International Journal of Advances in Engineering and Management (IJAEM)
Volume 3, Issue 5 May 2021, pp: 653-672 www.ijaem.net ISSN: 2395-5252

Using Ensemble Learning To Fix Imbalanced Data Set

Adesina-Adebayo Fatimat Opeyemi
Physics with Electronics, The Polytechnic,Ibadan, Ibadan,

Submitted: 05-05-2021 Revised: 17-05-2021 Accepted: 20-05-2021

ABSTRACT: Presently knowledge of imbalanced data sets are very demanding for many data mining as well as machine learning application such as information retrieval, fraud detection, medical diagnosis etc. When data is imbalanced, that is when two classes doesn't have the same size of instance, one class is majority and the other class is minority. Many method have been developed to handle imbalance datasets case and one of most popular method of handle imbalanced data is sampling based method which is adopted for this research.

Ensemble learning techniques are model output for aggregating techniques to improved predictive classifier learning systems, therefore ensemble learning algorithms will construct the set of classifiers and classify a new classifier by voting for the prediction. The aim and objectives for this research is to present results obtained from Ensemble learning techniques to compare its performance of classification algorithm in terms of accuracy, sensitivity and specificity with focus on two-class problem. In addition thorough comparison will be made to show whether ensemble learning classifier makes a difference with sampling base method than ensemble learning with original data in terms of accuracy, high sensitivity and low specificity.
Meanwhile five method were choose from sampling based method to balance the dataset which are; Under-sample, Oversample, BOTH, ROSE and SMOTE and for ensemble learning classification boosting and bagging are considered using several machine learning algorithms like AdaBoost, XGBTree, TreeBag and Random Forest was considered in respective of ensemble learning. All the data used are collected from UCI machine learning.
Keywords: Imbalanced Data Set, Sample Based Method, AdaBoost, Random Forest,XGBTree

I. INTRODUCTION

An ensemble learning works when the disagreement occur between which models is best fit, it helps to improve machine learning results by combining several models for better production of predictive performance which is tend to work well compare to single model.

Ensemble learning are meta-algorithms that combine several machine learning techniques into one predictive model for decrease variance, bias or to improve prediction. The bagging is one that decrease variance, boosting bias and stacking improve our prediction.
The method for ensemble can be divided into two group:

- Sequential Ensemble method where the base learners are generated sequentially for example AdaBoost, Ada, etc. the basic purpose of this method is to exploit the dependence between the base learners, where the overall performance boosted.
- The second group is parallel ensemble method where base learners are generated in parallel for example Random forest, the purpose for this second method is to exploit independence between base learners.

Therefore, ensemble classifier are more effective than data balancing techniques to enhance the classification performance of imbalance data. This problem can easily approach by analyzing the data with some techniques to balance up the data and ensemble the classifier.

Imbalance Dataset

In field of machine learning, data is fundamental for the model's training and imbalance data sets is problem for both practical and research. Imbalance data is a highly potential problem in data mining and machine learning where class level is imbalance, which causes classification problem. In classification problem, a disparity in the frequencies of the observed classes can have a significant negative impact on model fitting. There are different technique of solving class imbalance, but One of the techniques will consider for this research which is Sampling BaseMethod. Sampling method are divide into various parts:

- Under-sampling or down sampling

International Journal of Advances in Engineering and Management (IJAEM)
Volume 3, Issue 5 May 2021, pp: 614-620 www.ijaem.net ISSN: 2395-5252

- Oversampling or up sampling, and
- Hybrid which are BOTH, SMOTE and ROSE

Chawla (2002) discuss the method to construct a classifier from imbalanced dataset. He combine the over-sampling (minority) and under-sampling (majority) to achieve better classifier performance using ROC space than use only Under-sampling that is majority class for achieving the better classifier performance.

❖ **Under-sampling or down sampling Method:** This method reduced the number of instances from the majority class so that it will balanced up with that of minority class. This enable the minority class have the same number of instances as majority class, the disadvantages of this method is that most times is removes the most important samples when trying to balance up with minority class. The diagram below show the distribution of under-sampling method

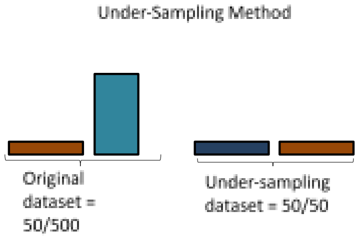

Figure 2.1: Under-sampling Method

❖ **Over-Sampling Method:** Oversample method can be define as adding instances to minority class for it to have the same number of instance with majority class. Advantages of this method is that using this method can lead to no information loss and disadvantage are it replicate observations in original data set which is leading to overfitting. Although the accuracy for training on such data will be high but accuracy for unseen data will worse.

Below are the outcome oversampling which show that class 1 is now increase

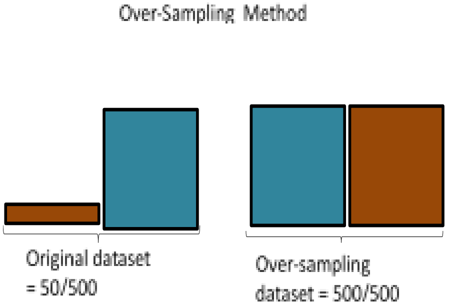

Figure 2.2: Over-Sampling Method

When compare the first diagram above, we can see that class 1 has the same number of instances with class 2when aplying oversampling method to the training dataset.

❖ **BOTH SAMPLING:** This method is the combination of over-sampling and under-sampling together, this method is that the majority class is under-sample without adding or replace it and minority class is favoured by replace, for the data use in this research it might be different case in other data set.

International Journal of Advances in Engineering and Management (IJAEM)
Volume 3, Issue 5 May 2021, pp: 653-672 www.ijaem.net ISSN: 2395-5252

Figure 2.3: BOTH-Sampling Method

❖ **RANDOM OVER-SAMPLING EXAMPLE (ROSE) Sampling:** This method provides a solution to effects of an imbalanced distribution of classes both generates data synthetically and provide a better accuracy on predictive classifier. It gives more values absolute impossible and draw artificial samples from minority class

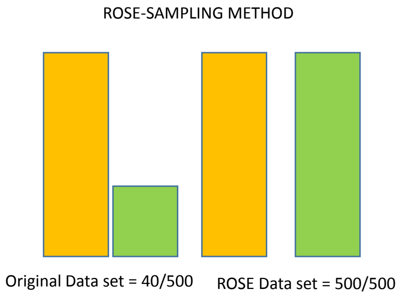

Figure 2.3: ROSE-SAMPLING METHOD

❖ **Synthetic Minority Over-Sampling Technique Example (Smote) Sampling:** This method also consider over-sampling approach where minority class is over-sampled by creating synthetic data example rather than by over-sampling with replacement. SMOTE generate equal number of synthetic class for minority class.

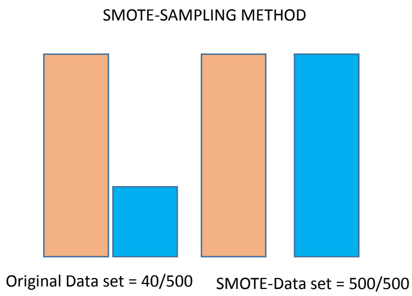

Figure 2.3: SMOTE-SAMPLING METHOD

Data Set Analysis

The data set used for this research are from UCI Machine Learning Repository, is a free domain, the data was prepare by checking the percentages of the classes , renaming the attributes, missing value was checked and treated using most frequently for valued numeric variables. The data was test with different algorithm on R machine learning and it summarize in table 1.

