]. Random oversampling has two major shortcomings: it increases the possibility of overfitting of the classifier on the training data set, and if the original data already has high dimensionality, it mounts the computation cost thus increasing the training time of the classifier. Chawla (2002) proposed the SMOTE algorithm, which has good performance in oversampling processing of sample sets [33]. This algorithm can randomly create and generate new minority class sample points based on a certain rule, and merge these newly generated sample points with the original data set in order to generate new training sets [33]. SMOTE is a technique in which oversampling of the minority class is carried out by generating synthetic examples. The process of SMOTE is to calculate the nearest same class neighbours for every minority example and then based on the required oversampling rate, randomly choose from these neighbouring examples [16]. The synthetic examples are then generated at random points along the line segments joining the minority examples with these chosen neighbours [16]. Many algorithms have been proposed to improve SMOTE, such as SMOTE-SMV, WEMOTE [36, 37], and SMOOTEBoost [38].

This paper replicates the original minor class data with different ratios. The number of replicas of each instance is referred to as the replication factor. For example, a replication factor of 1 means that there is only one copy of each instance of minority class, a replication factor of 2 means two copies of each instance and so on [34].

2.3 Hybrid

Hybrid method is the combination of undersampling and oversampling methods. In this method, some of the minority class instances expanded by the oversampling method in order to eliminate overfitting [38]. SMOTE-Tomek links is one of the hybrid sampling approaches. SMOTE-Tomek links applied to the oversampled training set as the cleaning method, so instead of removing only the majority class instances, instances from both classes are removed [40]. SMOTE-ENN is another method which is similar to SMOTE-Tomek Links method. ENN tends to remove more instances than the SMOTE-Tomek links does, so it is expected to provide a more in-depth data cleaning. In contrast to NCL, which modifies the ENN to increase the data cleaning, ENN is used to remove instances from both classes. Thus, any instance that is misclassified by its three nearest neighbors is removed from the training set [40]. Another method is Borderline-SMOTE1 which is only oversamples or strengthens the borderline minority instances [38]. This approach, first finds out the borderline minority examples *P*; then, synthetic instances are generated from them and added to the original training set [39]. On the other hand, another approach, Borderline-SMOTE2, is not only generates synthetic instances from each example in DANGER and its positive nearest neighbors in *P*, but also does so for its nearest negative neighbor in *N* (majority class) [39]. Safe-Level-SMOTE approach which assigns each positive instance its safe level before generating synthetic instances [41]. Each synthetic instance is positioned closer to the largest safe level so that all synthetic instances are generated only in the safe regions [41].

In this work, two financial data sets are both classified into three imbalance classes, one minority and two majority classes. To balance the number of instances in each class, it is proposed to change the definitions of class limits. Changing class definitions of binary classes with strict definitions may not work, for example, a cancer search data set, a patient has a cancer or not, it cannot be said both. However, it can be applied for the multiclass data if there are no strict definitions, like financial data. With the new majority class definitions, the instances that are very close to minority class instances in majority class are added to minority class without losing the effective information of each class. This process is applied several times to get balanced classes. Thus, with the new definition, minority class will be oversampled while the majority class will be undersampled. Then, the performances of classification algorithms are tested and compared for the new data set. In this paper three definitions are proposed to solve the imbalance problems efficiently.

The first definition is implemented with the following steps: 1-Search for the values of the instances in the majority class which are very close to the value of the instances in minority class in order to keep the same features of both majority and minority classes. 2-Check for which interval values in the balanced classes can be obtained among different trials in step 1. 3-Select the interval length in the minority class as small as possible among the trials. Then change the definition of the minority class accordingly. 4-Remove the instances in the defined intervals of the majority classes and add them to the minority class. 5- Employ the classification algorithms to the new data set and compare them by their performance measures to find out the best classification algorithm.

The second definition is implemented with the following steps: 1-If your data set is numerical, convert it into percent change format. If your data set has percent changes as the values, use your data set for the hybrid method. 2- Find small percent change intervals which are very close to minority class. 3- Try different small percent change intervals. 4- Add the values which fall in those percent change intervals to the minority class. 5-Check the number of instances or percentages of each class. 6- Decide which trial gives more accurate class percentages.

Finally, the third definition follows the following steps: 1-Remove extreme values to obtain a symmetric data set. This definition is applicable for imbalance binary classes or three imbalance classes if the minority class is in the middle. 2-Compute mean of the data set, 3- Find different percent areas around the mean. 4-Add the instances that fall into those percent areas to the minority class. 5- Select the best percent area that does not affect the majority class features.

3 Classification Algorithms

There are many methods in literature to classify big data. Classification is one of the commonly used Data

Mining (DM) method [42]. It is a process of partitioning data into different classes or groups and collect the items into target classes aiming to predict the target class for the data [42-44]. There are many different classifiers or algorithms. Among those, it is not exactly known which algorithms will perform most efficiently and accurately in any given case. To find out which algorithms classify the data more accurately, at least some of the widely used one should be run [42 ,44]. In this paper, one of the DM software, WEKA, will be used as a classification tool. In the rest of this section, some properties of WEKA software, classifier performance measures and nine classification algorithms used in this paper are summarized.

3.1 WEKA

WEKA is an open source DM software developed by the University of New Zealand [42]. It implements data mining algorithms using a java language and supports several standard data mining tasks, more specifically, data pre-processing, clustering, classification, regression, visualization, and feature selection. There are many advantages of using WEKA, namely 1) It is fully implemented in the Java programming language, therefore it runs on almost any architecture; 2) it is easy to use due to its graphical user interface; 3) It has a huge collection of data pre-processing and modelling techniques [44, 45]; 4) It supports multiple dataset format like csv data files, Json Instance files, libsvm data files, Matlab ASCII files etc., with the default being ARFF. There are three steps to classify the data: 1- prepare the data, 2- select and apply appropriate algorithm, 3- analyze the results [45].

3.2 Classifier Performance Measures

The following performance measures can be used to evaluate the operation of the algorithm, such as Kappa statistic, accuracy, square mean error, ROC curve, confusion matrix, precision, recall and so on. The results of analysis are summarized in the output in the form of summary tables and graphical formats. The classification accuracy is measured as the percent ratio of the number of correctly predicted data points to the total number of data points [44]. In literature, 80% is assumed as the threshold accuracy point [46] for financial data. For imbalance data, accuracy is an inappropriate performance measure because there is no balance of the instances in the majority and minority classes. Besides accuracy, precision, recall or their harmonic mean F statistic can be used as a performance measure. Precision and recall are both performance measures that can be used for both binary data and multiclass data. Precision quantifies the number of correct positive predictions made whereas recall quantifies the number of correct positive predictions made from all positive predictions that could have been made. In an imbalanced classification problem with more than two classes, precision is calculated as the sum of true positives across all classes divided by the sum of true positives and false positives across all classes [47], thus it calculates the accuracy for the minority class. However, recall is calculated as the sum of true positives across all classes divided by the sum of true positives and false negatives across all classes, thus it calculates the accuracy for majority class. Maximizing precision will minimize the number of false positives, whereas maximizing the recall will minimize the number of false negatives [47]. Since neither precision nor recall can provide all information, the harmonic mean of precision and recall is used as a single performance measure as suggested, it is the most variant measure when learning from imbalanced data [48]. Like precision and recall, a poor F-Measure score is 0.0 and a best or perfect F-Measure score is 1.0 [47]. Another performance measurement of classifier is Kappa statistic which is used to indicate the agreement between the model’s prediction and true values. Kappa statistic measures the inter-rater agreement for categorical items and has value that ranges from 0 to 1 [44, 49]. It has been suggested the kappa result can be interpreted as degree of agreement based on the following values: values ≤ 0 indicates no agreement and 0.01–0.20 as none to slight, 0.21–0.40 as fair, 0.41–0.60 as moderate, 0.61–0.80 as substantial, and 0.81–1.00 as almost perfect agreement [42, 50]. The other measure is confusion matrix which is a technique for summarizing the performance of classification algorithm. It presents a visualization of the classification performance based on a table that contains columns representing the instances in a predicted class and rows representing the instances in an actual class. Another performance measure is ROC curves which measures overall performance of the classifier according to the area under the curve. It can be used to compare two or more class performances. The area under the curve is the highest and the best classifier. The range of values for the area under the curve changes from 0 to 1. 1 indicates the classifier is perfect. A ROC curve can be used to select a threshold for a classifier which maximizes the true positives, while minimizing the false positives [48].

3.3 Classification Algorithms

WEKA has various classification algorithms. Classification in WEKA 3.9.4 contains seven different types of classifiers: Bayes, Functions, Lazy, Meta, Misc, Rules and Trees [44, 51]. Each classifier contains different number of algorithms. In addition, nine of algorithms were used in this study, Bayes Net (BN), Navie Bayes (NB), J48, Random Forest (RF), Meta-Attribute Selected Classifier (MAS), Meta-Classification via Regression (MR) and Meta-Logitboost (ML), Logistic Regression (LR), Decision Tree (DT) [43]. The definitions of classification algorithms are taken Ruzgar (2019) who used the same data sets but with different objective as in this study [42, 44].

Bayesian classifiers, BN and NB, are both probabilistic algorithms. BN are directed acyclic graphs (DAG) whose nodes represent random variables. The nodes can be any observable quantities, variables, unknown parameters, or hypotheses [42, 51-53]. Edges are the conditional dependencies. Nodes which are not connected represent the independent variables. Each node is associated with a probability function that takes input a set of values for the node's parent variables and gives the probability of the variable represented by the node [42, 53]. NB is a probabilistic classification algorithm using estimator classes, where numeric estimator precision values are chosen based on the analysis of the training data [44]. J48 induces classification rules in the form of a pruned/unpruned decision tree. J48 creates a decision node higher up in the tree using the expected value of the class. It can handle both continuous and discrete attributes, training data with missing attribute values and attributes with differing costs. Further it provides an option for pruning trees after creation, while RF is bagging of Random Trees [44, 54]. Meta classification indicates the usage of a combination of multiple classifiers. This combination is carried out in three steps: In first step, multiple training subsets are constructed from a training set. In second step, each classifier is solely constructed according to both the algorithm and data training subset. In third step, the results of base classifiers are integrated, and results are obtained in a higher-level step called Meta classifier [44, 55]. There is also a Multiclass Classifier Meta classifier that does this for any binary class classifier [44, 55]. With MAS Algorithm, the range of the training data and testing data is lessened by this algorithm before being departed onto the classifier. The classifier is raised, so various search approaches are used during the phase of attribute selection. ML is a boosting algorithm and is an extension of Adaboost algorithm. It replaces the exponential loss of Adaboost algorithm with conditional Bernoulli likelihood loss. This Class is used for performing additive logistic regression. [55]. MR uses regression approach for classification. Finally, LR is a classifier building the linear logistic regression models. LogitBoost with simple regression functions as base learners is used for fitting the logistic models [44].

