International Journal of Computing and Digital Systems
ISSN (2210-142X)
Int. J. Com. Dig. Sys. 10, No.1 (Feb-2021)
http://dx.doi.org/10.12785/ijcds/100128

A SMOTe based Oversampling Data-Point Approach to Solving the Credit Card Data Imbalance Problem in Financial Fraud Detection

Nhlakanipho Mqadi[1], Nalindren Naicker[1] and Timothy Adeliyi[1]

[1] *ICT and Society Research Group; Durban University of Technology, Durban, South Africa*

Received 16 Aug. 2020, Revised 15 Sep. 2020, Accepted 25 Dec. 2020, Published 8 Feb. 2021

Abstract: Credit card fraud has negatively affected the market economic order, broken the confidence and interest of stakeholders, financial institutions, and consumers. Losses from card fraud is increasing every year with billions of dollars being lost. Machine Learning methods use large volumes of data as examples for learning to improve the performance of classification models. Financial institutions use Machine Learning to identify fraudulent patterns from the large amounts of historical financial records. However, the detection of credit card fraud remains as a significant challenge for business intelligence technologies as most datasets containing credit card transactions are highly imbalanced. To overcome this challenge, this paper proposed the use of the data-point approach in machine learning. An experimental study was conducted applying Oversampling with SMOTe, a data-point approach technique, on an imbalanced credit card dataset. State-of-the-art classical machine learning algorithms namely, Support Vector Machines, Logistic Regression, Decision Tree and Random Forest classifiers were used to perform the classifications and the accuracy was evaluated using precision, recall, F1-score, and the average precision metrics. The results show that if the data is highly imbalanced, the model struggles to detect fraudulent transactions. After using the SMOTe based Oversampling technique, there was a significant improvement to the ability to predict positive classes.

Keywords: Data Imbalance; Fraud Detection; Machine Learning; Oversampling

1. INTRODUCTION

In 2015, losses from card fraud reached approximately $21.84 billion, and by 2020, card fraud across the world was expected to reach nearly $32 billion [1]. Card fraud has negatively affected the market economic order, broken the confidence and interest of stakeholders, financial institutions, and consumers. The ability to detect fraud mitigates the risk of fraudulent activities and financial losses [2]. Machine learning (ML) is the science of designing and applying algorithms that are able to learn patterns from historic data [3]. According to Jiang *et al.* in [3], one aspect of ML refers to the ability of systems to recognize and classify classes existing in the data. The ML methods use large volumes of data as examples for learning. The collection of instances of data is referred to as datasets and machine learning methods uses two sets of data to learn: training dataset and testing dataset [4]. The introduction of ML has enabled financial institutions to use historical credit card data to learn the patterns with an aim of distinguishing between fraudulent and legitimate transactions [5]. However, existing methods are not sufficient in real world situations. Adewumi & Akinyelu in [6] stated that, in real life, the amount of legitimate transaction recorded highly outweigh the fraudulent transactions. The outweighing is known as class imbalance and as a result, most techniques of detecting card fraud are still incapable of achieving ideal fraud detection abilities [6]. In consequence, detection of credit card fraud remains as a significant challenge for business intelligence technologies as most datasets containing credit card transactions are highly imbalanced.

This study was conducted to investigate if the data-point approach can help reduce the impact of the class imbalance problem. In this paper, the case where the majority classes (legitimate transactions) dominate over minority classes (fraudulent transaction), causing the machine learning classifiers to be more biased towards majority classes is referred to as imbalanced data. Imbalanced data and bias are one of the major problems in the field of data mining and machine learning as most ML algorithms assume that data is equally distributed [7]. The

E-mail: 21349052@dut4life.ac.za, naickern@dut.ac.za, timothya2@dut.ac.za

failure to handle imbalance data compromises the integrity and predictive abilities of machine learning system resulting in high financial impact. The data-point level approach consists of techniques for re-sampling the data in order to deal with imbalanced classes. These techniques include oversampling, under-sampling, and feature selection [8]. The aim of this paper was to assert the precision, recall, and F1 score of ML algorithms before and after the application of the data-point technique. The scope of this paper covers the investigation of ML model's predictive accuracy with imbalance credit card dataset. The term accuracy can be defined as the percentage of correctly classified instances (TP + TN) / (TP + TN + FP + FN). Where TP, FN, FP and TN represent the number of true positives, false negatives, false positives and true negatives, respectively. Predictive Accuracy refers to the ability to classify legitimate and fraudulent transactions successfully.

2. RELARED WORK

Many other studies [9-11] reviewed and compared the existing financial fraud detection models to identify the method with the best performance. Patil *et al.* in [10] used the confusion matrix and found that, the Random Forest model performed better as compared to Logistic Regression and Decision Tree in terms of accuracy, precision and recall parameters, whereas, Albashrawi in [11] found that the Logistic Regression model appeared to be the leading machine learning technique in detecting financial fraud. Other researchers [12-13] have proposed using a hybrid approach. These approaches show some improvements on the existing methods and recognize strengths of fraud detection models; for example, Chouiekha *et al.* in [14] who found that Deep Learning algorithms such as Convolution Neural Networks (CNN) technique has better accuracy versus traditional machine learning algorithms. Rekha *et al.* in [15] presented a comparison of the performance of several boosting and bagging techniques from imbalanced datasets. According to Rekha *et al.* in [15], Oversampling technique takes full minority samples in the training data into consideration while performing classification. However, the presence of some noise (in the minority samples and majority samples) degrades the classification performance. The study proposed noise filtering using boosting and bagging. The performance was evaluated the with the state-of-the-art methods based on ensemble learning like AdaBoost, RUSBoost, SMOTEBoost, Bagging, OverBagging, SMOTEBagging on 25 imbalance binary class datasets with various Imbalance Ratios (IR). The experimental results show that their approach works as promising and effective for dealing with imbalanced datasets using metrics like F-Measure and AUC.