International Journal of Advances in Engineering and Management (IJAEM)
Volume 3, Issue 5 May 2021, pp: 614-620 www.ijaem.net ISSN: 2395-5252

Table 1: Data Set Distribution

Data Name	Majority Class	Minority Class	Number Of Attribute	Class Name
Breast Cancer	218	65	10	Recurrence And Non-Recurrence
Bank Marketing	4000	521	9	0 And 1
Htrus	16259	1639	17	No and Yes
Fertility	88	12	10	N And O

Data Pre-Processing Phase

The data pre-processing includes re-sampling of the dataset. This research considered Sampling based method, because is a widely used method to convert an imbalanced data to balanced data using some structures. The conversion develop by modify the number of instances of original data and provide the same number of instance to balance each class. Different techniques was adopted from Sample Base Method to modify the data as to original data. The sampling based method can be categories into these group into Under-sampling, Oversample, BOTH, ROSE and SMOTE e.t.c.

Classification Phase

There are a lot of classification algorithm that were utilized to predict class given a set futures. In this research multiple classifier that is ensemble learning were considered to predict accuracy, sensitivity and specificity of all the dataset used. Furthermore single classification was also used to do the comparison with multiple classification (Ensemble Learner)

Single Classification

Classification can be define as a supervised learning approach in machine learning which learn from data input to predict or classify new observation from the given data set. The data may be balance or imbalance, noisy or have multiple classes.Classification can also define as learn to classify unclassified data by decide whether to play when weather is windy or not to play when weather is hot. We have classification modelling or algorithm to predict classifier includes Logistic Regression (LR), Support Vector Machines (SVM), K-Nearest Neighbour (KNN) and C.45 so on.

Ensemble Learning Classification

The main objective of this ensemble learning is to increase the performance of single classifiers. Ensemble learning is an overall average classifier of balancing or imbalance dataset it mostly occur when there is disagreement between which of the model is best to fit. Boosting and bagging are most widely used ensemble learning techniques because the applications in classification problems led to meaningful improvement. The diagram below shows the difference between bagging and boosting ensemble learning classification.

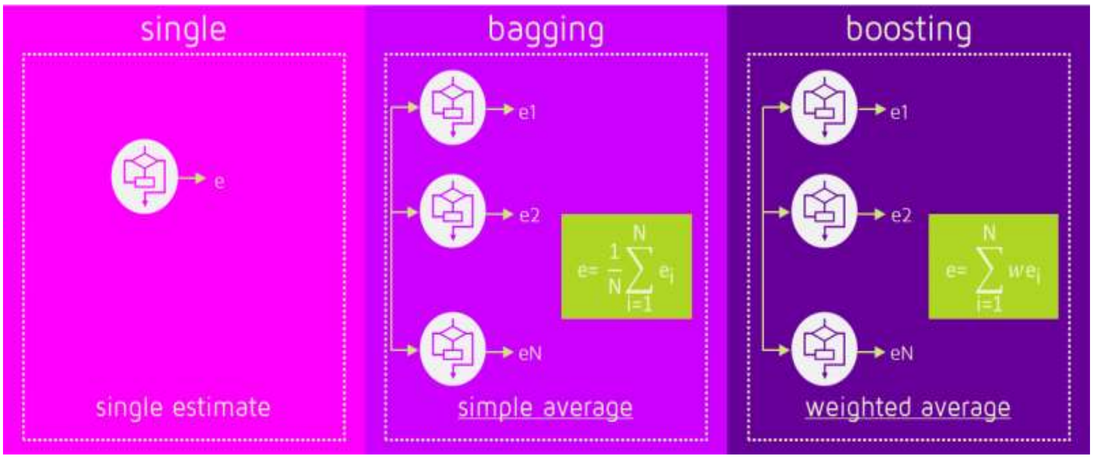

Figure 2.6: Differences between Single, Bagging and Boosting (accessed online:https://quantdare.com/what-is-the-difference-between-bagging-and-boosting/ on 16th Jan 2020)

International Journal of Advances in Engineering and Management (IJAEM)
Volume 3, Issue 5 May 2021, pp: 653-672 www.ijaem.net ISSN: 2395-5252

Comparison of Classification Phase

This phase will discuss about the improvement of different models used for this research, model sensitivity toward the negative (minority) class using different type of balancing techniques. Model performance will be evaluated using various measures such as Sensitivity, Specificity and Accuracy.

- SENSITIVITY: This mean how often we can predict minority class correctly
- SPECIFICITY: This mean how often we can predict majority class correctly
- Accuracy: this is overall how often the classifier is correct

Below are the definition of measures use:

i. $\text{Sensitivity: TP} = \frac{\text{TP}}{\text{TP}+\text{FN}}$

ii. $\text{Specificity: TP} = \frac{\text{TN}}{\text{FP}+\text{FP}}$

iii. $\text{Accuracy: TP} + \text{TN}/\text{OVERALL}$

Where;

$$\text{TP} = \text{TRUE POSITIVE}$$
$$\text{FN} = \text{FALSE NEGATIVE}$$
$$\text{TN} = \text{TRUE NEGATIVE}$$
$$\text{FP} = \text{FALSE POSITIVE}$$

And the comparison between different types of sample base method with ensemble classifier based on sensitivity, specificity and accuracy to compare them with original data set.

Experimental Design

This section is applied the earlier discussed techniques to improve the classifier predictive ability of minority class using four different types of imbalanced data sets from UCI machine learning. Also to access which of the Sampling Based Method works better to balance imbalance dataset, and to investigate improvement of predictive accuracy, sensitivity and specificity with respect to Ensemble learning.

In designing the experiment, we are compare the performance of the various models, the following processes were considered:

- Problem definition
- Design of Test
- Model Testing
- Final model selection

Therefore, the experiment will carried out in Rstudio, though Python can also be used for this project but Rstudio was selected because it has robust packages imbalanced imputation. Also R is developed by scientist and academician for statistical problem, machine learning and data science, R had many libraries packages and equipped with many packages to carry out time series analysis and data mining with all this there is no better tools compared to R language.

- **Problem Definition**

When looked at some imbalance data related to the datasets like unlabelled, missing value in selected data from UCI machine learning, statistical technique was performed to each of the dataset where it applicable.

- **Design of Test**

The design of test usually involves testing the different models on selected data from all the four datasets. This normally includes following;

- Splitting the dataset into training and testing set
- Balancing the data (Sampling Based Method)
- Fit a model on the training set
- Comparison of sample base method against original data
- Comparison of ensemble learning algorithm

MODEL TESTING

The following models were tested and compared for evaluation performance:

BALANCE METHOD

- Over-Sample
- Under-Sample
- BOTH
- ROSE
- SMOTE

SINGLE CLASSIFER

- K-Nearest Neighbour (KNN)
- SVM
- Linear Regression

Multiple Classifer (Ensemble Learner)

- AdaBoost
- XGBoost
- Random Forest
- TreeBag

- **Model Comparison**

After all model testing is completed, we selected the best Ensemble model based on the performance accuracy, sensitivity and specificity along with balance data and original data. This process is discussed in detail in the implementation in the next chapter.

All the data set was load, the data set was rename, cleaned and missing values was input were it appropriate.