4 Methology

This paper is set up to four objectives. The present work aims to show how the multiclass imbalance data can be classified from different point of view using resampling methods. The objective is to find out how the performance metrics of nine classification algorithms are influenced from the resampling methods and which classification algorithm among nine classification algorithms shows the best performance. The third aim is to find which classification algorithm classify the stock price changes for the two Canadian banks more effectively and accurately when they are employed to the imbalance multiclass data with nine algorithms by comparing the classifier’s performance measures, such as recall, precision, kappa statistic and ROC curve. The fourth aim is to show how the different point of views for resampling methods can be applicable to multiclass imbalance data.

4.1 Data

Stock market prices are very important parameters for the investors. They would like to invest on any financial instrument which gives more profit from the stock market [42, 44]. The price changes affect their investments. For this purpose, two large Canadian banks’, TD and RBC banks, stock market daily prices were collected from NASDAQ [56] over the period from 1980 to 2017. Each data set has twenty-one independent variables, and one dependent variable. Independent variables are Daily Opening price, Daily Opening bid, Daily Opening ask, Daily Closing price, Daily Closing bid, Daily Closing ask, Daily High, Daily Low, Daily Transactions, Daily Volume, Daily Quotes, Daily Quote changes, Daily Return, S&P/TSX Composite Price Index, S&P/TSX Composite Total Return Index, Sector 40 (Financials) Price Index, Sector 40 (Financials) Total Return Index, S&P/TSX 60 Price Index, S&P/TSX 60 Total Return Index, Call Loan Interest Rate and Foreign Exchange Rate (CA$/US$) [42, 44]. The independent variables are all numerical. However, the dependent variable is categorized into three groups, “up”, “same” and “down” even though the daily stock market closing price variable is numerical. The change is measured by comparing the daily stock market closing price with the previous day's stock market closing price. If the closing price higher relative to the previous day's closing price, "up" was assigned as response to the new variable component, if the closing price lower relative to the previous day's closing price, "down" was assigned as a response to the new variable component, and similarly, if the closing price remained the same, "same" was assigned as a response to the new variable component [42, 44]. These three responses represent the multiclass data in our study. Three resampling methods for handling the class imbalance problem were applied to the imbalance multiclass data with different point of view by employing nine classifiers with four performance measures. WEKA, 3.9.4 is used to facilitate the proposed analysis for the nine classifiers.

Classification is a two-step process. In the first step, the training data are analyzed with a classification algorithm. In the second step, test data are used to estimate the accuracy of the classification rules. Classification algorithms were applied to each original training set and resampled data sets. Each time running the system, 10-fold cross validation was carried out. Cross validation includes splitting the data into 10 equal parts with 9 parts used in the training phase and the remaining part employed in testing [16]. Results obtained in terms of the four-performance metrics were evaluated for each data set.

5 Findings

5.1 Method 1: Undersampling

In this paper, a different approach other than the undersampling methods studied in literature is proposed for multiclass imbalance data. For the proposed method, the following steps are followed. In the first step, each majority class data is uploaded to a statistical software and 5 samples are randomly sampled. The different percentages of data are selected from each majority classes to best fit the similar number of instances in minority class and is confirmed that 20% of each majority data corresponded to similar number of instances in minority class. According to the random sampling, the new instance percentages of the classes are found as "Up", 35.22% (928), "Down", 32.26 % (850) and "Same" 32.52 % (857) for TD bank and "Up", % 35.14 (910), "Down", % 33.09 (857) and "Same" % 31.78 (823) for RBC bank. In the second step, each sample was tested statistically if they are good representative of the majority classes. Statistical results showed that each sample satisfied the conditions and can be used to represent the majority classes. This confirmed that each sample can be used instead of using the original majority classes without losing the effective information of the majority classes which is very important fact for undersampling. In the fourth step, one out of 5 samples from each majority class was randomly selected as a class representative. Finally, in the last step, nine classification algorithms are employed on the new data sets of TD and RBC banks. The performance measures result of the classification algorithms are illustrated in Figures 1 to 9.

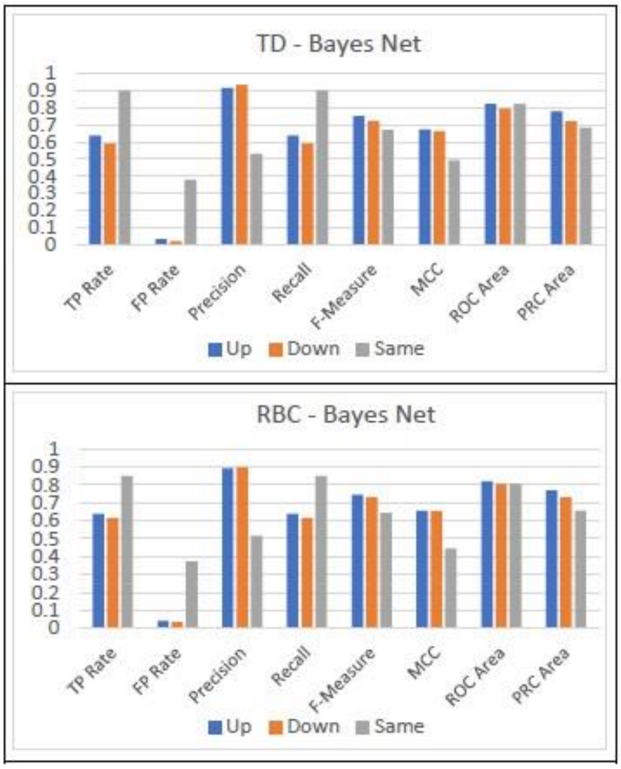


Fig 1. Performance measure comparison of Bayes Net (BN)

Precision measures of the classes are equally distributed on both datasets (Fig 1). Minority class precisions are low when compared with the majority classes. It is obvious that the recall measures are opposite to the precision measure values. The area under the ROC curves of all classes on both datasets are in the range of 0.795-0.82. However, the majority classes' F measures of both datasets are greater than the minority class. There is a big difference between the true positive rates of minority class and majority classes on both datasets.

The precision values are not equally distributed among the classes, from the majority classes. "Down" has the highest value in the range of 0.893-0.876 and "Up" has around 0741-0.772 for TD and RBC data respectively, while it is in the range of 0.502-0.492 for the minority class (Fig 2). The area under the ROC curve of minority class is a little bit higher than the majority classes for TD data, whereas it is approximately same for the RBC data. The F measures of classes are not equally distributed for majority and minority classes of both data sets. There is a big gap between true positive rates of minority and majority classes for both data sets.

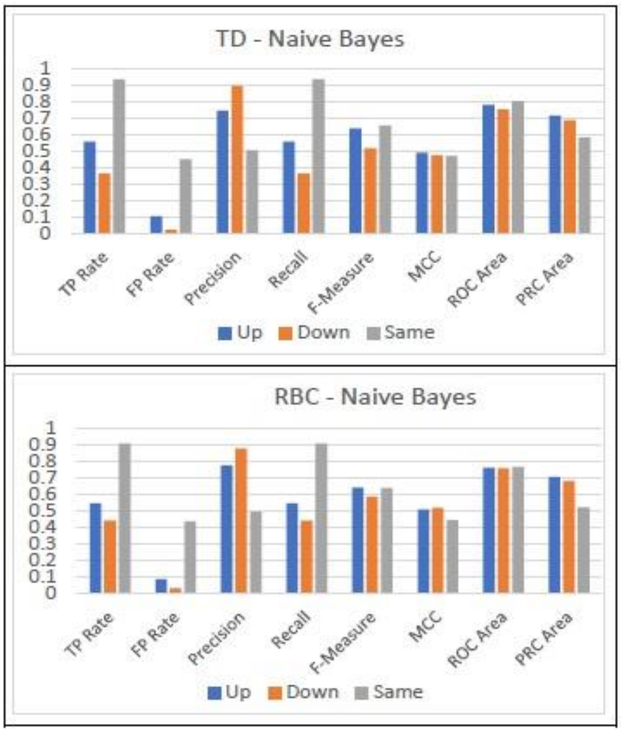


Fig 2. Performance measure comparison of Navie Bayes (NB)

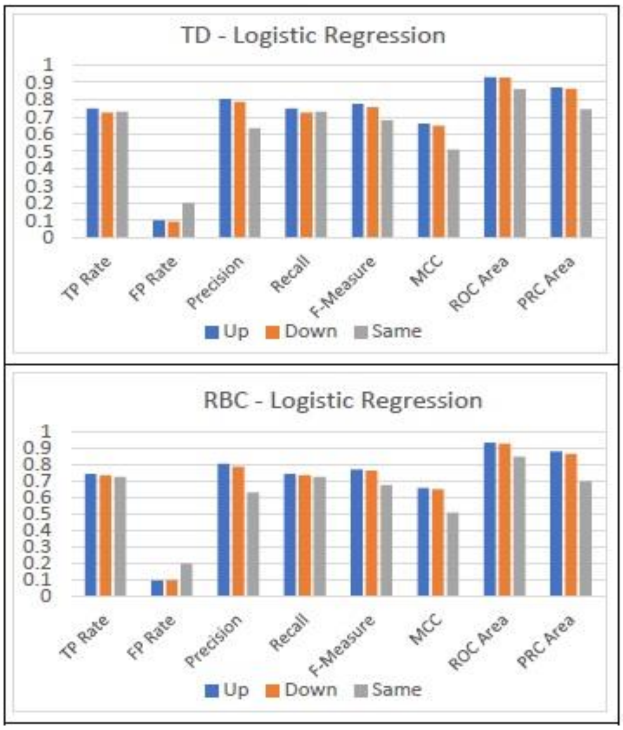


Fig 3. Performance measure comparison of Logistic Regression (LR)

Both data sets show similar performance measures for LR; area under the ROC curve approximately 0.82 for minority class and 0.92 for the majority classes (Fig 3). Similarly, the F-measures are around 0.68 for minority class and 0.77 for majority class and the precisions are 0.68 for minority class and 0.79 for majority classes. Their true positive rates change from minority to majority class from 0.71 to .073 whereas the false positive rates change from majority to minority classes from 0.99 to 0.2. Thus, the performance measures of LR are better than the performance measures of BN and NB.