Bauder and Khoshgoftaar in [16] focused on finding the ability to recognize the fraudulent activities of Medicare Part B (i.e., medical insurance) providers, which comprised of falsified actions, which was the exploitation of patients and the billing for non-rendered services. Providers and individuals that have been expelled from partaking in Federal healthcare programmes in the United States committed this fraud. The study discusses the processing of Part B dataset and proposed a novel fraud label mapping method using the providers that have been recognized as fraudulent. The dataset was labelled and extremely imbalanced with only a few number of cases which were flagged as fraud. Seven class distributions were generated from the dataset and their behaviors were evaluated using six ML techniques, in the interest of fighting the class imbalance problem while also achieving a good fraud identification performance. The findings revealed that the learner with the best Area Under the ROC Curve (AUC) score of 0.87302 was RF100 using a class distribution of 90:10. In addition, learners using a class distribution that is more balanced as the 50:50 distribution produced less favourable results. The study concluded that keeping more of the dominant class improved the ability to detect Medicare Part B fraud.

Similarly, Liu *et al.* in [17] conducted an experiment to propose two algorithms to overcome the deficiency of using under-sampling in handling the problem of class imbalance. The deficiency was that when under-sampling is applied, many majority classes are ignored. Therefore, the study proposed EasyEnsemble and BalanceCascade. EasyEnsemble divides the majority class into several smaller chunks, then the chunks are independently used to train the learner and at the end, all the outputs by the learners are combined. BalanceCascade uses a sequential training-based approach, wherein each sequence, the correctly classified examples of the majority class are eliminated from being further evaluated in the next sequence. The findings showed that compared to many existing methods, both the EasyEnsemble and BalanceCascade have a higher F-measure, G-mean, and AUC values and the training time was found to be closely similar to under-sampling, which according to Liu *et al.* in [17], was significantly faster compared to other approaches.

A paper by Ebenuwa *et al.* in [18] presented Variance Ranking (VR), which is a feature selection-based method for solving the problem of datasets with imbalanced classes. The work-involved data from four databases, namely, Wisconsin Breast Cancer dataset, Pima Indians Diabetes dataset, the Cod-RNA dataset, and BUPAliver disorders dataset. The Information Gain Technique (IGT) and The Pearson Correlation (TPC), which are two popular feature selection methods that were used to compare the results of VR using a novel comparison technique, called the Ranked Order Similarity (ROS). The decision tree, logistic regression, and support vector machine were used to train the classifiers and it was found that the proposed method performed better than the benchmarks used in the experiment.

While there have been many studies on financial fraud detection, class imbalance problems and classification algorithms using machine learning, the overwhelming

conclusion is that misclassification of fraud and non-fraud transactions continues to be a persisting problem when the dataset is imbalanced. There has been little research to find the best combination of the data-point approach with the classification algorithm to address class imbalance in credit card fraud. To investigate this problem, the study examined four well-known ML fraud identification algorithms with imbalanced credit card fraud dataset to determine whether using the Oversampling method based on Synthetic Minority Oversampling Technique (SMOTe) improves the predictive accuracy. The performance of credit card fraud identification models was then analyzed using standard performance metrics. This paper provides an intensive comparative and statistical analysis of the prediction results.

The remainder of this paper is structured as follows; discussion of the Research Methodology in section 3; the presentation of the experimental results, discussion, and conclusion of the study in section 4.

3. RESEARCH METHODOLOGY

An experimental study was conducted surveying oversampling, one of the data-point level techniques to prove the effect of handling class imbalance on the credit card dataset. The design of an experimental research is more suitable where there is manipulation of the independent variable and the effect are tested on the dependent variable [19]. An experimental design was more suitable for this study to investigate the predictive accuracy of machine learning models for fraud identification before manipulation and after the manipulation using Oversampling to handle imbalanced data on the credit card dataset.

A. Classifications

The Support Vector Machine (SVM), Logistic Regression (LR), Decision Tree (DT) and Random Forest (RF) algorithms were selected for the experiment. The algorithms were used to train and test the fraud detection model following the train, test, and predict approach. Support Vector Machine is a classification algorithm that uses supervised learning to distinguish normal and fraud classes [20]. SVM will create a hyperplane to segregate the transactions by grouping them on either side of the hyperplane as normal or fraud respectively. Logistic Regression is a statistical classifier used to allocate interpretations to individually separate and distinct set of classes. The classification is transformed using the logistic sigmoid function to return a probability value, which can then be mapped, to be either normal or fraud. Logistic Regression predictions allow only specific categories or values [21]. Decision Tree is a method for making a Decision Tree from training data classification. The classifier creates a tree like structures, where, the leaves symbolize the classifications, the non-leaf nodes symbolize features, and the branches symbolize combinations of features that lead to the classifications [22]. The Random Forest algorithm is a supervised learning classifier for regression and classification. The ensemble technique is made up of numerous decision trees; during the experiment, the forest was made of 600 trees. According to Jiang *et al*. in [23], the trees each produce a class prediction and the class with more occurrences come to be the final prediction of the classifier. The four classification algorithms are used with the data-point approach to find the best combination and strategy for solving the class imbalance problem in credit card fraud detection.

B. Dataset

The experiment was conducted using a credit card dataset from a provider called Kaggle found at https://www.kaggle.com/mlg-ulb/creditcardfraud/home. The dataset comprises of European cardholders' transactions, where there are 492 frauds out of a sample size of 284807 transactions. The minority class, which was recorded as actual fraud cases in the dataset only made up for 0.172% of all transactions.

$$\frac{fraud}{sample\ size} * 100 = \text{fraud}_{\text{cases}} \quad (1)$$

There are 31 features in the dataset. Features V1, V2, up to V28 were the principal components gained through the Principal Component Analysis (PCA) conversion due to issues of confidentiality; the only features, which were not converted with PCA, were 'Time', 'Amount', and 'Class'. The 'Class' feature contains a numeric value of 0 to indicate a normal transaction and 1 to indicate fraud. The dataset was chosen because it is labelled, highly imbalanced, and convenient to the researcher because it is easily accessible making it more suitable for the requirements of this experiment.