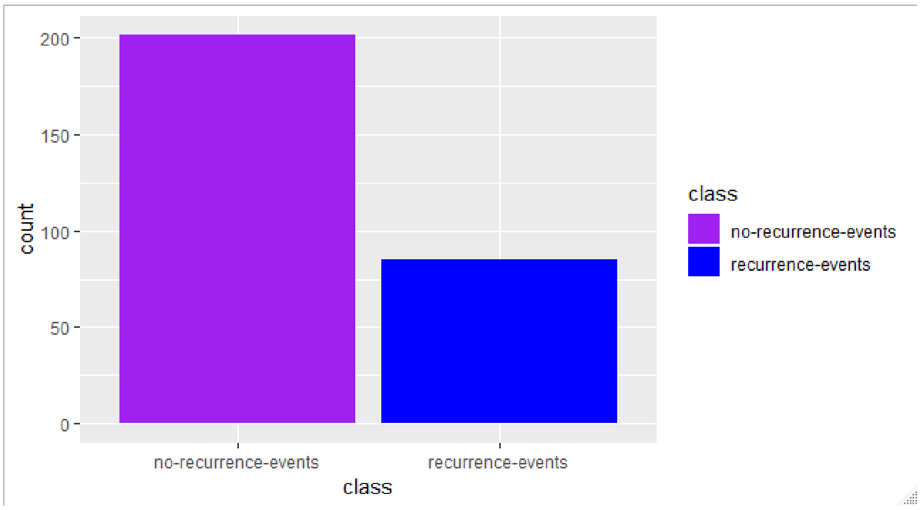

Figure 3.8: Class imbalance Problem

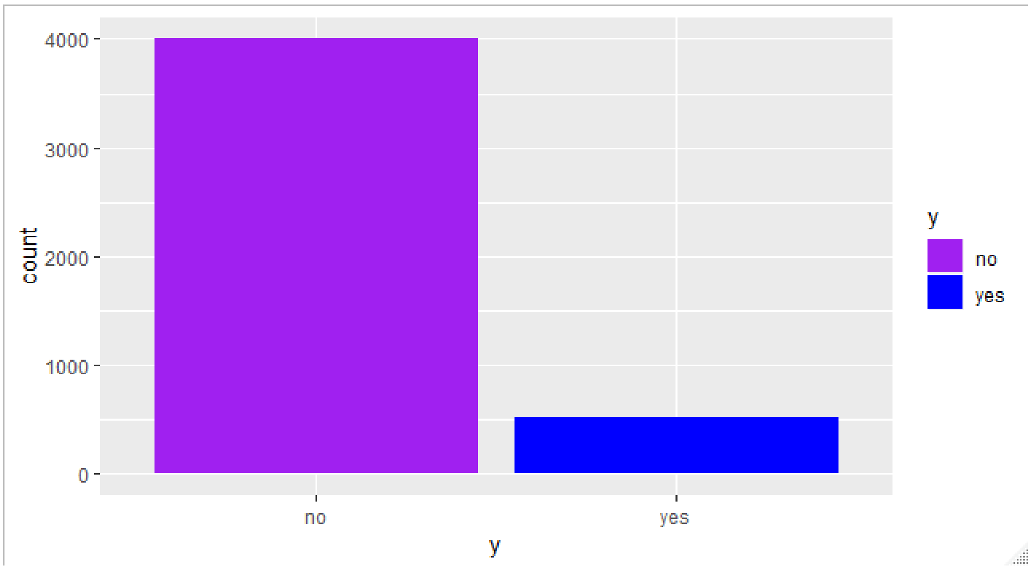

Figure 3.10: Class imbalance Problem Data 2

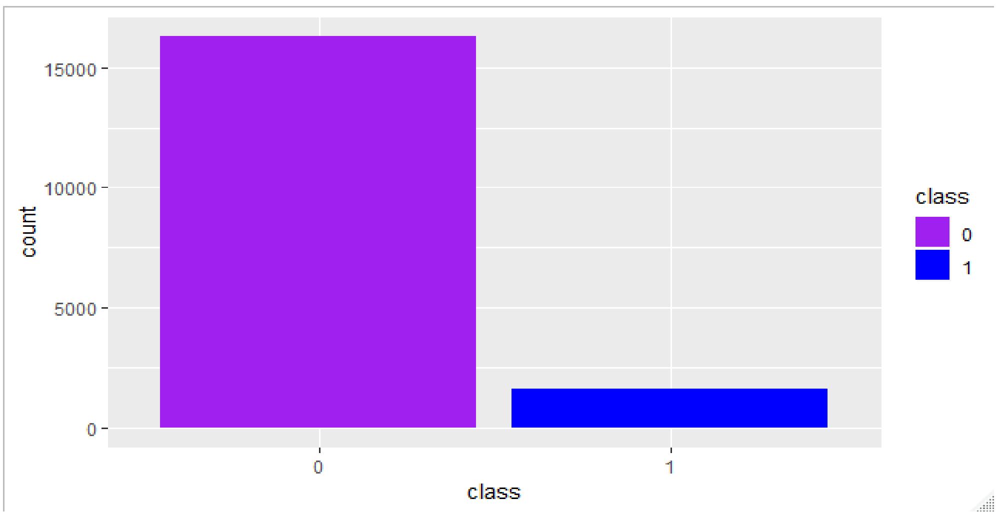

Figure 3.12: Class imbalance Problem Data 3

International Journal of Advances in Engineering and Management (IJAEM)
Volume 3, Issue 5 May 2021, pp: 653-672 www.ijaem.net ISSN: 2395-5252

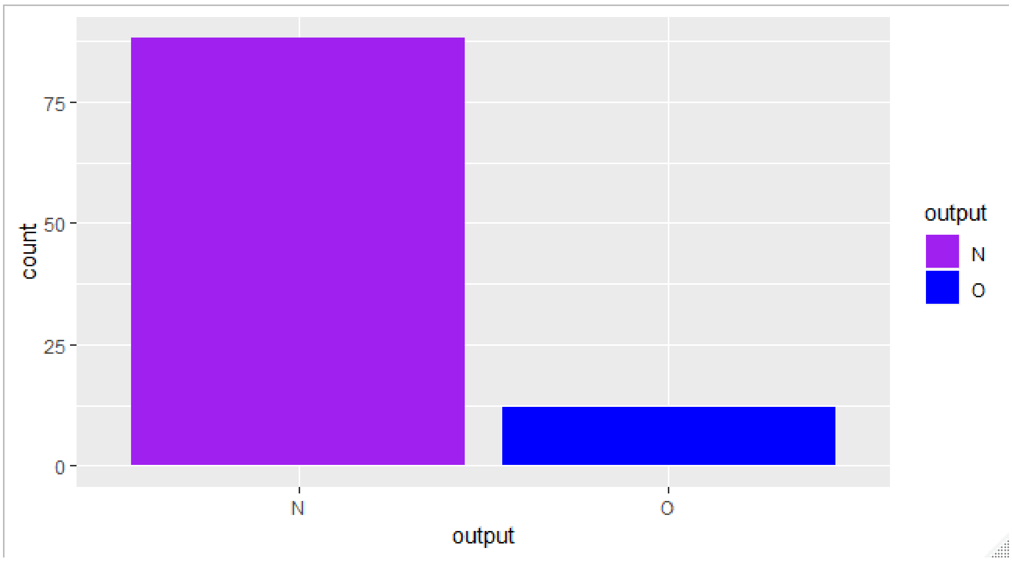

Figure 3.14: Class imbalance Problem

The summary result above shows that there is class imbalance problem in all the four data we want to use. However, we will go further to conduct the exploratory data analysis on the four datasets.

