SUNWAY UNIVERSITY

Capstone Project Report

Fraud Detection using Data sampling Methods

by

Muhammad Aizad Bin Mohd. Ridzo
(16053936)

BSc (Hons) Information Technology (Computer Networking and Security)

Supervisor : Assoc. Prof. Dr Lau Sian Lun

Date : 14 November, 2020

Project title : Fraud Detection using Data Sampling Methods
Date : 13/11/2020
Student : Muhammad Aizad bin Mohd. Ridzo
Supervisor : Assoc. Prof. Dr Lau Sian Lun

Abstract

The events of frauds are growing increasingly day by day and are more targeted towards banks and businesses. A solution to this is the implementation of fraud detection system. However, the datasets are very imbalanced which can lead to misclassifications of transaction due to the lack of information of the minority class. In this paper we will be utilizing the imbalanced dataset, Synthetic Financial Dataset for Fraud Detection from Kaggle to improve the classification accuracy of the minority class by using different sampling methods. Three supervised machine learning algorithms are also used to assess the performance of the classification accuracy across all sampling methods. The classification performance will be measured using precision, recall, F-measure, and area under precision recall curve (PRC) on the same testing dataset. We demonstrate that resampling data, particularly SMOTE will increase the accuracy of the classification on the minority class across all classifiers.

Table of Contents

List of Figures

List of Tables

1 Introduction

Fraud is defined as an illegal activity which uses criminal deception that is intended for personal gain. It can be carried out in different ways and each method is different from the other. The main victim of fraud are businesses rather than individuals due to the higher reward. To prevent this unlawful act, many of the businesses have implemented their own fraud detection system [1].

However, there has been little progress over the years in the development of fraud detection system due to the limited exchange of ideas regarding the topic [2]. In addition, the dataset is highly imbalance with genuine data as the majority class and fraud data as the minority class. Due to the imbalance class distribution the system may produce biased and inaccurate readings. Therefore, an effective classification method that can accurately classify the unbalance data for fraud detection is needed [1].

The aim of the research is to improve the accuracy of classification of the minority class by using different sampling techniques. In this paper we will be using three different sampling techniques which includes regular sampling, random under sampling (RUS) and Synthetic Minority Over-Sampling Technique (SMOTE) which will be tested on three classifiers which are Logistic Regression (LR), Random Forest (RF) and Extreme Gradient Boost (XGB). The classifiers will be then evaluated using performance metrices such as precision, recall, F-measure, and the area under the precision recall curve (PRC).

This section of the report consists of the problem statement, project goals, project objective, project scope and expected outcome. The rest of the report consists of three chapters that are organized as follows; Chapter 2 provides literature review on existing studies of fraud detection systems. Chapter 3 presents the experiment outline and the methodology. Chapter 4 discusses the outcomes of the project. Chapter 5 talks about the conclusion and suggestions for future works.

1.1 Problem Statement

The issues of this study are that the discussions regarding the development of fraud detection systems are kept private [2]. It is very unlikely to exchange ideas in an open environment due the information being valuable, and fraudsters might obtain it and use it against the systems. Therefore, banks kept their datasets private and their results censored to avoid any stolen data. As a result, discoveries, and improvements in the study of fraud detection systems are very limited [2] [3].

In addition, the state of the transaction dataset remains to be a problem for the study of fraud detection systems. The number of fraud cases are smaller compared to genuine transactions. For example, approximately 350 million transactions per year is carried by the credit card company Barclays in the UK alone. However, only 0.1% of the transactions are believed to be fraudulent. This causes an imbalance amongst the classes within the dataset. When there is an imbalance in the class distribution, the system may favour the majority class and produce bias results which

may lead to inaccurate predictions and accuracy readings. This is due to the lack of information within the minority class [2].

Finally, outputs such as predictive accuracy cannot be taken as a measurement to define the success of the classifier due to the skewness of the dataset. The minority class in an imbalanced dataset will have a higher cost which can have an influence on the accuracy of the classifier during the model training phase. To increase accuracy, the classifier will attempt to minimize the error rate and provide a predictive generalization towards the classes [4]. For example, a system is regarded as effective if achieve a score of 99% on both detection of genuine and fraudulent records. However, for every 1000 records only one record is fraudulent, which means on average the system will flag fraudulent for 100 records however only 9 are fraudulent. This can be resulted in misclassification of data where classifying genuine data as fraud and frauds as genuine [2].

1.2 Project Goals

The goal of this project is to improve the accuracy of classification of the minority class by using different sampling techniques on a highly imbalanced dataset.

1.3 Project Objective

The main objective of this study is:

- To develop a fraud detection system which classifies transactions according to their classes by using sampling techniques.
- To increase the performance metric scores of precision, recall, F-measure, and PRC for the classifiers.
- To plot the scores of precision and recall on the PRC curve for each of the classifiers.
- To perform comparative analysis between the classifiers on different sampling methods.