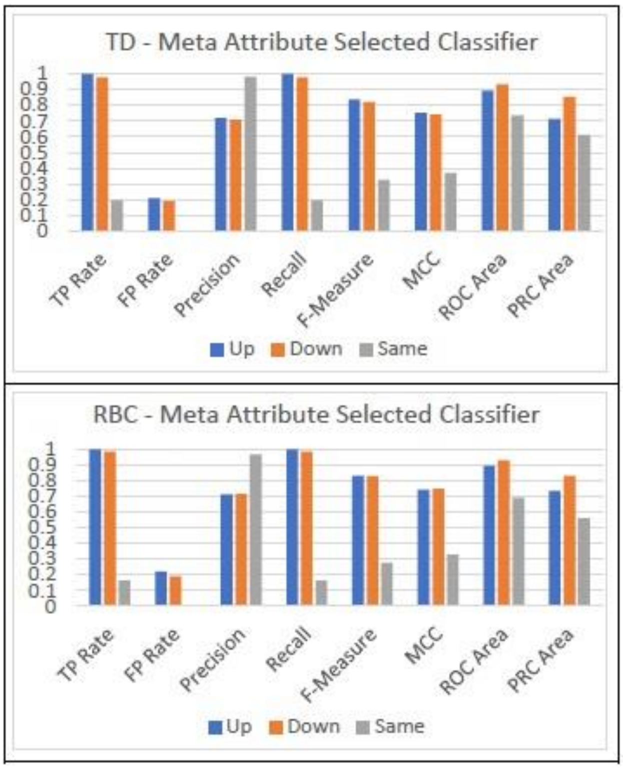
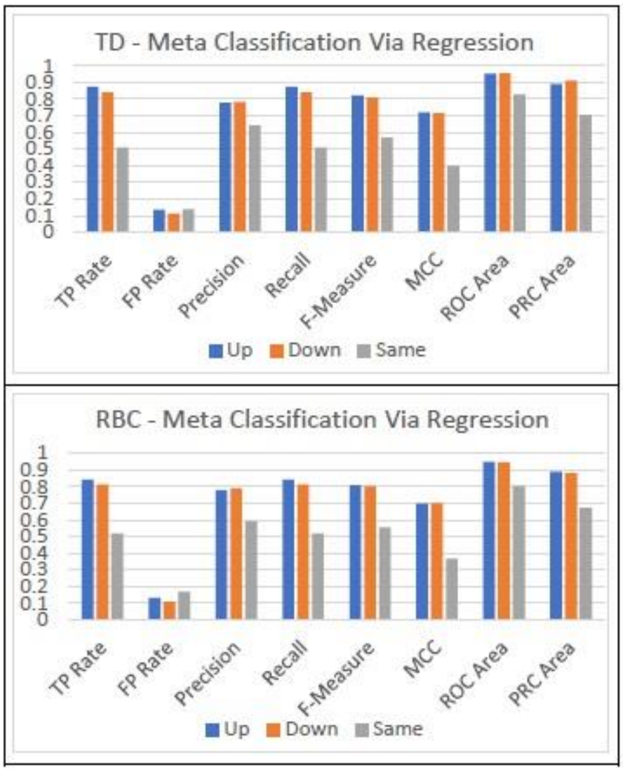


Fig 4. Performance measure comparison of Meta Attribute Selected Classifier (MAS)

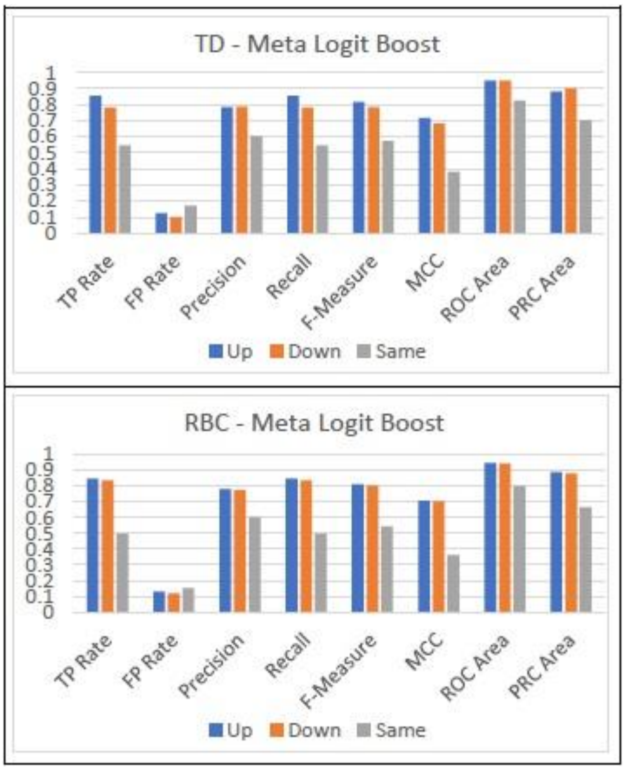
Both TD and RBC data show the same performances; the true positive rates of majority classes are very high whereas minority class value is very low (Fig 4). The precision measures of the majority classes are around 20% less than the minority class value. The distribution of the areas under the ROC curves are not equivalent according to the classes, minority class value is 20% less than the majority classes.

For both data sets, precision values are equally distributed for the majority classes, but minority class value around 28% less than the values of the majority classes (Fig 5). The area under the ROC curve and F measures for both data sets show the similar pattern as the precision values, majority classes are having equal values whereas minority class has less value.


TD - Meta Classification Via Regression

RBC - Meta Classification Via Regression

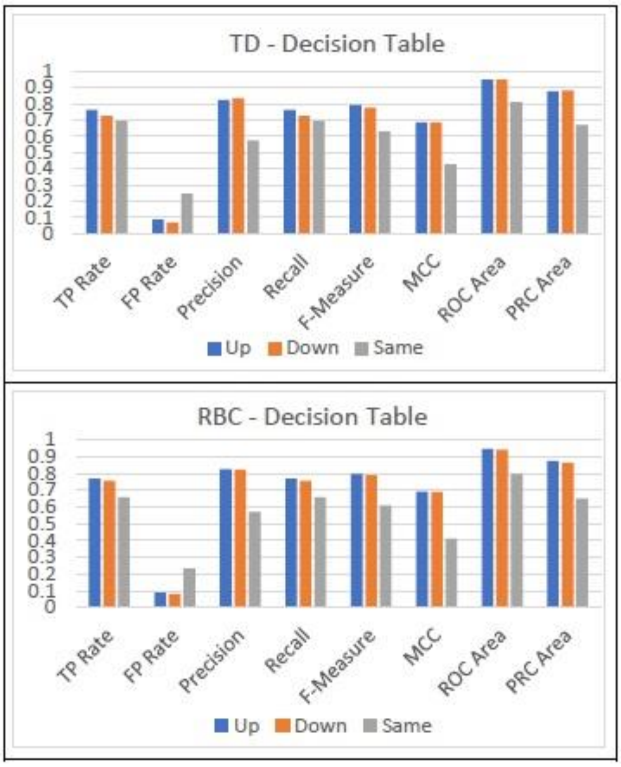
Fig 5. Performance measure comparison of Meta Classification Via Regression (MR)


TD - Meta Logit Boost

RBC - Meta Logit Boost

Fig 6. Performance measure comparison of Meta Logit Boost (ML)

The performance measures of ML are similar to the performance measures of MAS for both data sets, majority classes greater measures than that of minority class (Fig 6). The good thing is that the area under the ROC curve is more than 0.90 for majority classes and approximately 0.82 for the minority class. True positive rates change between 0.80 and 0.85 for the majority classes, it changes in the range of 0.49-0.54 for the minority class. F measures for both data sets are close to each other for the majority classes about 0.80, whereas it is around 0.54 for the minority class.


TD - Decision Table

RBC - Decision Table

Fig 7. Performance measure comparison of Decision Table (DT)

DT has better performances than the other classification measures discussed before (Fig 7). The area under the ROC curves is around 0.94 for the majority classes and 0.80 for minority classes. F measures are approximately 0.80 for majority classes, 0.60 for minority class. The precision values are more than 0.80 for the majority classes and 0.57 for the minority class. The true positive rates of the minority and majority classes are similar to the LR, they are close to each other around 0.72.

The true positive rates of TD data are not equally distributed, but close to each other, even though the distances are very high for RBC data (Fig 8). For both data sets, the area under the ROC curve are greater than 0.93 for the majority classes, but it is around 0.76 for the minority classes. F measures are approximately 0.79 for the majority classes and 0.55 for the minority class. On the other hand, the precision measures are the same for the majority classes and less for the minority class for TD data set.

For RBC data set, however, the majority classes have unequal precision values, 0.82 and 0.72, and the minority class has less precision value, 0.59 like TD data set.