Exploratory Analysis

We will now explore our data to get a perception about the model that will be appropriate to use for ensemble learning to fix imbalance data set. Firstly,we will pre-process the data and carry out predictive ability as stated in above chapter. Therefore, in exploratory analysis the first thing to do is to training and validation of our data because of overfitting. We do this because we want our machine learning algorithm to learn something new from historical data to make prediction performance. Furthermore we create a model on training data set that is random sample and apply it on validation to see how well our model to fit our desire dependent variable is. Model accuracy will now tested on both training and validation, validation accuracy or performance is considered more realistic as training performance may reflect overfitting. Splitting of data into training and validation can be 50:50, 60:40, 70:30, 80:20 e.t.c. depending on availability of data and computational power. In this research we are using 70:30 for training and testing of all the four data use. The diagram below show the process of splitting our data and test it for predictive performance.

Figure 3.15: Splitting of Bank Dataset into train and test.

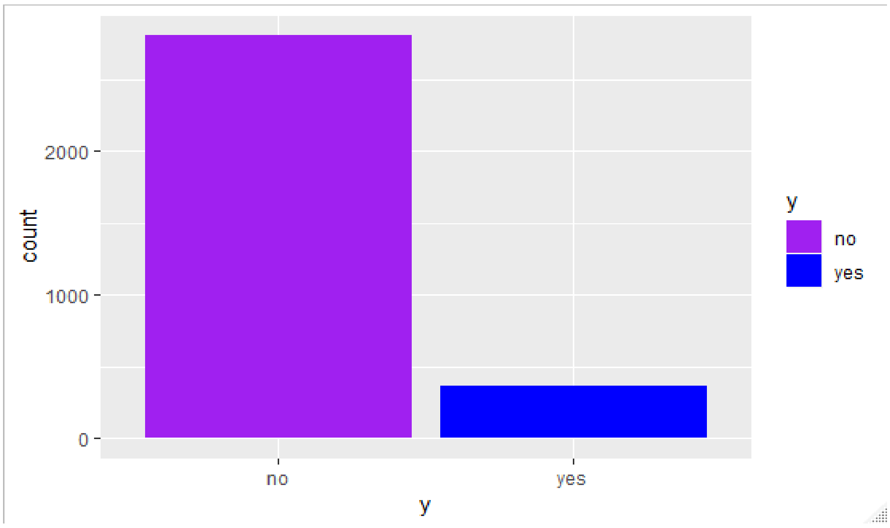

Figure 3.18: Class imbalance Problem for Train graph

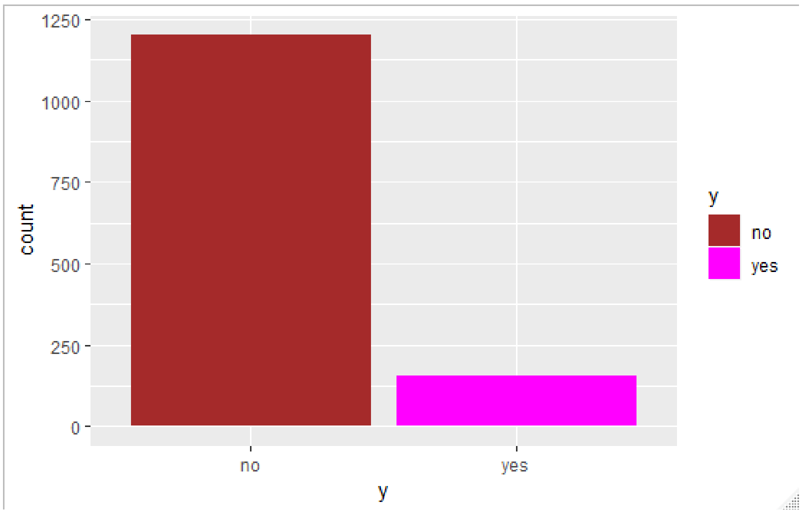

Figure 3.19: Class Imbalance Problem for Test graph

Balancing Exploratory Analysis

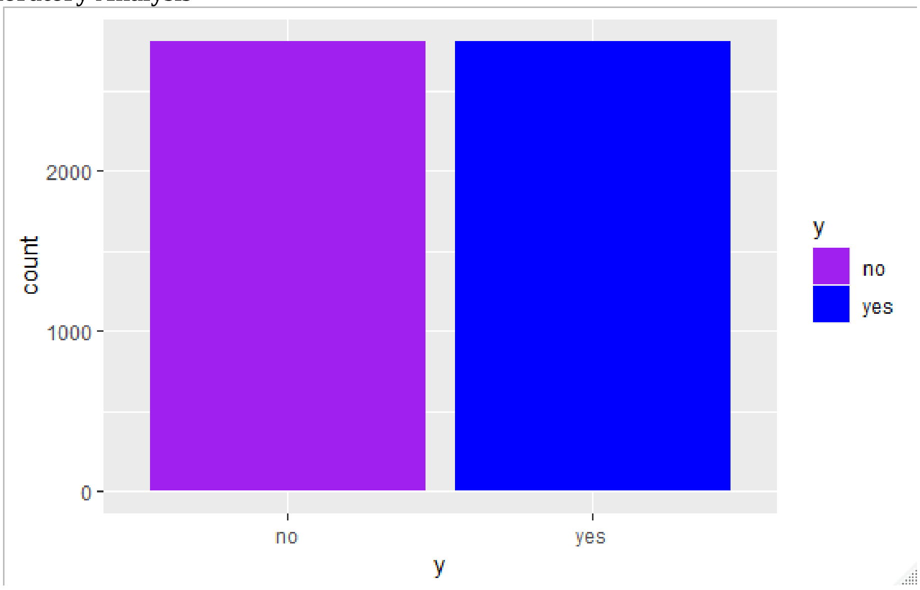

Figure: 3.23: Over-sample Graph Performance.

International Journal of Advances in Engineering and Management (IJAEM)
Volume 3, Issue 5 May 2021, pp: 653-672 www.ijaem.net ISSN: 2395-5252