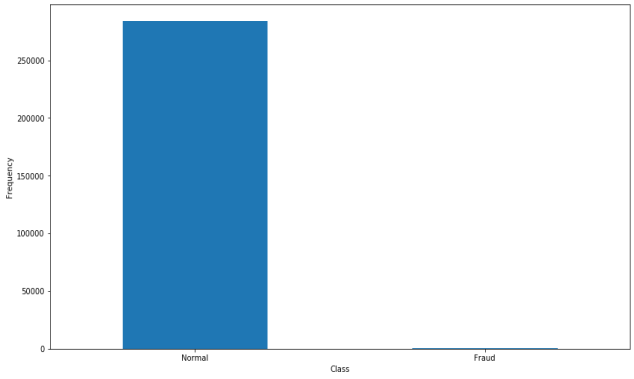

Figure 1. Original Transaction Class Distribution

Fig. 1 shows a bar graph representation of the frequency of the normal (legitimate) classes versus fraud classes. A dataset is imbalanced if at least one of the classes constitutes only a very small minority. The bar for the fraud class is almost invisible in Figure 1. An imbalance dataset is best evaluated with sensitivity matrices, whereas, a balanced dataset is best evaluated

using the standard accuracy score [24]. In this paper, we observed both sensitivity and standard performance matrices on both the balanced and imbalanced datasets to ensure a fair comparison and to gain an in-depth understanding of the ability to predict the positive and negative classes of the credit card fraud dataset.

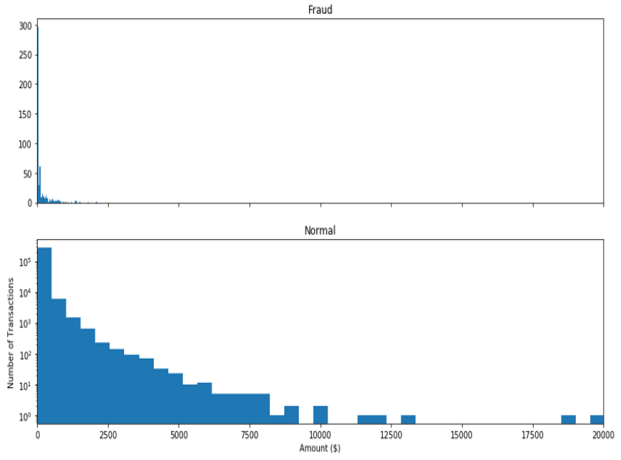

Figure 2. Amount per transaction by class

Fig. 2 provides a visual representation of the amount in dollars that the fraudulent transactions cater for in the dataset versus the legitimate transactions. The amount is within the range of the majority of the normal amount, which makes it difficult to use amount as a parameter to distinguish between the classes of transactions.

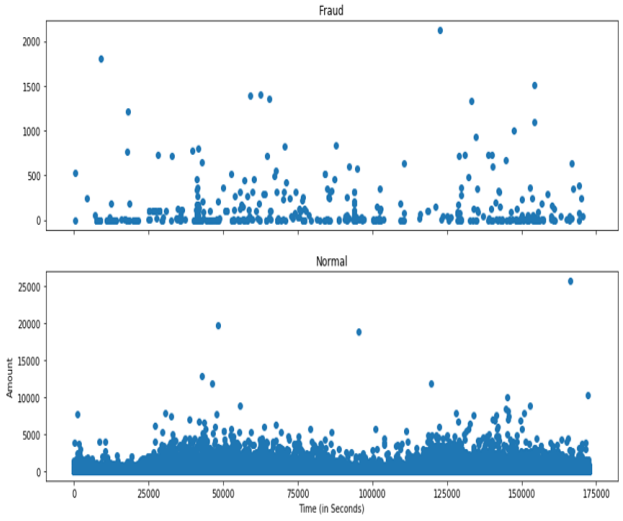

Figure 3. Time of transaction vs Amount by class

The plot in Fig. 3 shows a visual representation of how often do fraudulent versus legitimate transactions occur during certain periods.

C. The Data-Point Approach

This study investigated using the data-point approach to solve the data imbalance problem. The data-point level approach consists of interventions on the data to alleviate the effect of the class imbalance, and it has the flexibility to be used with the latest algorithms such as support vector machines, decision tree, and logistic regression as stated by Hassib in [25]. Our paper used the Oversampling technique to investigate the effect of the data-point method. Oversampling refers to increasing the count of the minorities to balance with the majority class. According to Somasundaran & Reddy in [26], this method tends to duplicate the data already available or generate data based on available data. Oversampling attempts to balance the dataset by adding the number of minority classes. The objective of using oversampling is to avoid losing samples of the majority class, because that could result in losing some valuable data. Instead, new samples of the minority classes are produced using methods such as SMOTe, bootstrapping, and repetition [27].

The experiment was conducted using SMOTe, which is a method based on nearest neighbours judged by Euclidean Distance amongst data points within a feature space. The number of artificial samples to be produced is indicated by a percentage passed as a parameter and this percentage is always a multiple of 100 [28]. An Oversampling percentage of 100 will create new samples for each minority instance, therefore doubling the total count of the minority class in the dataset, for example, the 492 fraud cases would become 984. Likewise, an oversampling percentage of 200 would triple the total count of the minority class.

In SMOTe,

- The k nearest neighbours is established for each of the minority class, given that they are belonging to the same class.

$$(\text{SMOTe }\%)\ /100 = k \qquad (2)$$

- The difference between the feature vector of the considered instance and the feature vectors of the k nearest neighbours are found. So, k number of difference vectors are obtained.
- Each of the k difference vectors are multiplied using a random number between the range of 0 and 1 (exclusive of 0 and 1).
- Lastly, at each repetition, the product of the random numbers and the difference vectors, are added to the feature vector of the original minority instance.