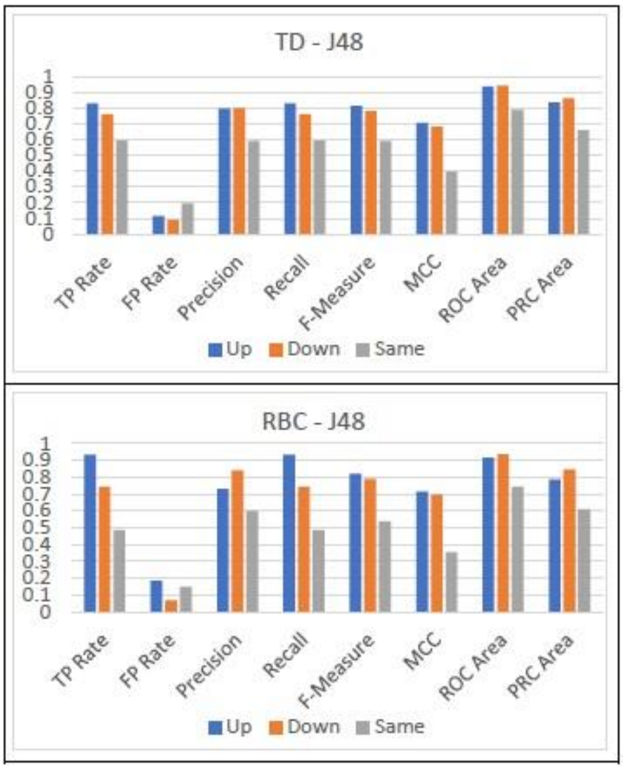


Fig 8. Performance measure comparison of J48

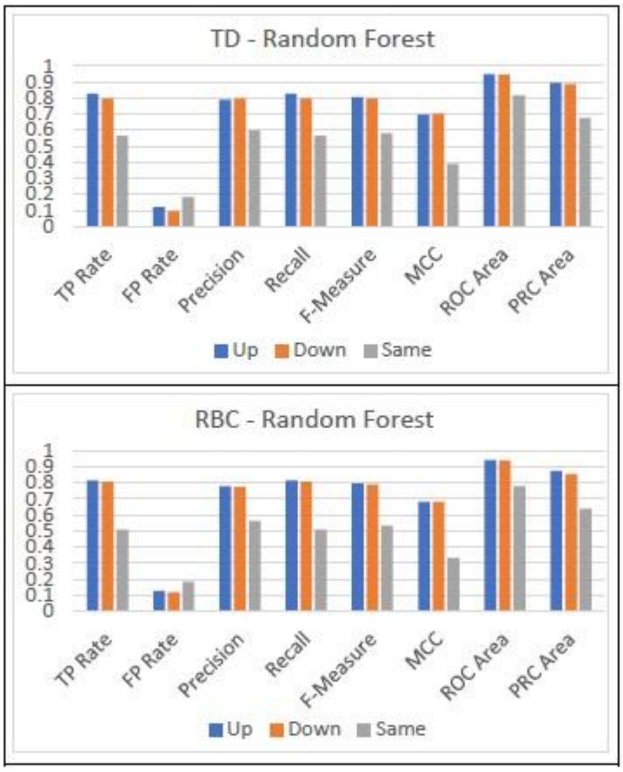


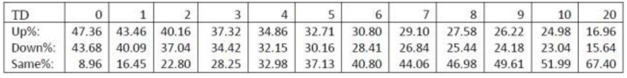
Fig 9. Performance measure comparison of Random Forest (RF)

The performance measures of RF are similar to the J48 (Fig 9); the area under the ROC curves are more than 0.94 for the majority classes, and 0.80 for the minority class for both data sets. The precision values are approximately 0.78 for the majority classes, and 0.58 for the minority class. The true positive rates also show the same pattern for both datasets, around 0.79 for the majority classes and 0.53 for the minority class. Similarly, F measures are around 0.80 for the majority classes and 0.55 for the minority class.

5.2 Method 2: Oversampling

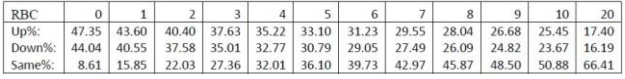
Traditional classification algorithms tend to be overwhelmed by the majority classes and ignore the minority ones, which is not acceptable in many real applications [17].The goal of oversampling is to balance the highly imbalanced class distribution of the given problem by replicating randomly the instances of the minority class [34]. Although there are different approaches to replicate the instances of minority class with different ratios, in this work, we only replicated the instances in the original minority class from 1 to 20 times keeping the original class. features same, respectively. The number of replicas of each instance is referred as the replication factor. For example, a replication factor of 1 means that there is only one copy of each instance of minority class, a replication factor of 2 means two copies of each instance and so on [34]. This replication factor is calculated with the total majority class instances and the total instances of the class of the instance that we want to replicate [34].

Table 1. The replication factors by the instance percentages of TD



TD	0	1	2	3	4	5	6	7	8	9	10	20
Up%:	47.36	43.46	40.16	37.32	34.86	32.71	30.80	29.10	27.58	26.22	24.98	16.96
Down%:	43.68	40.09	37.04	34.42	32.15	30.16	28.41	26.84	25.44	24.18	23.04	15.64
Same%:	8.96	16.45	22.80	28.25	32.98	37.13	40.80	44.06	46.98	49.61	51.99	67.40

Table 2. The replication factors and the instance percentages of RBC



RBC	0	1	2	3	4	5	6	7	8	9	10	20
Up%:	47.35	43.60	40.40	37.63	35.22	33.10	31.23	29.55	28.04	26.68	25.45	17.40
Down%:	44.04	40.55	37.58	35.01	32.77	30.79	29.05	27.49	26.09	24.82	23.67	16.19
Same%:	8.61	15.85	22.03	27.36	32.01	36.10	39.73	42.97	45.87	48.50	50.88	66.41

Table 1 and Table 2 illustrate the replication factors and the instance percentages of TD and RBC banks, respectively. The instance percentages of TD price changed for “Up” from 47.36% to 16.96%, for “Down” from 43.68% to 15.64% and for “Same”, which is the minority class, from 8.96% to 67.40% by consecutive 20 replications of the minority class. Similarly, the instance percentages of RBC price changed for “Up” from 47.35% to 17.40%, for “Down” from 44.04% to 16.19% and for “Same”, which is again the minority class of RBC data, from

8.61% to 66.41% by consecutive 20 replications. From those 20 replications of minority class instances, the approximate balance of the three classes for both TD and RBC data has been reached in the fourth or fifth replication.

Now, to see how the performance measures of classification algorithms can be influenced from those replications, the nine algorithms were employed for each replication of TD and RBC data. Then the comparison of the performance measures of eight algorithms are depicted in Figures 10 to 13 with precision, recall, kappa statistics and ROC curve for TD and RBC data, respectively.

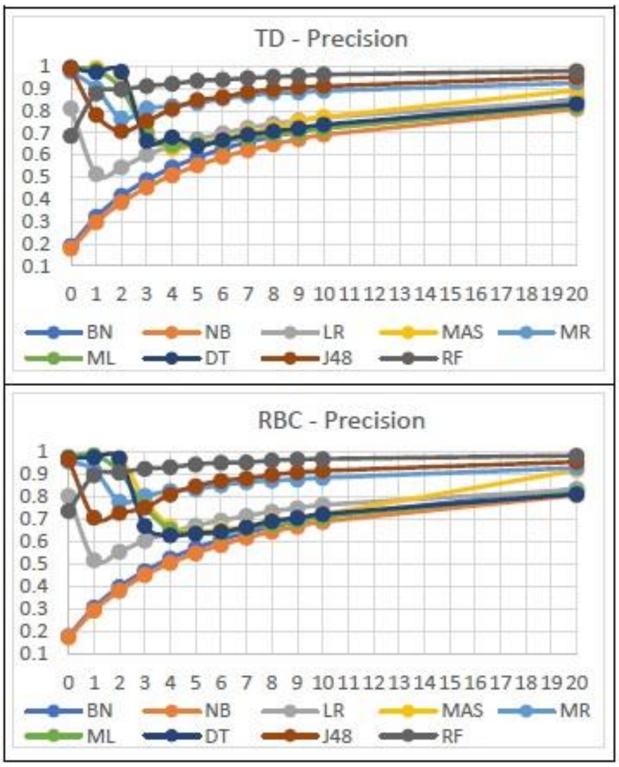


Fig 10. Precision results of nine classification algorithms for replications

Fig 10 shows that precision measures of BN and NB algorithms for both TD and RBC data are positively affected from the replications, precision starts from 0.177 for TD, 0.173 for RBC data and increases smoothly towards 0.807 for RBC and 0.809 for TD. For both data sets of TD and RBC banks, LR classification algorithm is seriously affected in the first replication, dropping from 0.81 to 0.52, but then goes up to 0.845 and 0.831 through replications, respectively. After this, while interpreting the data for TD and RBC, the results for RBC will be written in parenthesis after the TD value. Precision measures of J48 for TD (RBC) starts with 0.988 (0.964) precision, but slightly drop until the second (second) replication to 0.708 (0.707), then increases to 0.953 (0.955), whereas the precision of MR starts with 0.977 (0.957), then drops to 0.765 (0.778) in the second (second) replication and then increases smoothly to 0.926 (0.928) in the 20th replication. Precision value for ML starts at 0.994 (0.978) it drops smoothly to 0.645 (0.637) in the seventh (sixth) replication, then increase to 0.823 (0.818) in subsequent replications. For DT, precision starts from 0.994 (0.971), then decreases up to the sixth (fourth) replication to 0.642 (0.628), starts to increase in the seventh (fifth) replication to 0.831 (0.811). Similar to the others, precision value for MAS starts at 0.988 (0.965), then drops until the fourth (fifth) replication to 0.629 (0.643), and then increases to 0.893 (0.918) in the twentieth replication. Finally, precision value for RF starts at 0.687 (0.736) and increases smoothly to 0.980 (0.983) in the twentieth replication. According to the precision values for all classification algorithms, BN, NB and DT are positively affected, LR, MAS, ML and DT are seriously negatively affected and MR and J48 are less affected. For seriously negatively affected algorithms, precision decreases until the fifth or sixth replications whereas for the less affected algorithms, precision decreases until the first - third replications. Among the nine classification algorithms, DT reflects the best precision change throughout the replications.

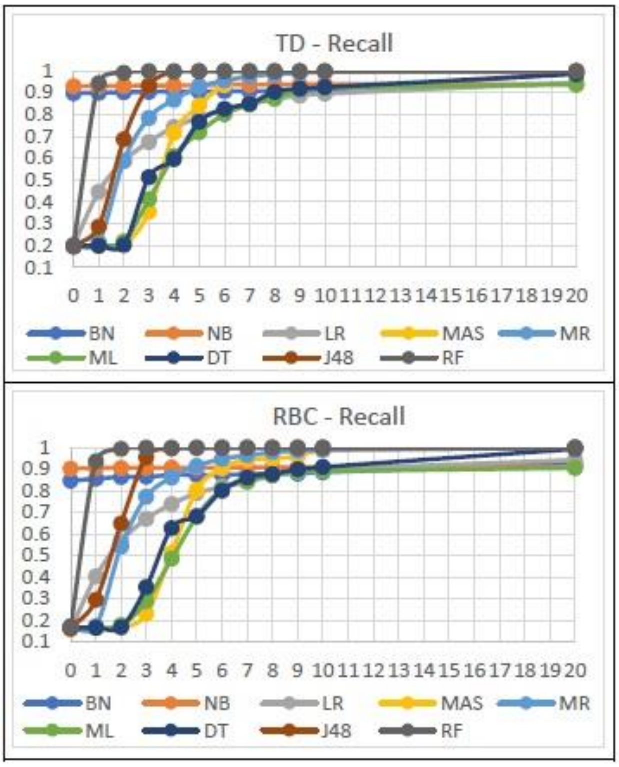


Fig 11. Recall results of nine classification algorithms for replications

According to the recall values for all classification algorithms in Fig 11, there is no effect on BN and NB whereas the MAS, ML and DT are seriously affected, they stay the same or decrease very small amount until the second replication, then increase toward the value in the range of (0.938, 1) in the twentieth replication. LR, MR and J48 show the similar pattern starting with value in the range of (0.198, 0.2) slightly increase to fifth replication then smoothly increase to the value in the range of (0.946, 1) in the 20th replication . Besides, for RF, after the first replication it reaches the perfect recall value in the third replication.