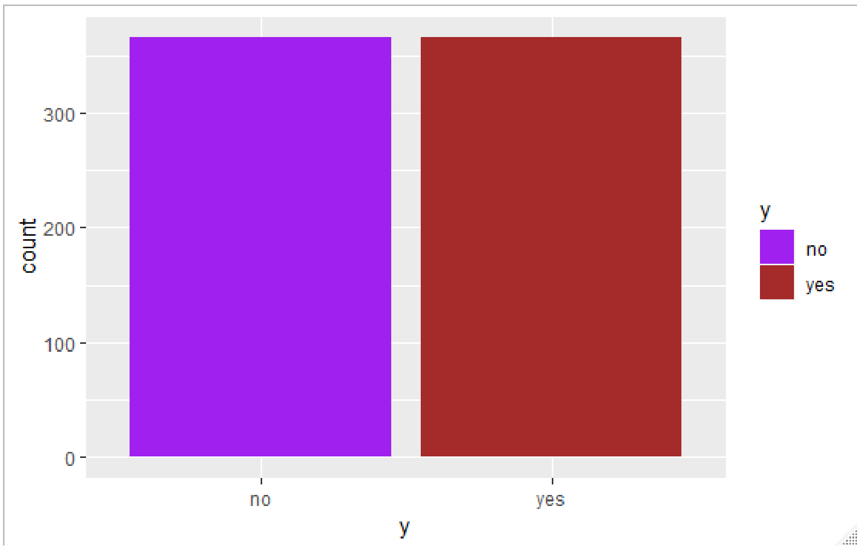

Figure: 3.27: Under-sample Graph Performance

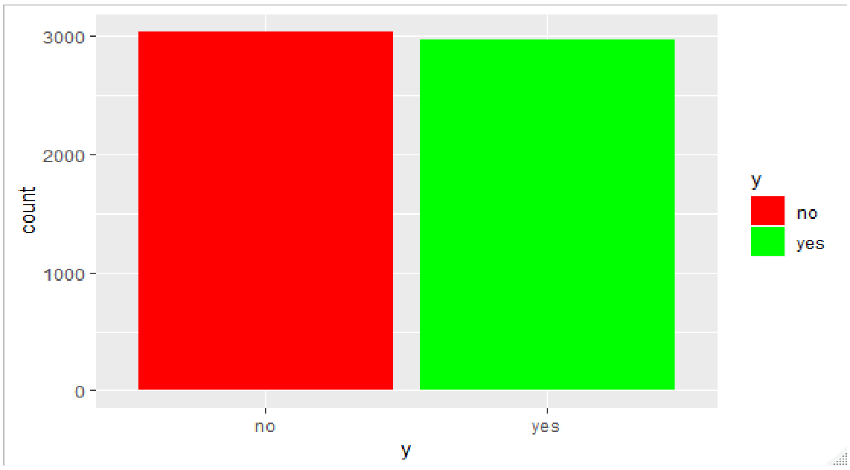

Figure 3.31: BOTH Graph Performance

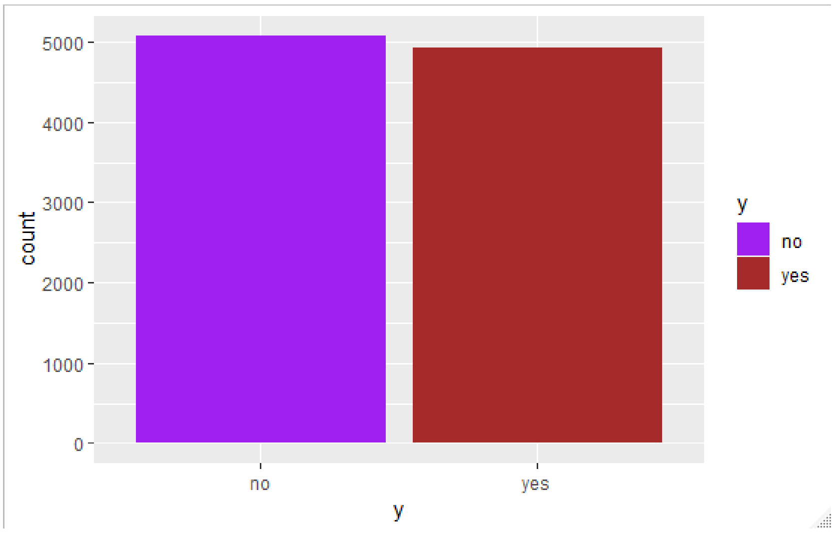

Figure 3.35: ROSE Graph Performance

International Journal of Advances in Engineering and Management (IJAEM)
Volume 3, Issue 5 May 2021, pp: 614-620 www.ijaem.net ISSN: 2395-5252

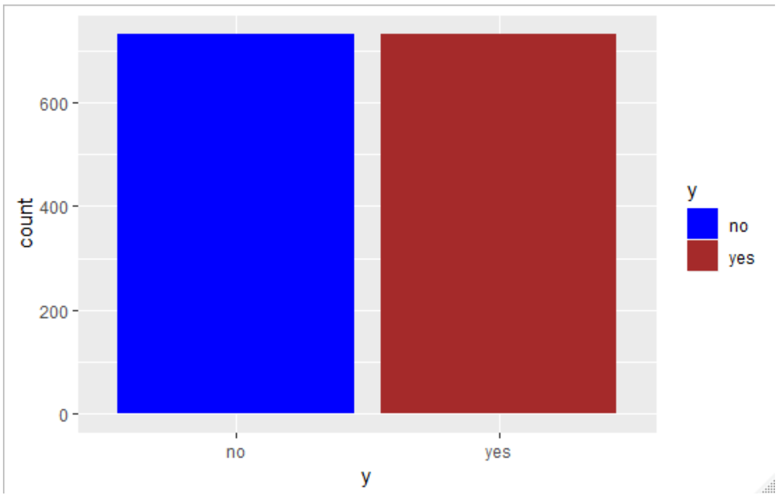

Figure 3.39: SMOTE Graph Performance

II. COMPARISON BETWEEN ENSEMBLES LEARNING WITH SAMPLE BASE METHOD AND ORIGINAL DATA SET

In the previous chapter ensemble learning algorithm was discuss and four model was built from ensemble learning algorithm to do justice with data set used after the data set was balanced with five different techniques to balancethe data.

Adaboost, XGBtree, TreeBag and Random Forest were built from two methods adopted from ensemble learning for this project i.e. Bagging and Boosting Ensemble learning.

Below are the table for each data use for sample base method with ensemble learning.

Data Set 1

Table 4.1: Over-sample method/Original with Ensemble Learning

Model Name	Accuracy Original	Specificity Original	Sensitivity Original	Accuracy Over	Specificity Over	Sensitivity Over	% Improvement Accuracy	%Improvement Specificity	%Improve ment Sensitivity
AdaBoost	0.7529	0.9000	**0.4000**	0.6941	0.8167	**0.4000**	0.0600	0.0800	**0.000**
XGBTree	0.7529	0.9167	**0.3600**	0.7176	0.8667	**0.3600**	0.0400	0.0500	**0.000**
TreeBag	0.7176	0.8500	**0.4000**	0.6588	0.7500	**0.4400**	0.0600	0.1000	**0.0400**
Random Forest	0.7059	0.800	**0.3200**	0.7059	0.8667	**0.4800**	0.0000	0.0700	**0.1600**

Table 4.2: Under-sample method/Original with Ensemble Learning

International Journal of Advances in Engineering and Management (IJAEM)
Volume 3, Issue 5 May 2021, pp: 653-672 www.ijaem.net ISSN: 2395-5252

Model Name	Accuracy Original	Specificity Original	Sensitivity Original	Accuracy Under	Specificity Under	Sensitivity Under	% Improvement Accuracy	%Improvement Specificity	%Improvement Sensitivity
AdaBoost	0.7529	0.9000	**0.4000**	0.6118	0.5833	**0.6800**	0.1411	0.3167	**0.2800**
XGBTree	0.7529	0.9167	**0.3600**	0.6000	0.5667	**0.6800**	0.1529	0.35	**0.3200**
TreeBag	0.7176	0.8500	**0.4000**	0.6235	0.5833	**0.7200**	0.0941	0.2667	**0.3200**
Random Forest	0.7059	0.800	**0.3200**	0.7647	0.8167	**0.6400**	0.0588	0.05	**0.3200**

Table 4.3: BOTH-sample method/Original with Ensemble Learning

Model Name	Accuracy Original	Specificity Original	Sensitivity Original	Accuracy Both	Specificity Both	Sensitivity Both	% Improvement Accuracy	%Improvement Specificity	%Improvement Sensitivity
AdaBoost	0.7529	0.9000	**0.4000**	0.6000	0.6833	**0.4000**	0.1529	0.2167	**0.000**
XGBTree	0.7529	0.9167	**0.3600**	0.5765	0.7000	**0.2800**	0.1764	0.2167	**0.2800**
TreeBag	0.7176	0.8500	**0.4000**	0.6588	0.5833	**0.5600**	0.0588	0.1500	**0.1600**
Random Forest	0.7059	0.800	**0.3200**	0.6588	0.7333	**0.7333**	0.0471	0.1334	**0.1600**