Resampling using SMOTe was implemented by importing and inheriting a library from *imblearn* to reduce development time. The implementation was conducted by calling the SMOTe method and passing parameters. Using inheritance allowed the researcher to reuse existing code to reduce programming time, increase efficiency and to allow flexibility. Table 1 below shows the parameters and the values used during the experiments [29].

Table 1. SMOTe method call parameters [29]

Parameter	Value
Sampling Strategy	Auto
Random State	None
K Neighbours	5
M Neighbours	Deprecated
Out Step	Deprecated
Kind	Deprecated
SVM Estimator	Deprecated
N Jobs	1
Ratio	None

During the experiment, different combination of the parameters was investigated to find a combination of parameters that produced the ideal results. SMOTe represents an improvement over Random Oversampling in that the minority class is oversampled resulting in a sub-optimal performance [30–31]. However, Douzas *et al.* in [32] stated that, in highly imbalanced datasets, too much Oversampling might result in overfitting. To combat this issue of oversampling we used data-point approach with SMOTe to interpolate existing dataset to generate new instances. This approach aims at eliminating both between-class imbalances and within-class imbalances hence avoiding the generation of random samples.

D. Experiment

The study used the python programming language and Google Colab. Python offers succinct and human readable code, a wide range of libraries and frameworks for implementing ML algorithms that will reduce development time hence it will be more suitable for this study. The code was executed on the Google Colab notebook, which execute code on Google's cloud servers, leveraging the power of Google hardware, including Graphics Processing Units (GPUs) and Tensor processing unit (TPUs), running on a Google browser. The first step was to import all the libraries. Once all the libraries were imported, the creditcard.csv dataset was uploaded. The dataset was validated to ensure that there were no null values or missing columns. An exploratory data analysis was performed to visualize and gain insight on the data. We then identified independent and dependent features. The dependent feature was stored separately on the Y variable and the independent features were stored in the X variable. The Y variable was the column that contained the indicator, of whether the transaction was normal (labelled as 0) or fraud (labelled as 1), which was the variable we were trying to predict. The next step was to split the data into a training set and a testing set using a class from the sklearn library to call the train-test-split function. The train-test-split function accepts the independent variable X, dependent variable Y and test size. The test-size parameter specifies the ratio to split the original size of the dataset, which indicate that 70% of the original dataset was used to train the model and 30% of the dataset was used to test the model. The next phase of the experiment was to build and train our model. We used each of the selected algorithms discussed in the classifications section of the research methodology. We fit each model with the x-train and y-train training data. We then used the x-test data to try to predict the y-test variable.

4. Results and Discussion

A. Results

Once the experiment was concluded, we compare the y-test variable to the prediction results to generate a classification report. Precision measures the ability of a model to predict the positive class. Precision = TP / (TP + FP). Recall describes how good the model is at predicting the positive class when the actual outcome is positive. Recall = TP / (TP + FN). A precision-recall curve is a plot of the precision (y-axis) and the recall (x-axis) for different thresholds. Askari in [33] stated that, using both recall and precision is valuable to measure the predictive strengths of the model in situations where the distribution between two classes is imbalanced. The F_1 score is the accuracy measurement of the test. Both the precision and the recall score are considered when calculating the F1 score. The initial results we achieved with an imbalance dataset. The sensitivity perfomance metrices used to evaluate the imbalance dataset results are:

The Precision score that was calculated as follows:

$$\text{Precision} = \frac{True_{positives}}{True_{positives} + False_{positives}} \quad (3)$$

The Recall score that was calculated as follows:

$$\text{Recall} = \frac{True_{positives}}{True_{positives} + False_{negetives}} \quad (4)$$

The F_1 score that was calculated as follows:

$$F_1 = 2 * \frac{precision * recall}{precision + recall} \quad (5)$$

The Average Precision (AP) is a score that is computed from the prediction score. AP summarizes a precision-recall curve as the weighted mean of precisions achieved at each threshold, with the increase in recall from the previous threshold used as the weight [34]:

$$\sum_n * (R_n - R_{n-1}) P_n = \text{AP} \quad (6)$$

The above formula computes the average precision, where P_n and R_n are the precision and recall at the nth threshold. Precision and recall are always between 0 and

1. Therefore, AP falls within 0 and 1, AP is metric used to measure the accuracy of a classifier, which means if number is closer to 1, the classifier is more accurate. To present the results, the zero (0) was used represent legitimate transactions and the one (1) represent the fraudulent transactions. The lowest possible value is represented by 0.00 (0%) and the highest possible value is represented as 1.00 (100%). Table 2 below uses ALG for algorithm, C for Class, P for Precision, R for recall, F1 for F1-score, and AC for accuracy. Table 2 below compares the scores of all the four classifies; SVM, LR, DT, and RF before Oversampling. Table 2 also shows the initial classification report comparison for all the algorithms before the data-point level approach technique was applied on the credit card dataset.

Table 2. Comparison of imbalance dataset classification before Oversampling

ALG	C	P	R	F1	AC
SVM	0	1.00	1.00	1.00	1.00
	1	0.00	0.00	0.00	AP = 0.00
LR	0	1.00	1.00	1.00	1.00
	1	0.42	0.47	0.44	AP = 0.46
DT	0	1.00	1.00	1.00	1.00
	1	0.58	0.65	0.61	AP = 0.38
RF	0	1.00	1.00	1.00	1.00
	1	0.90	0.53	0.67	AP = 0.48

The closer the curve to the value of one on upper right corner, the better the quality. If the the curve is leaning towards the lower left corner, then the quality of the classification is poor. Fig 4, 5, 6, and 7 below are the precision-recall curve before Oversampling with SMOTe was applied. The curves represent the quality of each classifier with an imbalance dataset.