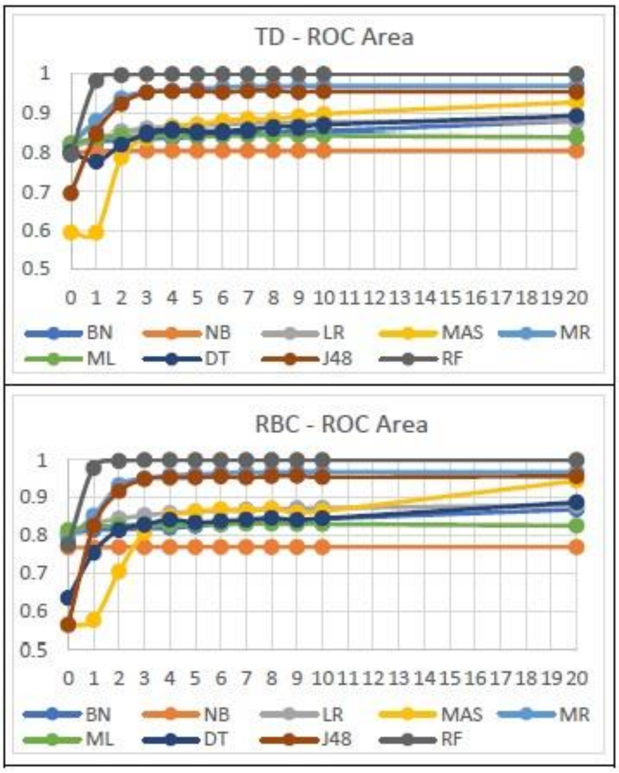


Fig 12. ROC area results of nine classification algorithms for replications

Fig 12 depicts the area under the ROC curve for classification algorithm versus replication. It is found that RF is the best classification algorithm based on the overall performances for both TD and RBC data. It reaches the perfect value in the second replication. Through all replications, MR and with very close value of MAS, J48 have the highest areas under the ROC curve, though J48 has a small decrease in the fifth replication. Although the precision and recall values for BN have the same pattern, area under the ROC curve remains stable for NB while it increases for BN. The results showed that NB and RF are not affected from the replications, while the other algorithms are affected for both data set.

Kappa statistic is another performance measure of classification algorithm, which measures the inter-rater agreement for categorical items. Fig 13 shows Kappa statistics of nine classification algorithms with each replication for minority class instances.

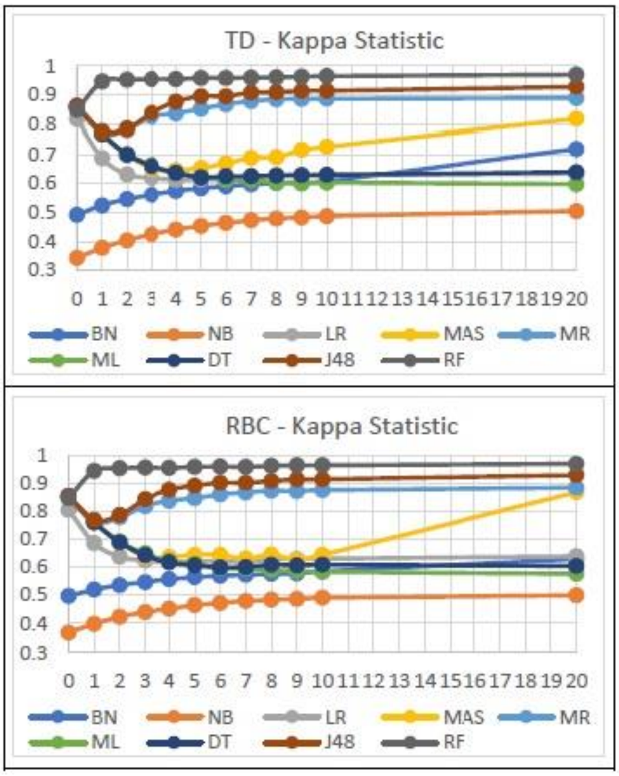


Fig 13. Kappa statistic of nine classification algorithms for replications

It is seen that both TD and RBC data show the same pattern for Kappa statistic. The RF seems the best classification algorithm, J48 and MR follow next. BN and NB again present the same stable pattern, but their Kappa statistics do not measure the performance as the others through the replications.

5.3 Method 3: Hybrid

Hybrid method is the combination of oversampling and undersampling methods for the class imbalance problems. In this paper, to balance the number of instances in multi class, it is proposed to change the definitions of data limits. The instances in the majority classes that are very close to minority class instances are added to the minority class by changing the class definitions without losing the effective information of each class. The three different definitions are proposed for the definitions of classes. The categorization of the original data is labelled as "Decision 1". The categorization of the new class definition is labelled as "NewDecision 1", "NewDecision 2" and "NewDecision 3", respectively.

Decision 1: The dependent variable of the data originally categorized as follows. The change between daily closing price and the previous day's stock market closing price is computed. If the closing price increased relative to the previous day's closing price, "up" is assigned to the new variable, if the closing price decreased relative to the previous day's closing price, "down" is assigned to the new variable, and similarly, if the closing price remained the same, "same" is assigned to the new variable [42, 44]. With this category definitions, TD bank data is categorized with the following percentages and the number of instances as "Up", 47.36% (4529), "Down", 43.68% (4177) and "Same" 8.96 % (857), and similarly RBC Bank data as "Up", 47.34% (4527), "Down", 44.06 % (4213) and "Same" 8.61% (823). As it is seen, category "Same" is the minority class for both data sets.

To solve the imbalance class problem, class definitions can be changed, so in this study, three different changes are applied to the definitions of classes and they are given below.

NewDecision 1: For the "Same" minority group, the new range is defined instead of taking the difference between the daily closing price and the previous day's stock market closing price as 0. The ranges of data sets for TD and RBC are [-48.43, 6.32] and [-48.79, 8.75], respectively, the majority class data cannot be affected if the small amount of price changes are ignored or removed from the majority class and added to minority class. For this purpose, the values of the price changes in cents fall in the intervals (-10,10), (-12.5, 12.5) and (-15, 15) are tested to get similar number of instances in majority and minority classes by removing the instances in the intervals then adding them to the minority class. When the above definitions applied to the TD data set the number of instances and percentages of the new classes are found for (-10, 10) interval as "Up": 4169 (43.60%) , "Down": 3829 (40.04%), "Same": 1565 (16.37%); for (-12.5, 12.5) interval as "Up": 3376 (35.3%) , "Down": 3068 (32.08%), "Same": 3119 (32.62%); and for (-15, 15) interval as "Up": 3259 (34.08%) , "Down": 2957 (30.92%), "Same": 3347 (35.00%). Similarly, the number of instances and the percentages of each class are found for the RBC data set are for (-10, 10) interval as "Up": 4180 (43.71%), "Down": 3894 (40.72%), "Same": 1489 (15.57%); for (-12.5, 12.5) interval as "Up": 3468 (36.26%) , "Down": 3201 (33.47%), "Same": 2894 (30.26%); and for (-15, 15) interval as "Up": 3370 (35.24%) , "Down": 3113 (32.55%), "Same": 3080 (32.21%).

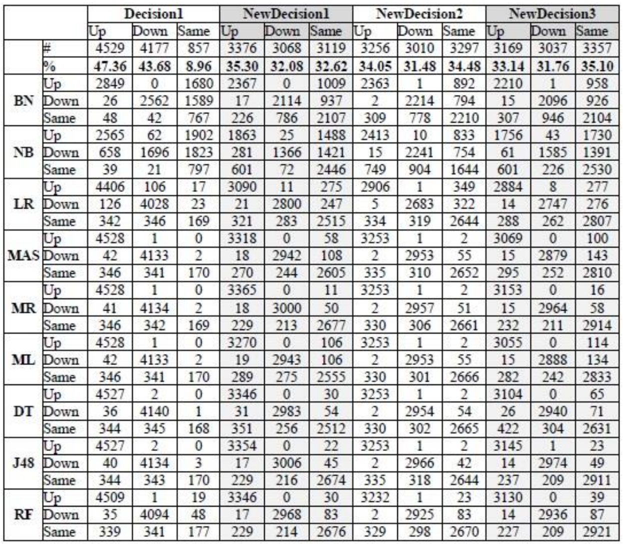
When the number of instances and percentages of the classes are compared for three intervals it is seen that the class imbalance still continues for the interval (-10, 10) , on the other hand, class imbalance disappear for the intervals (-12.5, 12.5) and (-15, 15). Since the general rule is not letting the majority classes lose their main features while balancing classes, the smallest range should be selected for the new definition. In this work, since the range of (-12.5, 12.5) interval is the smallest range compared with the interval (-15, 15), the interval (-12.5, 12.5) is selected. The price changes fall in this interval are removed from the majority classes and added to the minority class.