Table 4.4: ROSE-sample method/Original with Ensemble Learning

Model Name	Accuracy Original	Specificity Original	Sensitivity Original	Accuracy ROSE	Specificity ROSE	Sensitivity ROSE	% Improvement Accuracy	%Improvement Specificity	%Improvement Sensitivity
AdaBoost	0.7529	0.9000	**0.4000**	0.6941	0.8000	**0.4400**	0.0588	0.1000	**0.4000**

International Journal of Advances in Engineering and Management (IJAEM)
Volume 3, Issue 5 May 2021, pp: 614-620 www.ijaem.net ISSN: 2395-5252

XGBTree	0.7529	0.9167	**0.3600**	0.7176	0.8000	**0.5200**	0.0353	0.1167	**0.1600**
TreeBag	0.7176	0.8500	**0.4000**	0.6471	0.7333	**0.4400**	0.0705	0.1167	**0.4000**
Random Forest	0.7059	0.800	**0.3200**	0.6824	0.7833	**0.4400**	0.0235	0.0834	**0.1200**

Table 4.5: SMOTE-sample method/Original with Ensemble Learning

Model Name	Accuracy Original	Specificity Original	Sensitivity Original	Accuracy ROSE	Specificity ROSE	Sensitivity ROSE	% Improvement Accuracy	%Improvement Specificity	%Improvement Sensitivity
Ada-Boost	0.7529	0.9000	**0.4000**	0.5412	0.5000	**0.6400**	0.2117	0.4000	**0.2400**
XGBTree	0.7529	0.9167	**0.3600**	0.6471	0.6500	**0.6400**	0.1058	0.2667	**0.2800**
TreeBag	0.7176	0.8500	**0.4000**	0.6353	0.6167	**0.6800**	0.0823	0.2333	**0.2800**
Random Forest	0.7059	0.8667	**0.3200**	0.6235	0.6000	**0.6800**	0.0824	0.2667	**0.3600**

Data Set 2

Table 4.6: Over-sample method/Original with Ensemble Learning

Model Name	Accuracy Original	Specificity Original	Sensitivity Original	Accuracy Over	Specificity Over	Sensitivity Over	% Improvement Accuracy	%Improvement Specificity	%Improvement Sensitivity
Ada-Boost	0.9012	0.9792	**0.3462**	0.8923	0.9733	**0.3462**	0.0089	0.0059	**0.000**
XGBTree	0.9012	0.9750	**0.3333**	0.8864	0.9108	**0.6987**	0.0148	0.0642	**0.3654**
Tree-Bag	0.9071	0.9683	**0.4359**	0.88886	0.9383	**0.5064**	0.0185	0.03	**0.0705**

Random Forest	0.9122	0.9725	**0.4487**	0.8968	0.8950	**0.4808**	0.0154	0.0775	**0.0321**

Table 4.7: Under-sample method/Original with Ensemble Learning

Model Name	Accuracy Original	Specificity Original	Sensitivity Original	Accuracy Under	Specificity Under	Sensitivity Under	% Improvement Accuracy	%Improvement Specificity	%Improvement Sensitivity
AdaBoost	0.9012	0.9792	**0.3462**	0.8142	0.8142	**0.8141**	0.089	0.1650	**0.4679**
XGBTree	0.9012	0.9750	**0.3333**	0.8201	0.8142	**0.8654**	0.0811	0.0168	**0.5321**
TreeBag	0.9071	0.9683	**0.4359**	0.8053	0.7975	**0.8654**	0.1018	0.1708	**0.4295**
Random Forest	0.9122	0.9725	**0.4487**	0.8119	0.8008	**0.8974**	0.1003	0.1717	**0.4487**

Table 4.8: BOTH-sample method/Original with Ensemble Learning

Model Name	Accuracy Original	Specificity Original	Sensitivity Original	Accuracy Both	Specificity Both	Sensitivity Both	% Improvement Accuracy	%Improvement Specificity	%Improvement Sensitivity
AdaBoost	0.9012	0.9792	**0.3462**	0.9120	0.9733	**0.2244**	0.000	0.0060	**0.1218**
XGB Tree	0.9012	0.9750	**0.3333**	0.8768	0.8992	**0.7051**	0.0244	0.7580	**0.3718**
TreeBag	0.9071	0.9683	**0.4359**	0.8842	0.9200	**0.6090**	0.0229	0.0483	**0.1731**
Random Forest	0.9122	0.9725	**0.4487**	0.8768	0.9242	**0.5128**	0.0354	0.0483	**0.0641**

Table 4.9: ROSE-sample method/Original with Ensemble Learning

Model Name	Accuracy Original	Specificity Original	Sensitivity Original	Accuracy ROSE	Specificity ROSE	Sensitivity ROSE	% Improvement Accuracy	%Improvement Specificity	%Improvement Sensitivity
Ada-Boost	0.9012	0.9733	**0.2244**	0.8709	0.8925	**0.7051**	0.0303	0.0808	**0.4807**
XGBTree	0.9012	0.9750	**0.3333**	0.8798	0.9000	**0.7244**	0.0214	0.0750	**0.3911**
Tree-Bag	0.9071	0.9683	**0.4359**	0.8673	0.8825	**0.7500**	0.0398	0.0858	**0.3141**
Random Forest	0.9122	0.9725	**0.4487**	0.8739	0.8850	**0.7885**	0.0383	0.0875	**0.3398**

Table 4.10: SMOTE-sample method/Original with Ensemble Learning

Model Name	Accuracy Original	Specificity Original	Sensitivity Original	Accuracy ROSE	Specificity ROSE	Sensitivity ROSE	% Improvement Accuracy	%Improvement Specificity	%Improvement Sensitivity
Ada-Boost	0.9012	0.9733	**0.3462**	0.8378	0.8450	**0.7821**	0.0634	0.1283	**0.4359**
XGBTree	0.9012	0.9750	**0.3333**	0.8407	0.8458	**0.8013**	0.0605	0.1292	**0.4680**
Tree-Bag	0.9071	0.9683	**0.4359**	0.8673	0.8825	**0.7500**	0.0398	0.0858	**0.3141**
Random Forest	0.9122	0.9725	**0.4487**	0.8326	0.8358	**0.8077**	0.0796	0.1367	**0.3590**

Data Set 3

Table 4.11: Over-sample method/Original with Ensemble Learning

Model Name	Accuracy Original	Specificity Original	Sensitivity Original	Accuracy Over	Specificity Over	Sensitivity Over	% Improvement Accuracy	%Improvement Specificity	%Improvement Sensitivity
Ada-Boost	0.9812	0.9938	**0.8557**	0.8378	0.8450	**0.7821**	0.1434	0.1488	**0.0736**

International Journal of Advances in Engineering and Management (IJAEM)
Volume 3, Issue 5 May 2021, pp: 653-672 www.ijaem.net ISSN: 2395-5252

XGB Tree	0.9012	0.9750	**0.3333**	0.8407	0.8458	**0.8013**	0.0605	0.1292	**0.4680**
Tree-Bag	0.9071	0.9683	**0.4359**	0.8673	0.8825	**0.7500**	0.0398	0.0858	**0.3141**
Random Forest	0.9122	0.9725	**0.4487**	0.8326	0.8358	**0.8077**	0.0796	0.1367	**0.3590**