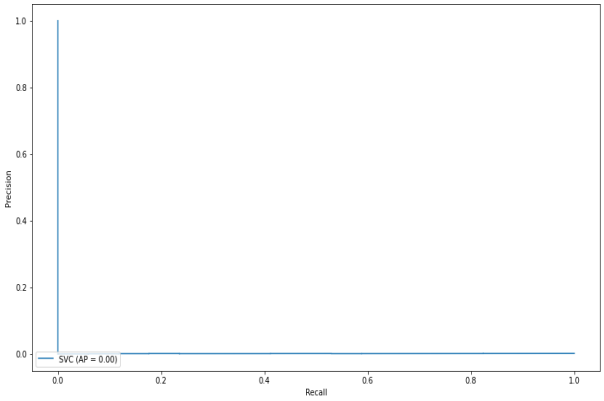

Figure 4. SVM Precision-Recall curve

Fig. 4 shows the precision-recall curve of the SVM classification where the average precision computed was 0.00. The SVM is leaning towards the lower left corner, which represent a classifier with poor performance. In our case, the SVM classifier performed the worse than all other classifies. There was high bias towards the majority class.

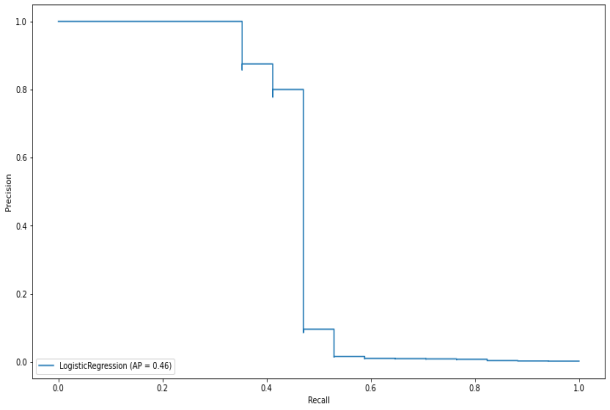

Figure 5. Logistic Regression Precision-Recall curve

Fig. 5 shows the precision-recall curve of the Logistic regression classification where the average precision computed was 0.46.

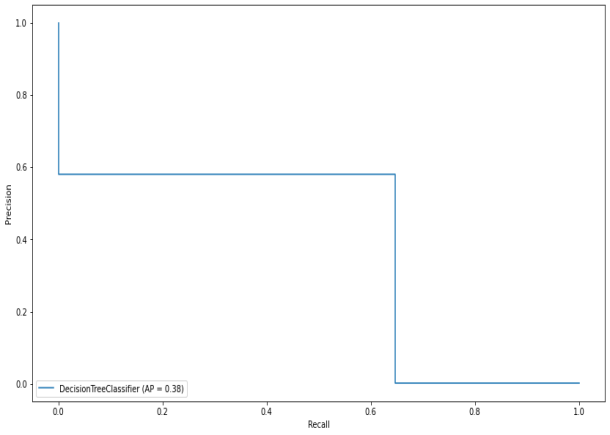

Figure 6. Decision Tree Precision-Recall curve

Fig. 6 shows the precision-recall curve of the Decision tree classification where the average precision computed was 0.38.

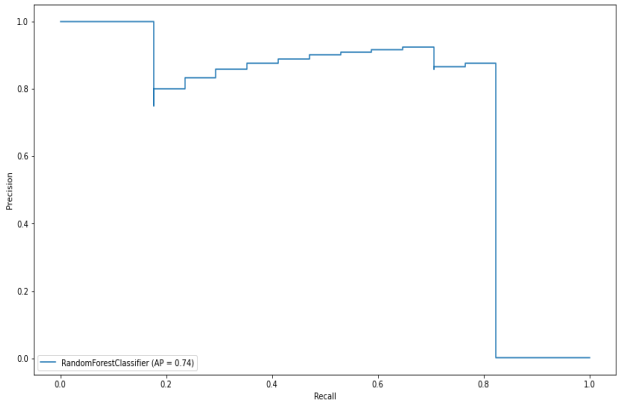

Figure 7. Random forest Precision-Recall curve

Fig. 7 shows the precision-recall curve of the Random forest classification where the average precision computed was 0.48.

B. Oversampling with SMOTe

The next step of the experiment was to use SMOTe to resample the original dataset. We used the default values on most of the parameters, except for the random-state. The random-state parameter controls both the randomness of the bootstrapping of the samples used when building trees and the sampling of the features to consider when looking for the best split at each node. After multiple iterations, the results presented were obtained using a random state of 42.

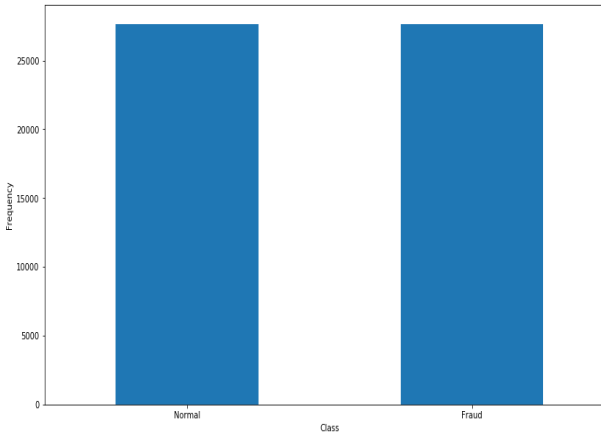

Figure 8. Transaction Class Distribution after Oversampling

Fig. 8 shows the transaction class distribution after Oversampling. The cases are evenly balanced for both normal and fraud. The dataset was then fed into the prediction model following the split-test-train-predict cycle. We compared the y-test to the prediction to generate a classification report of oversampling. Table 3 shows the classification report for all the algorithms after Oversampling was applied to mitigate the effect caused by class imbalance.

Table 3. Comparison of classifications after Oversampling

ALG	C	P	R	F1	AC
SVM	0	0.60	0.37	0.46	0.57
	1	0.55	0.76	0.64	AP = 0.53
LR	0	0.97	0.97	0.97	0.97
	1	0.97	0.97	0.97	AP = 0.96
DT	0	1.00	1.00	1.00	1.00
	1	1.00	1.00	1.00	AP = 1.00
RF	0	1.00	1.00	1.00	1.00
	1	1.00	1.00	1.00	AP = 1.00

Table 3 shows high precision, recall, F1-score and accuracy for the decision tree and random forest.