NewDecision2: For the "Same" minority group, the new range is defined instead of the difference between the daily closing price and the previous day's stock market closing price as 0. Since the percent change of the data set is from -13.11% to 12.05 for the TD data set and from -13.40% to 14.10% for the RBC data set, a small percent of the price changes which are negligible or very close to the value of the minority class, 0, can be removed from the majority classes and added to minority class. Since those values are very small, they do not affect the features of the majority. For this purpose, the price percent changes fall into the intervals between -0.4% and 0.4%, -0.5%, 0.5% and -0.6% and 0.6% are removed from the majority classes and added to the minority classes for both data sets. The number of instances and percentages of TD data set for the percent changes between -0.4% and 0.4% are "Up": 3554 (37.16%), "Down": 3293 (34.43%) and "Same": 2716 (28.4%); for the percent changes between -0.5% and 0.5% are "Up": 3256 (34.05%), "Down": 3010 (31.48%) and "Same": 3297 (34.48%); and for the percent changes between -0.6% and 0.6% are "Up": 2914 (30.47%), "Down": 2709 (28.33%) and "Same": 3940 (41.20%). Similarly, the number of instances and percentages of RBC data set for the precent changes between -0.4% and 0.4% are "Up": 3418 (35.74%), "Down": 3171 (33.16%) and "Same": 2974 (31.10%); for the percent changes between -0.5% and 0.5% are "Up": 2881 (30.13%), "Down": 2711 (28.35%) and "Same": 3971 (41.52%); and for the percent changes between -0.6% and 0.6% are "Up": 2602 (27.21%), "Down": 2414 (25.24%) and "Same": 4547 (47.55%). According to the findings, the percent change interval between -0.5% and 0.5% gives a more accurate number of instances in each class for TD data set, while the percent change interval of -0.4 % and 0.4% gives more accurate number of instances for RBC data set. Hence, the percent change interval between -0.5% and 0.5% is applied to the TD data set, and the percent change interval between -0.4% and 0.4% is applied to the RBC data set.

NewDecision3: For the "Same" minority group, the new range is defined instead of taking the difference between the daily closing price and the previous day's stock market closing price as 0. For resizing the data set, a different approach is used. First, 20 extreme (outliers) values are removed from the data set to obtain a symmetric distribution. Then the mean of the remaining data set is computed. 15%, 20% and 25% of areas about the mean are selected and the instances in these areas are added to the minority class for both data sets.

The number of instances and percentages of TD data set for the 15% area about mean are "Up": 3446 (34.99%), "Down": 3849 (40.25%) and "Same": 2368 (24.76%); for the 20% area about mean are "Up": 3169 (33.14%), "Down": 3037 (31.76%) and "Same": 3357 (35.10%); and for the 25% area about mean are "Up": 3005 (31.42%), "Down": 2880 (30.12%) and "Same": 3678 (38.46%). Similarly, the number of instances and percentages of RBC data set for the 15% area about mean are "Up": 3441 (35.98%), "Down": 3924 (41.03%) and "Same": 2198 (22.98%); for the 20% area about mean are "Up": 3240 (33.88%), "Down": 3156 (33.00%) and "Same": 3167 (33.12%); and for the 25% area about mean are "Up": 3088 (32.32%), "Down": 3019 (31.60%) and "Same": 3456 (36.07%). According to the results, 20% area about mean gives more appropriate number and percent of instances in each class for both data sets. Then, the nine classification algorithms are applied to the new data sets defined using the new definitions, the confusion matrices of each are combined in Table 3 and Table 4 for TD and RBC, respectively.

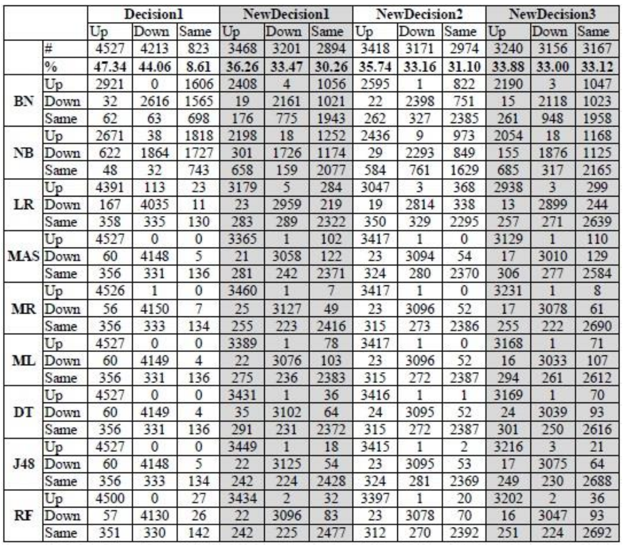
Table 3. Confusion matrices of nine classification algorithms for original and three new decisions for TD data set



		Decision1			NewDecision1			NewDecision2			NewDecision3		
		Up	Down	Same	Up	Down	Same	Up	Down	Same	Up	Down	Same
	#	4529	4177	857	3376	3068	3119	3256	3010	3297	3169	3037	3357
	%	**47.36**	**43.68**	**8.96**	**35.30**	**32.08**	**32.62**	**34.05**	**31.48**	**34.48**	**33.14**	**31.76**	**35.10**
BN	Up	2849	0	1680	2367	0	1009	2363	1	892	2210	1	958
	Down	26	2562	1589	17	2114	937	2	2214	794	15	2096	926
	Same	48	42	767	226	786	2107	309	778	2210	307	946	2104
NB	Up	2565	62	1902	1863	25	1488	2413	10	833	1756	43	1730
	Down	658	1696	1823	281	1366	1421	15	2241	754	61	1585	1391
	Same	39	21	797	601	72	2446	749	904	1644	601	226	2530
LR	Up	4406	106	17	3090	11	275	2906	1	349	2884	8	277
	Down	126	4028	23	21	2800	247	5	2683	322	14	2747	276
	Same	342	346	169	321	283	2515	334	319	2644	288	262	2807
MAS	Up	4528	1	0	3318	0	58	3253	1	2	3069	0	100
	Down	42	4133	2	18	2942	108	2	2953	55	15	2879	143
	Same	346	341	170	270	244	2605	335	310	2652	295	252	2810
MR	Up	4528	1	0	3365	0	11	3253	1	2	3153	0	16
	Down	41	4134	2	18	3000	50	2	2957	51	15	2964	58
	Same	346	342	169	229	213	2677	330	306	2661	232	211	2914
ML	Up	4528	1	0	3270	0	106	3253	1	2	3055	0	114
	Down	42	4133	2	19	2943	106	2	2953	55	15	2888	134
	Same	346	341	170	289	275	2555	330	301	2666	282	242	2833
DT	Up	4527	2	0	3346	0	30	3253	1	2	3104	0	65
	Down	36	4140	1	31	2983	54	2	2954	54	26	2940	71
	Same	344	345	168	351	256	2512	330	302	2665	422	304	2631
J48	Up	4527	2	0	3354	0	22	3253	1	2	3145	1	23
	Down	40	4134	3	17	3006	45	2	2966	42	14	2974	49
	Same	344	343	170	229	216	2674	335	318	2644	237	209	2911
RF	Up	4509	1	19	3346	0	30	3232	1	23	3130	0	39
	Down	35	4094	48	17	2968	83	2	2925	83	14	2936	87
	Same	339	341	177	229	214	2676	329	298	2670	227	209	2921

According to the confusion matrices for Decision 1 in Table 3, J48 and DT best classify the TD data, then in order, RF, MAS, MR, ML and LR classify the original data in similar way, however BN and NB do not when compared with the others. For the NewDecision 1, the classification algorithms MR, MAS, J48, RF,DT, ML and LR in order classify the TD data set in good level, but BN and NB do not classify the data set as good compared to the others. Similar to the classifications for NewDecision1, classification algorithms MR, ML, DT, J48, RF and MAS classify the TD data well under the NewDecision 2. BN and NB are not as good compared to the others. For NewDecision 3, MR, J48, RF, ML, MAS and DT, in order, classify the balanced TD data well, however, LR, NB and BN do not show the similar performances.

Table 4. Confusion matrices of nine classification algorithms for original end three new decisions for RBC data



		Decision1			NewDecision1			NewDecision2			NewDecision3		
		Up	Down	Same	Up	Down	Same	Up	Down	Same	Up	Down	Same
	#	4527	4213	823	3468	3201	2894	3418	3171	2974	3240	3156	3167
	%	**47.34**	**44.06**	**8.61**	**36.26**	**33.47**	**30.26**	**35.74**	**33.16**	**31.10**	**33.88**	**33.00**	**33.12**
BN	Up	2921	0	1606	2408	4	1056	2595	1	822	2190	3	1047
	Down	32	2616	1565	19	2161	1021	22	2398	751	15	2118	1023
	Same	62	63	698	176	775	1943	262	327	2385	261	948	1958
NB	Up	2671	38	1818	2198	18	1252	2436	9	973	2054	18	1168
	Down	622	1864	1727	301	1726	1174	29	2293	849	155	1876	1125
	Same	48	32	743	658	159	2077	584	761	1629	685	317	2165
LR	Up	4391	113	23	3179	5	284	3047	3	368	2938	3	299
	Down	167	4035	11	23	2959	219	19	2814	338	13	2899	244
	Same	358	335	130	283	289	2322	350	329	2295	257	271	2639
MAS	Up	4527	0	0	3365	1	102	3417	1	0	3129	1	110
	Down	60	4148	5	21	3058	122	23	3094	54	17	3010	129
	Same	356	331	136	281	242	2371	324	280	2370	306	277	2584
MR	Up	4526	1	0	3460	1	7	3417	1	0	3231	1	8
	Down	56	4150	7	25	3127	49	23	3096	52	17	3078	61
	Same	356	333	134	255	223	2416	315	273	2386	255	222	2690
ML	Up	4527	0	0	3389	1	78	3417	1	0	3168	1	71
	Down	60	4149	4	22	3076	103	23	3096	52	16	3033	107
	Same	356	331	136	275	236	2383	315	272	2387	294	261	2612
DT	Up	4527	0	0	3431	1	36	3416	1	1	3169	1	70
	Down	60	4149	4	35	3102	64	24	3095	52	24	3039	93
	Same	356	331	136	291	231	2372	315	272	2387	301	250	2616
J48	Up	4527	0	0	3449	1	18	3415	1	2	3216	3	21
	Down	60	4148	5	22	3125	54	23	3095	53	17	3075	64
	Same	356	333	134	242	224	2428	324	281	2369	249	230	2688
RF	Up	4500	0	27	3434	2	32	3397	1	20	3202	2	36
	Down	57	4130	26	22	3096	83	23	3078	70	16	3047	93
	Same	351	330	142	242	225	2477	312	270	2392	251	224	2692

According to the confusion matrices for Decision 1 in Table 4, MAS, MR, ML and DT, then J48 and RF classify the original data in similar manner, however BN, NB and LR do not when compared with the others. For the NewDecision 1, MR and RF give the best classifications, then in order, J48, MAS, DT, ML and maybe LR classify the balanced data, but BN and NB do not. For the NewDecision 2, ML and equivalent to DT classify the balance data well, then MAS, MR, J48, RF and LR follow them, respectively. Again, BN and NB do not show better performance for the classification of RBC data under NewDecision 2. For the NewDecision 3, similar classification order is seen. RF is the best, then J48, equivalent to MR and DT, then ML and MAS classify the balanced data for RBC data. Classification with BN, NB and LR are not good compared to the others.