Table 4.12: Under-sample method/Original with Ensemble Learning

Model Name	Accuracy Original	Specificity Original	Sensitivity Original	Accuracy Under	Specificity Under	Sensitivity Under	% Improvement Accuracy	%Improvement Specificity	%Improvement Sensitivity
Ada-Boost	0.9812	0.9938	**0.8557**	0.9595	0.9660	**0.8949**	0.0217	0.0278	**0.0392**
XGBTree	0.9785	0.9941	**0.8240**	0.9689	0.9759	**0.8998**	0.0096	0.0182	**0.0758**
Tree-Bag	0.9794	0.9936	**0.8386**	0.9622	0.9695	**0.8900**	0.0172	0.0241	**0.0514**
Random Forest	0.9792	0.9943	**0.8289**	0.9662	0.9734	**0.8949**	0.0130	0.0209	**0.066**

Table 4.13: BOTH-sample method/Original with Ensemble Learning

Model Name	Accuracy Original	Specificity Original	Sensitivity Original	Accuracy Both	Specificity Both	Sensitivity Both	% Improvement Accuracy	%Improvement Specificity	%Improvement Sensitivity
Ada-Boost	0.9812	0.9938	**0.8557**	0.9783	0.9911	**0.8509**	0.0029	0.0027	**0.0048**
XGB Tree	0.9785	0.9941	**0.8240**	0.9729	0.8992	**0.8753**	0.0056	0.0949	**0.0513**
Tree-Bag	0.9794	0.9936	**0.8386**	0.9738	0.9850	**0.8631**	0.0056	0.0086	**0.0245**

International Journal of Advances in Engineering and Management (IJAEM)
Volume 3, Issue 5 May 2021, pp: 614-620 www.ijaem.net ISSN: 2395-5252

Random Forest	0.9792	0.9943	**0.8289**	0.9779	0.9887	**0.8411**	0.0013	0.0056	**0.0122**

Table 4.14: ROSE-sample method/Original with Ensemble Learning

Model Name	Accuracy Original	Specificity Original	Sensitivity Original	Accuracy ROSE	Specificity ROSE	Sensitivity ROSE	% Improvement Accuracy	%Improvement Specificity	%Improvement Sensitivity
AdaBoost	0.9812	0.9938	**0.8557**	0.9759	0.9904	**0.8313**	0.0053	0.0034	**0.0244**
XGB Tree	0.9785	0.9941	**0.8240**	0.9765	0.9906	**0.8362**	0.0002	0.0035	**0.0122**
TreeBag	0.9794	0.9936	**0.8386**	0.9738	0.9872	**0.8411**	0.0056	0.0064	**0.0025**
Random Forest	0.9792	0.9943	**0.8289**	0.9750	0.9887	**0.8411**	0.004	0.0056	**0.0122**

Table 4.15: SMOTE-sample method/Original with Ensemble Learning

Model Name	Accuracy Original	Specificity Original	Sensitivity Original	Accuracy ROSE	Specificity ROSE	Sensitivity ROSE	% Improvement Accuracy	%Improvement Specificity	%Improvement Sensitivity
AdaBoost	0.9812	0.9938	**0.8557**	0.9624	0.9707	**0.8802**	0.0188	0.0231	**0.0245**
XGB Tree	0.9785	0.9941	**0.8240**	0.9615	0.9673	**0.0946**	0.017	0.0268	**0.0806**
TreeBag	0.9794	0.9936	**0.8386**	0.9607	0.9673	**0.8949**	0.0187	0.0263	**0.0563**
Random Forest	0.9792	0.9943	**0.8289**	0.9671	0.9744	**0.8949**	0.0121	0.0199	**0.0660**

International Journal of Advances in Engineering and Management (IJAEM)
Volume 3, Issue 5 May 2021, pp: 653-672 www.ijaem.net ISSN: 2395-5252

Data Set 4

Table 4.16: Over-sample method/Original with Ensemble Learning

Model Name	Accuracy Original	Specificity Original	Sensitivity Original	Accuracy Over	Specificity Over	Sensitivity Over	% Improvement Accuracy	%Improvement Specificity	%Improvement Sensitivity
AdaBoost	0.8400	0.9545	**0.3333**	0.8800	0.9091	**0.3333**	0.0400	0.0454	**0.0000**
XGBTree	0.8400	0.9091	**0.3333**	0.8800	0.9545	**0.3333**	0.0400	0.0454	**0.0000**
TreeBag	0.8000	0.9091	**0.0000**	0.8400	0.9091	**0.3333**	0.0400	0.000	**0.3333**
Random Forest	0.8800	1.0000	**0.0000**	0.8800	0.8636	**0.3333**	0.0000	0.1364	**0.3333**

Table 4.17: Under-sample method/Original with Ensemble Learning

Model Name	Accuracy Original	Specificity Original	Sensitivity Original	Accuracy Under	Specificity Under	Sensitivity Under	% Improvement Accuracy	%Improvement Specificity	%Improvement Sensitivity
AdaBoost	0.8400	0.9545	**0.3333**	0.1200	0.0000	**1.0000**	0.7200	0.9545	**0.6667**
XGB Tree	0.8400	0.9091	**0.3333**	0.6400	0.5909	**1.0000**	0.2000	0.1364	**0.6667**
TreeBag	0.800	0.9091	**0.0000**	0.7200	0.7727	**0.3333**	0.0800	0.1364	**0.3333**
Random Forest	0.8800	1.000	**0.0000**	0.5600	0.5909	**0.3333**	0.3200	0.4091	**0.3333**

Table 4.18: BOTH-sample method/Original with Ensemble Learning

Model Name	Accuracy Original	Specificity Original	Sensitivity Original	Accuracy Both	Specificity Both	Sensitivity Both	% Improvement Accuracy	%Improvement Specificity	%Improvement Sensitivity
AdaBoost	0.9812	0.9938	**0.8557**	0.9783	0.9911	**0.8509**	0.0029	0.0027	**0.0048**

International Journal of Advances in Engineering and Management (IJAEM)
Volume 3, Issue 5 May 2021, pp: 614-620 www.ijaem.net ISSN: 2395-5252

XGBTree	0.8400	0.9545	**0.3333**	0.1200	0.000	**1.0000**	0.7200	0.9545	**0.6667**
Tree-Bag	0.8000	0.9091	**0.0000**	0.7200	0.7727	**0.3333**	0.0800	0.1364	**0.3333**
Random Forest	0.8800	1.000	**0.0000**	0.5600	0.5909	**0.3333**	0.3200	0.4091	**0.3333**

Table 4.19: ROSE-sample method/Original with Ensemble Learning

Model Name	Accuracy Original	Specificity Original	Sensitivity Original	Accuracy ROSE	Specificity ROSE	Sensitivity ROSE	% Improvement Accuracy	%Improvement Specificity	%Improvement Sensitivity
Ada-Boost	0.8400	0.9091	**0.3333**	0.8800	0.9545	**0.3333**	0.0400	0.0454	**0.000**
XGBTree	0.8400	0.9091	**0.3333**	0.8800	0.9545	**0.3333**	0.0400	0.0454	**0.000**
Tree-Bag	0.8000	0.9091	**0.0000**	0.8000	0.8636	**0.3333**	0.0000	0.0455	**0.3333**
Random Forest	0.9792	0.9943	**0.8289**	0.9750	0.9887	**0.8411**	0.004	0.0056	**0.0122**

Table 4.20 SMOTE-sample method/Original with Ensemble Learning

Model Name	Accuracy Original	Specificity Original	Sensitivity Original	Accuracy ROSE	Specificity ROSE	Sensitivity ROSE	% Improve ment Accuracy	%Improve ment Specificity	%Improve ment Sensitivity
Ada-Boost	0.8400	0.9091	**0.3333**	0.6800	0.7273	**0.3333**	0.1600	0.1818	**0.0000**
XGB Tree	0.8400	0.9091	**0.3333**	0.7200	0.7727	**0.3333**	0.1200	0.1364	**0.0000**
Tree-Bag	0.8000	0.9091	**0.000**	0.7200	0.7727	**0.3333**	0.0800	0.1364	**0.3333**
Random Forest	0.8800	1.000	**0.000**	0.6800	0.7273	**0.3333**	0.2000	0.2727	**0.3333**

III. CONCLUSION

This research considered five different sample based machine learning method to balance imbalanced dataset with four models for ensemble learning as discussed in previous page, comparisons of performance activity by each method used for single classifier and ensemble learning by considering, the accuracy, sensitivity and specificity.