Fig. 9, 10, 11 and 12 plots the respective Precision-Recall curve of the classification after Oversampling with SMOTe. The goal is to observe whether the P-R curve is towards the upper right corner of the chart to verify that the accuracy has improved. The closer the curve to the value of one in the y-axes, the better the quality.

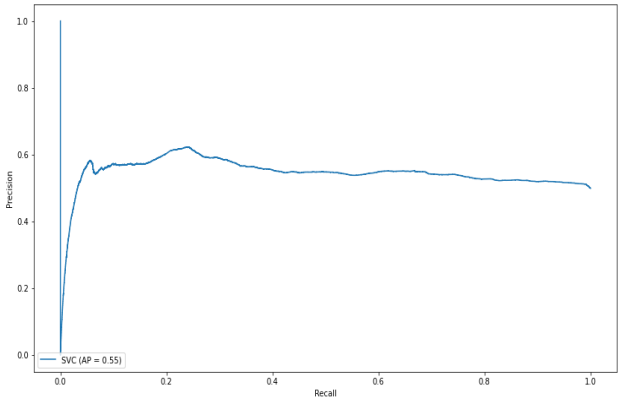

Figure 9. SVM Precision-Recall curve, AP = 0.53

Fig. 9 show the precision-recall curve of the SVM classification where the average precision computed was 0.53.

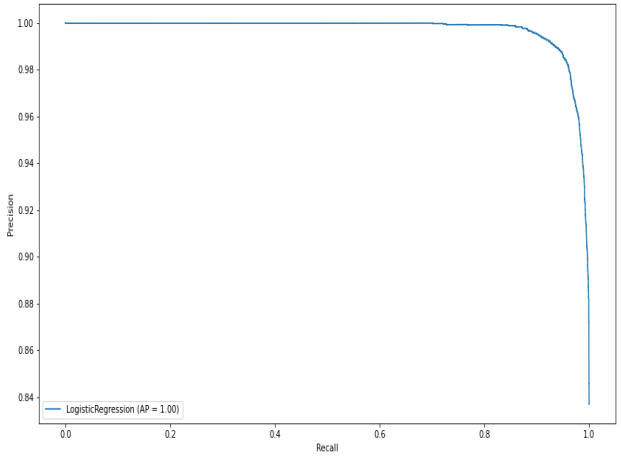

Figure 10. LR Precision-Recall curve, AP = 0.96

Fig. 10 shows the precision-recall curve of the Logistic regression classification where the average precision computed was 0.96.

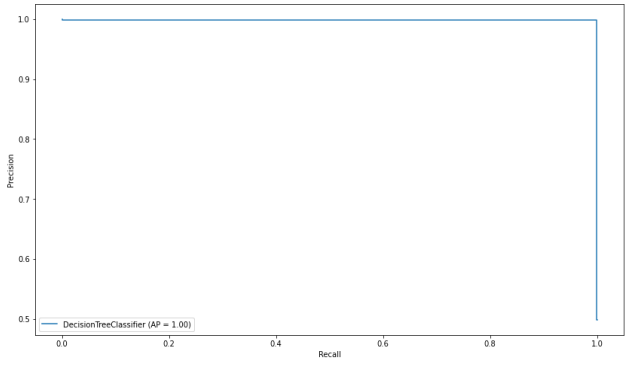

Figure 11. DT Precision-Recall curve, AP = 1.00

Fig. 11 shows the precision-recall curve of the Decision tree classification where the average precision computed was 1.00.

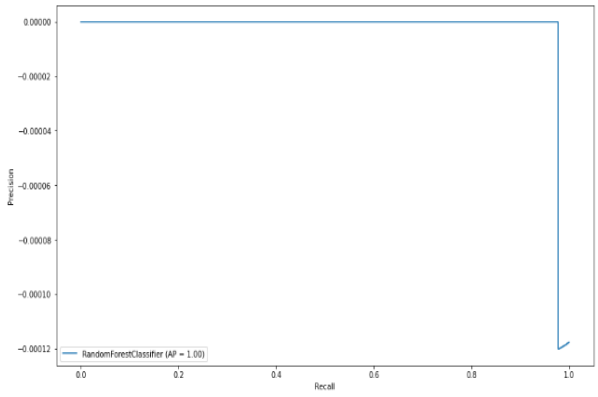

Figure 12. RF Precision-Recall curve, AP = 1.00

Fig. 12 shows the precision-recall curve of the Random forest classification where the average precision computed was 1.00. A P-R curve is a great way to provide a graphical visualization of the quality of a classifier. A P-R curve that is a straight line towards the upper right corner, such as the one of Fig. 11 and Fig. 12 represents the best possible quality. The two P-R curves tell us that the classifiers were able to predict the positive classes at a 100% accuracy.

C. Discussion

This section provides a discussion of the classification results before and after using oversampling by highlighting the improvements observed and ranking the algorithms. The classification report with the original dataset revealed that the Random forest model was the best performer, the precision score of 0.90, but the Recall was 0.53, therefore, the cross-validating shows that the precision score is misleading. To further validate the model, the computed average precision was 0.48, revealing that the model was not producing the ideal performance and further improvements were necessary.

The SVM model was the worst performing with a precision score of 0.00 for fraud. The score of 0.00 means that the SVM model failed to identify fraud cases with imbalance data. All the algorithms scored 1.00 for legitimate cases, which means that due the to the imbalance level, the majority class was completely dominant. To determine whether there was any improvement for fraud detection, the following formula calculated the improvement for the Precision, Recall, and F_1 score:

$$score_{after} - score_{before} = \%\, value \quad (7)$$

After using the SMOTe Oversampling technique, the Precision score improved by 55% for SVM, 55% for Logistic Regression, 42% for the Decision Tree, and 10% for Random forest for the positive class.