The performance measure of the nine classification algorithms comparisons of TD and RBC data with the four decisions is presented in Fig 14 and 15, respectively.

In Fig 14. the precisions are compared with TD data, highest precision values (over than 0.90) for Decision 1 are seen in order on DT, J48, MR, MAS and ML, whereas for NewDecision 1, J48, MR DT, then MAS, RF and ML; for NewDecision 2, MAS, J48, MR, DT, ML and RF; and for NewDecision 3, J48, MR, RF, DT, ML and MAS. When the area under the ROC curve are compared, the highest area according to the decisions are as follow; for Decision 1, in order, BN, MR, ML, LR and DT are closed to 0.81; for NewDecision 1, RF, J48, DT, MR, ML, MAS and LR are greater than 0.90; for NewDecision 2, MR, ML, DT and RF are greater than 0.90; for NewDecision 3, MR, ML, DT, RF, J48, MAS and LR are greater than 0.90. The comparison according to Kappa statistics is as follow: for Decision 1, MAS, ML, MR, DT, J48, RF and LR have Kappa statistics changing in the range of 0.81 to 0.83; for NewDecision 1, while MR, J48 and RF have the kappa statistic around 0.92, MAS, ML, LR and DT range from 0.80 to 0.90; for NewDecision 2, MAS, MR, ML, DT, J48 and FR have the kappa statistic very close to 0.90; and for NewDecision 3, MR, RF and J48, have kappa statistic more than 0.90, however, the Kappa statistics for ML, MAS and DT are very close to 0.90.

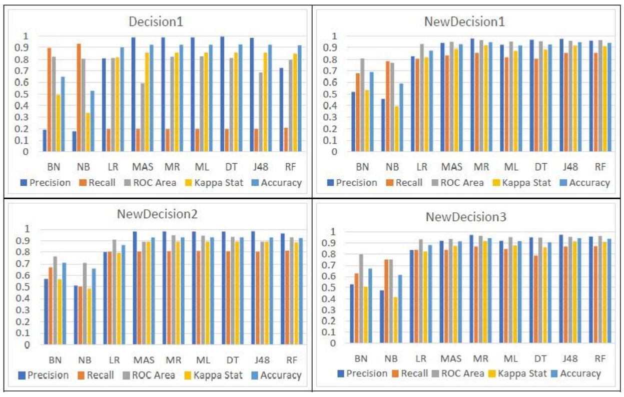


Fig 14. Nine algorithms' performance measure comparison of TD data according to four decisions

In Fig 15, the precisions are compared, highest precision values (over than 0.90) for Decision 1 are seen in order on MAS,MR, ML, DT and J48, whereas for NewDecision 1, MR, DT, J48, RF then ML and MAS; for NewDecision 2, similar to the TD data, MAS, J48, MR, DT, ML and RF; and for NewDecision 3, MR, J48, ML, DT and MAS. When comparing the area under the ROC curve, the highest area according to the decisions are as follow; for Decision 1, in order, BN, MR, ML and LR are closed to 0.80 but not greater than; for NewDecision 1, RF, J48, DT, MR, ML, MAS and LR are greater than 0.90; for NewDecision 2, MR, ML, DT and RF are greater than 0.90 and LR and J48 are close to 0.90; for NewDecision 3, MR, ML, RF, LR, J48, DT and MAS are greater than 0.90.

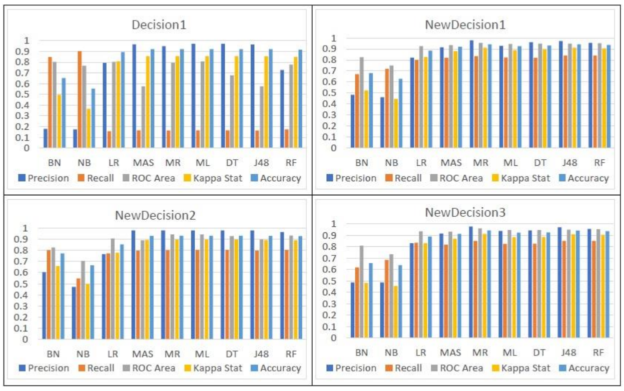


Fig 15. Nine algorithms' performance measure comparison of RBC data according to four decisions

The comparison according to Kappa statistics are as follow: for Decision 1, similar to the TD Kappa statistic values, MAS, ML, MR, DT, J48, RF and LR have kappa statistics changing in the range of 0.81 to-0.83; for NewDecision 1, only MR, RF and J48 have the value for Kappa statistic over than 0.90, but the others DT, MAS and ML have it around 0.90; for NewDecision 2, MAS, MR, ML, DT, J48 and RF have the Kappa statistic very close to 0.90; and for NewDecision 3, the Kappa statistic of J48, RF and Mr are greater than 0.90, whereas MAS, ML and DT have Kappa statistic very close to 0.90.

6 Conclusion and further Studies

Classification is the most commonly used DM method to group the data into different classes with the same properties. However, if the classes are classified with opposite number of instances in each class which causes imbalance class problems where the importance of majority classes generally overwhelm the importance of minority classes. To obtain a balance class there are various methods proposed in the literature. In this paper, we discussed three resampling methods with different point of views and showed how the performances of the classification algorithms are affected by our proposed resampling methods. The analysis was conducted with the stock price changes data of two Canadian banks over thirty-seven years which is categorized into three imbalance classes. The performances of nine classification algorithms were tested based on the performance measures.

In the first resampling method called the undersampling, works well, and can be used for imbalance problems. MR, ML, LR, J48, and then MAS, RF show the similar best performances of the area under the ROC curve for both data sets while BN and NB show the lowest area values. The best precision value is found in LR, but for NB and BN have very low precision values when compared with the others.

In the second resampling method called oversampling, where the instances in the minority class were replicated from 1 to 20 times and the influences of the performance measures of nine classification algorithms were tested. According to the precision metric, the algorithms whose precision values decrease until the fifth or sixth replications then increase are negatively affected algorithms whereas the algorithms whose precision values decrease until the first or second replications but increase later are less affected algorithms. The positively affected performances of classification algorithms are BN, NB and DT, all increases throughout the replications, whereas are LR, MAS, ML and DT are seriously negatively affected while MR and J48 are less affected. Among these nine classification algorithms, DT reflects the best precision change through the replications. According to the recall values for all classification algorithms, the MAS, ML and DT are seriously affected, they are not affected in the first two replications but then increase rapidly until the fifth or sixth replication, then increase smoothly obtaining the value in the range of (0.938,1) in the twentieth replication. Similarly, LR, MR and J48 show the same pattern, they start at very low recall value and slightly increase to fifth replication then smoothly increase towards a perfect value in the 20th replication. While the recall value remains the same for BN and NB up to the twentieth replications, recall value for RF reaches the perfect value after the first replication. According to the area under the ROC curve, MR, MAS and J48 have the highest areas through all replications whereas BN and NB have a same area. Since the area under the ROC curve reaches the perfect value at the second replication and remains the same. According to the Kappa statistic, RF achieves the best performance while BN and NB keep their smoothly increasing form but remains moderate in terms of the Kappa value. For the other classification algorithms, the Kappa statistic shows the similar performances. These results show that replication of the instances can be applied as an oversampling method.

In the third resampling method called hybrid method, the new definitions were proposed and tested. Both data sets show similar performance measures for the nine algorithms. In general speaking, six algorithms, MR, ML, MAS, DT, J48 and RF classified both data sets obtained by NewDecision 1, NewDecision 2 and NewDecision 3 definitions with high performance measures, however, LR, BN and NB did not classify the data sets as good as the others. This suggests that the definitions on the imbalance data set affect the classifications and their performance measures. This depends on the structure of the definitions. A new direction of research to explore the different decisions using different class range of definitions to define a new hybrid resampling method for an imbalance data set.

Although the three resampling approaches used in this work are worked well to solve the imbalance class problems, the hybrid method could be better than the others, because it is more flexible in defining a set of different definitions without changing the features of the classes.

The present work has shown some interesting conclusions and lead the researchers to work on different definitions on the resampling methods for imbalanced data sets, such as:

1. The proposed undersampling method improved the performance levels effectively, and they can be used to assess the imbalance level and classification of the new data set.

2- The proposed hybrid resampling method also improved the performance levels with different definitions on the range of minority class.

3- For the proposed hybrid method, new definitions can be proposed according to the data that allow the change in the range of definitions of classes.

4-The oversampling method also worked well on the multiclass imbalance data. When the performance measures of classification algorithms are tested the best classification algorithm can be found according to the replications that solve the imbalance problems.

For further studies, different decisions for hybrid resampling methods can be defined for an imbalance data set as an alternative to the methods in the literature and the effects of them on the performance measures of classification algorithms can be evaluated.

References:

[1] He, H., Garcia, E.: Learning from imbalanced data. IEEE Transactions on Knowledge and Data Engineering Vol. 21, No. 9, 2009, pp. 1263–1284.

[2] Fawcett, T., Provost, F., Adaptive fraud detection, *Data Minining and Knowledge Discovery*, Vol. 1, No. 3, 1997, pp. 291–316.

[3] García, V., Sánchez, J. S., Mollineda, R. A., Exploring the Performance of Resampling Strategies for the Class Imbalance Problem, *International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems*, *IEA/AIE 2010: Trends in Applied Intelligent Systems* , pp 541-549.

[4] Cieslak, D.A., Chawla, N.V.' Striegel, A., Combating imbalance in network intrusion datasets, *Proceedings of 2006 IEEE International Conference on Granular Computing*, 2006, pp.732-737.

[5] Thomas, C., Improving intrusion detection for imbalanced network traffic, *Security Communication Network,* 2012, pp. 1–17

[6] Perols, J. Financial statement fraud detection: an analysis of statistical and machine learning algorithms, *AUDITING: Journal of Prac. Theory,* Vol. 30, No 2, 2011, pp.19-50

[7] Mena, L., Gonzalez, J.A., Machine learning for imbalanced datasets: application in medical diagnostic, *Proceedings of the 19th International FLAIRS Conference (FLAIRS-2006),* Melbourne Beach, Florida, May 11–13, 2006.