Each method was compared along with original data set, though single classifier was compared too with multiple classifier to predict if single classifier performs better than multiple classifier in term of accuracy, sensitivity and specificity, though is not part of our objective to do this but it would be an advantage for future work.

Furthermore, for balanced data model, Under-sample base method and SMOTE-sample base method perform better in Data 1, while in Data 2 all the five balanced method perform well, Over-sample, Under-sample, Both-sample, ROSE-sample and SMOTE-sample are perform well which the case is reverse in Data 1.

Data 3 Over-sample based method perform well for this data set and Data 4 all of the sample base method perform equally.

For ensemble learning method, four different method was consider for this paper, in Data 1 Random Forest, XGBTree and TreeBag perform well. In Data 2, all the ensemble learning method perform well in different type of balanced method used. For Data 3, only XGBTree have the highest performance in five balanced method used.

For Data 4, Random forest has the highest number performance with sample base method used. In addition, for Boosting ensemble learning, AdaBoost model have a longer time to run in so it advisable to use XGBTree model when considering to use boosting method but health sector and bank sector are more likely to take risk which may be good along the way. For bagging ensemble learning method, TreeBag model are run faster than Random forest which TreeBag can also considered for his timely.

Our overall analysis, point out that Random forest from our ensemble learning used was perform better than remaining three ensemble method use. Likewise balanced sample base method, Under-sample and SMOTE-sample models perform better than remaining three models. In course of performing this comparison, the results shows that new balanced data with ensemble learning have the better accuracy, sensitivity and specificity than original data with the same classification. All this comparison base on type of data set, the case might be different for another data set.

To conclude, all the balanced method, single classifier and ensemble learning works better, though most of single classifier works better than ensemble learning for three data set out of four.

There should be comparison of effect of noise in each data for bagging and boosting model, and percentage increase in classification error between data and sample based method with ensemble learning further research should continue on this.

REFERENCES

[1]. Analytics vidhya content team, (2016.) Practical Guide to Principal Component Analysis (PCA) in R and Python.[Online]Availableat https://www.analyticsvidhya.com/blog/2016/03/practical-guide-principal-component-analysis-python/ [Assessed on 10th September 2019]

[2]. Analytics Vidhya Content Team, 2017. Tutorial on how to handle imbalanced classification problems in machine learning. [Online] Available at https://www.analyticsvidhya.com/blog/2017/03/imbalanced-classification-problem/ [Assessed on 13th Dec 2019]

[3]. Butte, S., Prashanth, A.R. and Patil, S., 2018, April. Machine learning based predictive maintenance strategy: a super learning approach with deep neural networks. In 2018 IEEE Workshop on Microelectronics and Electron Devices (WMED) (pp. 1-5). IEEE.

[4]. Chawla, N.V., Bowyer, K.W., Hall, L.O. and Kegelmeyer, W.P., 2002. SMOTE: synthetic minority over-sampling technique. Journal of artificial intelligence research, 16, pp.321-357.

[5]. Chawla, N.V., Lazarevic, A., Hall, L.O. and Bowyer, K.W., 2003, September. SMOTEBoost: Improving prediction of the minority class in boosting. In European conference on principles of data mining and knowledge discovery (pp. 107-119). Springer, Berlin, Heidelberg.

[6]. Dietterich, T.G., 2000. An experimental comparison of three methods for constructing ensembles of decision trees: Bagging, boosting, and randomization. Machine learning, 40(2), pp.139-157.

[7]. Dietterich, T.G., 1999. An Experimental Comparison of Three Methods for Constructing Ensembles of Decision Trees: Bagging, Boosting, and Randomization THOMAS G. DIETTERICH tgd@ cs. orst. edu Department of Computer Science, Oregon State University, Corvallis, OR 97331. Machine learning, 1, p.22.

[8]. DATACAMP, 2018. Ensemble learning in R with SuperLearner. [Online] Available at

International Journal of Advances in Engineering and Management (IJAEM)
Volume 3, Issue 5 May 2021, pp: 614-620 www.ijaem.net ISSN: 2395-5252

https://www.datacamp.com/community/tutorials/ensemble-r-machine-learning [Assessed on 14th Dec 2019]

[9]. Elhassan, T. and Aljurf, M., 2016. Classification of imbalance data Using tomek link (t-link) combined with random under-sampling (rus) as a data reduction method.

[10]. Galar, M., Fernandez, A., Barrenechea, E., Bustince, H. and Herrera, F., 2011. A review on ensembles for the class imbalance problem: bagging-, boosting-, and hybrid-based approaches. IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), 42(4), pp.463-484.

[11]. J.R. QUINLAN, 2006. Bagging, Boosting and C4.5. [Online] Available at: http://www.cs.ecu.edu/~dingq/CSCI6905/readings/BaggingBoosting.pdf/ [Accessed 5th Jan 2020]

[12]. LIOR ROKACH, 2010. Ensemble-based classifier. Artificial Intelligence Review. Vol. 33, Issue 1-2, pp 1-39. [Online] Available at https://doi.org/10.1007/s10462-009-9124-7 [Assessed on 20th Dec 2019]

[13]. M. PAL., 2005. Random Forest Classifier for remote sensing classification. International Journal of Remote sensing. Vol. 26, Issue 1. [Online] Available at https://doi.org/10.1080/01431160412331269698 [Assessed on 12th Dec 2019]

[14]. Max Kuhn, 2019. The caret Packages. [online] Available athttps://topepo.github.io/caret/subsampling-for-class-imbalances.html[Assessed on 19th Dec. 2019]

[15]. MathWork. 2019. Documentation TreeBagger class. [Online] Available athttps://ch.mathworks.com/help/stats/treebagger-class.html?cv=1[Assessed on 10th Dec.2019]

[16]. Neema, S. and Soibam, B., 2017. The comparison of machine learning methods to achieve most cost-effective prediction for credit card default. Journal of Management Science and Business Intelligence, 2(2), pp.36-41.

[17]. Satyasree, K.P.N.V. and Murthy, J., 2013. An exhaustive literature review on class imbalance problem. Int J Emerg Trends Technol Comput Sci, 2, pp.109-118.

[18]. UCI Machine Learning repository. [Online] Available at: https://archive.ics.uci.edu/ml/datasets.php [Assessed on 10th September 2019]

[19]. Wilson Mongwe, Top of the bell curve, 2018. Tutorial on Random Forest Example. [Online] Available at http://www.wilsonmongwe.co.za/an-interactive-random-forest-test-for-jumps-in-stock-markets-using-r/random_forest_diagram_complete/ [Assessed on 14th Dec 2019]

[20]. Yohannese, C.W., Li, T. and Bashir, K., 2018. A three-stage based ensemble learning for improved software fault prediction: an empirical comparative study. International Journal of Computational Intelligence Systems, 11(1), pp.1229-1247.