The Recall score shows that the strength of identifying True Positive (which are actual fraudulent cases) improved by 76% for SVM, 50% for Logistic Regression, 47% for Random forest, and 39% for the Decision Tree for the positive class.

The results reveal that F1-Score improved by 64% for SVM, 53% for Logistic Regression, 35% for the Decision Tree, and 33% for Random forest for the positive class. Comparing the F1 scores show that when the ability to detect positive classes was improved.

An interesting observation was that the classification of negative class for the Logistic Regression, Decision Tree and Random forest algorithms was good and consisted throughout the experiment. SVM performed well initially with the overall accuracy score of 100%; however, after using Oversampling, the score was 47%, meaning that even though the ability to recognize positive classes improved, the ability to recognize negative classes degraded. Therefore, SVM is not an ideal solution for credit card fraud detection.

Based on the results, the Random forest algorithm is the leading algorithm. The algorithms ranked from best in the following order: Random forest, Decision tree, Logistic regression, and SVM.

D. Conclusion

The results show that if the data is highly imbalanced, the model struggles to detect fraudulent transactions. After using the SMOTe based Oversampling technique, which is a data-point approach, there was a significant improvement to the ability to predict positive classes. Based on the findings, the random forest and decision tree algorithms produced the best performance with credit card dataset.

Future research can perform a cross validation or comparison across multiple datasets to verify the consistency of the data-point approach in handling imbalance credit card fraud datasets. Further studies can investigate building and deploying a real-time solution that can detect fraud as and when the transaction is occurring.

ACKNOWLEDGMENT

KIND ACKNOWLEDGMENT TO THE DURBAN UNIVERISTY OF TECHNOLOGY FOR PROVIDING THE RESOURCES FOR THIS RESEACH STUDY.

REFERENCES

[1] D. Robertson, The Nelson Report. October. Available online: http://www.nelsonreport.com/upload/content_promo/The_Nelson_Report_10-17-2016.pdf (accessed on 03 February 2019).

[2] D. Huang, D. Mu, L. Yang, and X. Cai, CoDetect: Financial Fraud Detection with Anomaly Feature Detection. National Natural Science Foundation of China, vol. 6, no. 2, pp. 19161-19174, 2018. DOI: 10.1109/ACCESS.2018.2816564

[3] C. Jiang, J. Song, G. Liu, L. Zheng, and W. Luan, Credit Card Fraud Detection: A Novel Approach Using Aggregation Strategy and Feedback Mechanism. IEEE Internet of Things Journal, vol. 5, no. 5, pp. 3637-3647, 2018. DOI:10.1109/JIOT.2018.2816007

[4] S. Kotsiantis, D. Kanellopoulos, and P. Pintelas, Handling imbalanced datasets: A review. GESTS International Transactions on Computer Science and Engineering, vol.30, no. 1, pp. 25-36, 2006. DOI: https://doi.org/10.1007/s13369-016-2179-2

[5] A. O. Adewumi, and A. A. Akinyelu, A survey of machine-learning and nature-inspired based credit card fraud detection techniques. Int J Syst Assur Eng Manag, vol. 8, no. 2, pp. 937-953, 2017. https://doi.org/10.1007/s13198-016-0551-y

[6] Y. Bian, M. Cheng, C. Yang, Y. Yuan, Q. Li, and J. L. Zhao *et al.*, Financial fraud detection: a new ensemble learning approach for imbalanced data. PACIS 2016 Proceedings, vol. 315, no. 1, pp. 1-11, 2016.

[7] A. T. Elhassan, M. Aljourf, F. Al-Mohanna, and M. Shoukri, Classification of Imbalance Data using Tomek Link (T-Link) Combined with Random Under-sampling (RUS) as a Data Reduction Method. Global J Technol Optim, vol. 1, no. 1, pp. 1-11, 2017. DOI: 10.4172/2229-8711.S1111

[8] K. Sotiris, K. Dimitris, and P. Panayiotis, Handling imbalanced datasets: A review. GESTS International Transactions on Computer Science and Engineering, vol. 30, no. 1, pp. 1-12, 2016.

[9] M. Zanin, M. Romance, S. Moral, and R. Criado, Credit card fraud detection through parenclitic network analysis. IEEE Access, vol. 1, no. 1, pp. 1-8, 2017. https://doi.org/10.1155/2018/5764370

[10] S. Patil, V. Nemade, and P. Kumar, Predictive Modelling for Credit Card Fraud Detection Using Data Analytics. Computational Intelligence and Data Science, vol. 132, no. 1, pp. 385–395, 2018. https://doi.org/10.1016/j.procs.2018.05.199

[11] M. Albashrawi, Detecting Financial Fraud Using Data Mining Techniques: A Decade Review from 2004 to 2015. Journal of Data Science, vol. 14, no. 1, pp. 553-570, 2016.

[12] R. A. Bauder, and T. M. Khoshgoftaar, The effects of varying class distribution on learner behavior for medicare fraud detection with imbalanced big data. Health Information Science and Systems, vol, 6 no. 9, pp. 1-14, 2018.

[13] K. Randhawa, C. K. Loo, M. Seera, C. P. Lim, and A. K. Nandi, Credit Card Fraud Detection Using AdaBoost and Majority Voting. IEEE Access, vol. 6, no. 1, pp. 14277-14284, 2018.