[8] Zarinabad, N., Wilson, M., Gill, S.K., Manias, K.A., Davies , N.P., Peet, A.C., Multiclass Imbalance Learning: Improving Classification of Pediatric Brain Tumors from Magnetic Resonance Spectroscopy, *Magnetic Resonance in Medicine, Vol.* 77, 2017, pp. 2114–2124

[9] Li, Y.L., Sun, G.S., Zhu, Y., Data imbalance problem in text classifications, *The third International Symposium in Information Processing,* 2010, pp. 301-305.

[10] Huang, Y.M., Hung, C.M., Jiau, H., Evaluation of neural networks and data mining methods on a credit assesssment task for class imbalance problem, *Nonlinear Analysis: Real World Applications*, Vol.7, No.4, 2006, pp. 720–757

[11] Liu, D.Y., Feature selection based on mutual information for gear faulty diagnosis on imbalanced dataset, *Journal of Computer Information System,* Vol.8, No.18, 2012, pp.7831-7838.

[12] Boyle, T., *Dealing with imbalanced data: a guide to effectively handling imbalanced datasets in Python*, 2018.

[13] Anwar, M.N., Complexity Measurement for Dealing with Class Imbalance Problems in Classification Modelling, Massey University, *Institute of Fundamental Sciences*, 2012, Thesis for Doctor of Philosophy .

[14] Qian, Y., Liang, Y., Feng, G., Shi, X., A resampling ensemble algorithm for classification of imbalance problems, *Neurocomputing, Vol.* 143, 2014, pp. 57–67

[15] Japkowicz, N., Stephen, S., The class imbalance problem: A systematic study, *Intelligent Data Analysis*, Vol. 6, No.5, 2002, pp. 429–449.

[16] Thabtah, F., Hammoud, S., Kamalov, F., Gonsalves, A., Data imbalance in classification: Experimental evaluation, *Information Sciences,* Vol. 513, 2020, pp. 429–441.

[17] Galar, M., Fernandez, A., Barrenechea, E., Bustince, H., Herrera, F., A review on ensembles for the class imbalance problem: bagging-, boosting-, and hybrid- based approaches, *IEEE Trans. Syst., Man Cybern.-Part C: Appl. Rev.* Vol. 42, No. 4, 2012, pp. 463–484.

[18] Chawla , N.V., *Data mining for imbalanced datasets: an overview*, in: O. Maimon, L. Rokach (Eds.), Data Mining and Knowledge Discovery Handbook, Springer, Boston, 2005, pp. 853–867 .

[19] Kaur, P., Gosain, A., Issues and challenges of class imbalance problem in classification, *International Journal of Technology*, 2018, pp.1-13.

[20] Rahman, M.M., Davis, D., Cluster based under-sampling for unbalanced cardiovascular data, *Proceeding World Congress of Engineering, Vol.* 3, 2013, pp. 3–5.

[21] Nakamura, M., Kajiwara, Y., Otsuka, A., Kimura, H., Lvq- smote–learning vector quantization based synthetic minority over–sampling technique for biomedical data. *BioData Min* Vol. 6, No. 1, 2013, pp. 16

[22] Sa´ez, .JA., Luengo, J., Stefanowski, J., Herrera, F., Managing borderline and noisy examples in imbalanced classification by combining SMOTE with ensemble filtering, *International Conference on Intelligent Data Engineering and Automated Learning*, 2014, pp. 61–68.

[23] Al-Rifaie, M.M., Alhakbani, H.A., Handling class imbalance in direct marketing dataset using a hybrid data and algorithmic level solutions. *SAI Comput Conf (SAI)* , 2016, pp. 446–451.

[24] Dai, H.L., Class imbalance learning via a fuzzy total margin based support vector machine, *Appl Soft Comput, Vol.* 31, 2015, pp. 172–184.

[25] Tomar, D., Agarwal, S., Prediction of defective software modules using class imbalance learning. *Appl Comput Intell Soft Comput*, Vol. 6, 2016.

[26] Wasikowski, M., Chen, X.W., Combating the small sample class imbalance problem using feature selection. *IEEE Trans Knowl Data Eng*, Vol. 22, No. 10, 2010, pp. 1388–1400.

[27] Salunkhe, U.R., Mali, S.N., Classifier ensemble design for imbalanced data classification: a

hybrid approach. *Proc Comput Sci, Vol.* 85, 2016, pp. 725–732.

[28] Buda, M., A systematic study of the class imbalance problem in convolutional neural networks. *KTH Royal Institute of Technology, School of Computer Science and Communication*, Sweden, 2017.

[29] Ding, M., Yang, Y., Lan, Z., Multi-label imbalanced classification based on assessments of cost and value**,** *Applied intelligence*, Vol. 48, 2018, pp. 3577–3590.

[30] Devi D., Biswas, S., Purkayastha, B., Redundancy-driven modified Tomek-link based undersampling: a solution to class imbalance. *Pattern Recogn Lett* Vol. 93, 2017, pp. 3–12.

[31] Charte, F., Rivera, A., del Jesus, M.J., Herrera, F. *A., First approach to deal with imbalance in multi-label datasets*. Springer, Berlin, 2013, pp 150–160.

[32] Laurikkala, J., Improving identification of difficult small classes by balancing class distribution, *Conference on Artificial Intelligence in Medicine in Europe*, Springer, Portugal, 2001, pp. 63–66 .

[33] Jiang, K., Lu, J., Xia, K., A Novel Algorithm for Imbalance Data Classification Based on Genetic Algorithm Improved SMOTE, *Arab J Sci Eng,* Vol. 41, 2016, pp. 3255–3266.

[34] Triguero, I., del Río, S., López, V., Bacardit, J., Benítez, J.M., Herrera, F., ROSEFW-RF: The winner algorithm for the ECBDL’14 big data competition: An extremely imbalanced big data bioinformatics problem, *Knowledge-Based Systems*, Vol. 87, 2015, pp.69–79.

[35] Estabrooks, A., Jo, T., Japkowicz , N., A multiple resampling method for learning from imbalanced data sets, *Comput. Intell.* Vol. 20, No. 1, 2004, pp. 18–36 .

[36] Chen, Q. B. L.L. J. X., Xu, R., *Wemote - word embedding based minority oversampling technique for imbalanced emotion and sentiment classification,* 2013.

[37] Padurariu, C., Breaban, M.E., Dealing with Data Imbalance in Text Classification, *23rd International Conference on Knowledge-Based and Intelligent Information & Engineering Systems*, *Procedia Computer Science*, Vol. 159, 2019, pp. 736–745.

[38] Han, H., Wang, W., Mao, B., Borderline-SMOTE: a new oversampling method in imbalance data set learning. *Proceedings of International Conference on Intelligent Computing. Springer*, Berlin Heidelberg, 2005, pp. 878–887.

[39] Ramentol, E., Caballero, Y., Bello, R., SMOTE-RSB: a hybrid preprocessing approach based on oversampling and undersampling for high imbalanced data-sets using SMOTE and rough sets theory, *Knowl Inf Syst,* Springer-Verlag London Limited, 2011.

[40] Batista G., Prati R.C., Monard, M.C ., A study of the behaviour of several methods for balancing machine learning training data. *SIGKDD Explor,* Vol. 6, No. 1, 2004, pp.20–29.

[41] Bunkhumpornpat, C., Sinapiromsaran, K., Lursinsap, C., Safe-Level-SMOTE: safe-level-synthetic minority over-sampling technique for handling the class imbalanced problem, *Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD09). LNCS 3644*. Springer, 2009, pp. 475–482.

[42] Ruzgar, N.S., Comparative Analysis of Classification Algorithms on Stock Market Price Changes, *WSEAS TRANSACTIONS on INFORMATION SCIENCE and APPLICATIONS*, Vol 16, 2019, E-ISSN: 2224-3402, pp. 174-184

[43] Fan, W., Bifet, A., Mining Big Data: Current Status, and Forecast to the Future, *SIGKDD Explorations*, Vol. 14, No. 2, 2012.

[44] Ruzgar, N.S., Comparison of Classification Algorithms on Financial data, *WSEAS Transactions on Computers*, Volume 18, 2019, pp. 256-263.

[45] Kasperczul, A., Dardzinska, A., Comparative Evaluation of the different data Mining Techniques used for the Medical Database, *acta mechanica et automatica*, Vol.10, No. 3, 2016, DOI 10.1515/ama-2016-0036

[46] Laurier, C., Meyers, O., Serra, J., Blech, M., Herrera, P., Serra, X., Indexing music by mood: design and integration of an automatic content based annotator. *Multimedia Tools Applications*, Vol. 48, 2010, pp. 161–184.

[47] machinelearningmastery.com

[48] He, H., Ma, Y., *Imbalance learning: foundations, algorithms, and applications*, Wiley, 2013.

[49] McHugh, M. L., Interrater reliability: the kappa statistic, *Biochem Med, Zagreb*, Oct; Vol. 22, No. 3, 2012, pp. 276–282.

[50] Cohen, W.W., Fast effective rule induction, *Proceedings of the 12th International Conference on Machine Learning,* 1995, pp. 115–123.

[51] Hemlata, Comprehensive Analysis of Data Mining Classifiers Using Weka, *International Journal of Advanced Research in Computer Science* (0976-5697), Vol. 9, No. 2, 2018, pp. 718-723.

[52] Ivasic-Kos, M., Ipsic, I., Ribaric, S., Multi-level Image Annotation Using Bayes Classifier and Fuzzy Knowledge Representation Scheme, *WSEAS TRANSACTIONS on COMPUTERS*, E-ISSN: 2224-2872, Vol. 13, 2014, pp. 635-644.

[53] Hussain, N. I., Choudhury, B., Rakshit, S., A Novel Method for Preserving Privacy in Big-Data Mining, *International Journal of Computer Applications*, (0975-8887) Vol. 103, No.16, October 2014.

[54] Zainudin, S.M.N., Sulaiman M.N., Mustapha, N., Mohamed, R., Comparison of Expectation Maximization and K-means Clustering Algorithms with Ensemble Classifier Model, WSEAS TRANSACTIONS on COMPUTERS, E-ISSN: 2224-2872, Vol 17, 2018, pp. 253-259.

[55] Devi, T. S., Sundaram, K. M., A Comparative Analysis of Meta and Tree Classification Algorithms Using Weka, *International Research Journal of Engineering and Technology(IRJET)*, www.irjet.net, Vol.3, No.11, 2016, pp. 77-83.

[56] http://clouddc.chass.utoronto.ca.ezproxy.lib.ryerson.ca/ds/cfmrc/displayTSX.do?ed=2018&t=ts&f=daily &lang=en#v2, Accessed: May 4, 2019.