[14] A. Chouiekha, E. Hassane, and E. Haj, ConvNets for Fraud Detection analysis. Procedia Computer Science, vol. 127, no. 1, pp133–138, 2018. https://doi.org/10.1016/j.procs.2018.01.107

[15] G. Rekha, A. K. Tyagi, and R. V. Krishna, Solving Class Imbalance Problem Using Bagging, Boosting Techniques, with and Without Using Noise Filtering Method. International Journal of Hybrid Intelligent Systems, vol. 15, no. 2, pp. 67–76, 2019. DOI: 10.3233/HIS-190261

[16] R. A. Bauder and T. M. Khoshgoftaar, The effects of varying class distribution on learner behaviour for Medicare fraud detection with imbalanced big data. Health Information Science and Systems, vol. 6, no. 9, pp. 1-14, 2018. doi: 10.1007/s13755-018-0051-3

[17] X. Liu, J. Wu, and Z. Zhou, Exploratory Undersampling for Class-Imbalance Learning. IEEE Transactions On Systems, Man, And Cybernetics, vol. 39, no. 2, pp. 539-550, 2009. Doi: 10.1.1.309.1465

[18] S. H. Ebenuwa, S. Sharif, and M. Alazab, Variance Ranking Attributes Selection Techniques for Binary Classification Problem in Imbalance Data. IEEE Access, vol. 7, no. 1, pp. 24649-24666, 2019.

[19] L. S. Feldt, A comparison of the precision of three experimental designs employing a concomitant variable. Psychometrika, vol. 23, no. 1, pp. 335-353, 1958. DOI: https://doi.org/10.1007/BF02289783

[20] E. Lejon, P. Kyosti, and J. Lindstrom, Machine learning for detection of anomalies in press-hardening: Selection of efficient methods. Process IT Innovations R&D Centre, vol. 1, no. 1, pp. 1079-1083, 2018. https://doi.org/10.1016/j.procir.2018.03.221

[21] G. Baader, and H. Krcmar, Reducing false positives in fraud detection: Combining the red flag approach with process mining. International Journal of Accounting Information Systems, vol. 31 no. 1, pp. 1–16, 2018. https://doi.org/10.1016/j.accinf.2018.03.004

[22] I. Sadgali, N. Sael, and F. Benabbou, Performance of machine learning techniques in the detection of financial frauds. Procedia Computer Science, vol. 148, no. 1, pp. 45-54, 2018. https://doi.org/10.1016/j.procs.2019.01.007

[23] K. Jiang, J. Lu, K. Xia, and L. Zheng, A Novel Algorithm for Imbalance Data Classification Based on Genetic Algorithm Improved SMOTE. Arab J Sci Eng, vol. 41, no. 1, pp. 3155-3266, 2016. DOI: https://doi.org/10.1007/s13369-016-2179-2

[24] N. Malini, and M. Pushpa, Analysis on Credit Card Fraud Identification Techniques based on KNN and Outlier Detection. Advances in Electrical, Electronics, Information, Communication and Bio-Informatics (AEEICB), vol. 1, no. 1, pp 1-12, 2017. DOI: 10.1109/AEEICB.2017.7972424

[25] E. M. Hassib, A. I. El-Desouky, E. M. El-Kenawy, and S. M. Ghamrawy, Imbalanced Big Data Mining Framework for Improving Optimization Algorithms Performance. IEEE Access, vol. 7 no. 1, pp. 170774-170795, 2019.

[26] A. Somasundaran, and U. S. Reddy, Data Imbalance: Effects and Solutions for Classification of Large and Highly Imbalanced Data. Proc. of 1st International Conference on Research in Engineering, Computers and Technology, vol. 25, no. 10, pp. 28- 34, 2016.

[27] A. D. Pozzolo, G. Boracchi, O. Caelen, C. Alippi, and G. Bontempi, Credit Card Fraud Detection: A Realistic Modelling and a Novel Learning Strategy. Transactions on neural networks and learning systems, vol. 29, no. 8, pp. 3784-3797, 2018. DOI: 10.1109/TNNLS.2017.2736643

[28] A. Mubalik, and E. Adali, Multilayer Perception Neural network technique for fraud detection. Computer Science and Engineering (UBMK), vol. 1, no. 1, pp. 383-387, 2017.

[29] Imbalanced-learn. Available online: https://imbalanced-learn.readthedocs.io/en/stable (accessed on 15 June 2020).

[30] K. Jiang, J. Lu, and K. Xia, A novel algorithm for imbalance data classification based on genetic algorithm improved SMOTE. Arabian journal for science and engineering, vol. 41, no. 8, pp. 3255-3266, 2016.

[31] A. Agrawal, H. L. Viktor, and E. Paquet, November. SCUT: Multi-class imbalanced data classification using SMOTE and cluster-based undersampling. In 2015 7th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management (IC3K). IEEE, vol. 1, pp. 226-234, 2015.

[32] G. Douzas, F. Bacao, and F. Last, Improving imbalanced learning through a heuristic oversampling method based on k-means and SMOTE. Information Sciences, vol. 465, no. 1, pp. 1-20, 2018.

[33] S. Askari, and A. Hussain, Credit Card Fraud Detection Using Fuzzy ID3. Computing. Communication and Automation (ICCCA), vol. 40, no. 1, pp. 446-452, 2017. DOI: 10.1109/CCAA.2017.8229897

[34] J. West, and M. Bhattacharya, Some Experimental Issues in Financial Fraud Mining. Procedia Computer Science, vol. 80, no. 1, pp. 1734–1744, 2016.

Mr. Nhlakanipho M. Mqadi was born in Durban, South Africa. He is currently a student towards a master's degree in Information and Communication Technology at the Durban University of Technology (DUT). Has obtained a Bachelor of Technology degree in Information Technology (IT) (*cum laude*) and a 3 year National Diploma in IT, both from the Durban University of Technology. He is a member of the Golden Key International Honour Society.

Dr N. Naicker education background is as follows: PhD [Information Systems & Technology]; MSc [Information Systems]; Hons BSc (Computer Science); BSc (Computer Science). He currently serves as head of the Information Systems Department at the Durban University of Technology. He is currently involved with the supervision of PhD and Masters students at the Department of Information Systems. He is a member of the ICT and Society Research Group for the Faculty of Accounting and Informatics at Durban University of Technology.

Dr. T. Adeliyi is an academic in the department of Information Technology at the Durban University of Technology. Active researcher in the field of computer science and has research interests in machine learning, digital image processing and intelligent systems. He is a member of the ICT and Society Research Group of Durban University of Technology.