1.4 Project Scope

The dataset that we will be utilizing in this project is highly imbalanced where the majority class is genuine, and the minority class is fraudulent. Therefore, we will be deploying resampling techniques to balance the classes to improve the accuracy of the classification of the minority class. To evaluate, different classifiers are used on different sampling techniques to perform evaluation.

1.5 Expected Outcome

At the end of the project, the expected outcome would be a fraud detection system where it can classify both majority class (genuine) and minority class (fraudulent) accurately.

2 Literature review

In this section we will provide a literature review of previous research on fraud detection systems. This section aims to study the different methods that have been used by previous authors.

2.1 Fraud Detection Systems

Credit Card frauds have been plaguing the industry for many years. It has caused billions of dollars of losses and has become a main problem to businesses and financial institutions [5]. The definition of credit card frauds according to Bhatla et al. [6] is the usage of another individual's credit card for personal gain without the knowledge of the card owner and the issuer.

To restrict fraudulent transaction companies and banks have implemented fraud detection systems into their network. Fraud detection is defined by Bolton & Hand [2] as a method that is used to capture fraudulent activities as soon as it happens by identifying it. Not to be confused with fraud prevention which is used to stop the criminals from committing fraud in the beginning. Although fraud prevention mechanisms exist it has not been able to give a similar impact as to fraud detection systems.

This is due to the compromises in security that have been made by companies to make it more usable for customers. Fraud detection systems on the other hand, are built to detect fraud once the pattern is traced [2]. Therefore, fraud detection systems are here to stay [3].

There are two types of fraud detection methods: Supervised Methods and Unsupervised Methods [2].

2.1.1 Supervised Methods

According to Carcillo et al. [7] Supervised Methods are based on binary classifications. Records of fraudulent and genuine transactions logs are used as models [2]. For example, when a transaction is processed, the outcome is compared to either one of the models. The transaction will be then classified based on their similarities [5]. The accuracy of the method is dependent on how well each model is constructed. It is also noted that Supervised Methods could only detect frauds that have occurred from the past [2]. Examples of fraud detection algorithms that use Supervised Methods are Logistic Regression, Random Forest Tree and Gradient Boosted Trees [8].

2.1.2 Unsupervised Methods

On the other hand, Unsupervised Methods does not use any techniques of classification as before. Instead, it uses a combination of two techniques: profiling and outlier detection [8]. According to [2], the outlier detection tool was used by data analysts to detect any falsified data. It could also be used to detect any foreign patterns that appear in the database. The model of this method is based on the general behavior of customers [8]. Any traces of different patterns from the norm will be detected and further examined [2].

2.2 Related Works

In the study of fraud detection systems, a common approach is resampling of the class distribution. In most cases, the amount of fraudulent transactions within a financial dataset is very small. Therefore, the dataset is highly imbalanced and without using the proper methods, it will be difficult to determine the accuracy of the algorithm. By applying resampling techniques such as under sampling the majority class or oversampling the minority class, an accurate reading of the model can be achieved [9].

2.2.1 Credit Card Fraud Detection System

Authors in [10] has built a credit card fraud detection system which classifies transactions as frauds and non-frauds. Resampling methods such as Random Under Sampling (RUS), Random Over Sampling (ROS) and Synthetic Minority Oversampling (SMOTE) technique was applied onto the dataset to improve the classification accuracy due to the highly imbalanced dataset.

Different classifiers such as Naïve Bayes, Linear Regression, Random Forest, and Multilayer Perceptron are used on the different sampling methods for model evaluation of the sampling technique. Performance metrics such as sensitivity, specificity, accuracy, precision, area under curve (AUC) and error rate was used in evaluating the performance of the model.

From the result of the experiment, the Random Forest classifier displayed excellent performance across all three sampling methods. It also has successfully recorded the highest accuracy score compared to the rest of the classifiers tested in this experiment. Apart from that, classifiers that are trained with ROS sampled data has performed very well with AUC scores of at least 0.99 [10].

2.2.2 Provider Claims Fraud Detection System

A study in [4] used different sampling methods to detect provider claims fraud on a highly imbalanced Medicare dataset where only 0.062% of the data is labelled fraud. The study applies six sampling methods which are Random Under Sampling (RUS), Random Oversampling (ROS), Synthetic Minority Oversampling Technique (SMOTE), Borderline SMOTE and Adaptive Synthetic (ADASYN) onto the dataset.

In addition, different classifiers are used in this experiment such as Logistic Regression, Random Forest Trees and Gradient Boosted Trees onto different sampling method for the model evaluation of the sampling technique. Area Under the Receiver Operating Characteristics Curve (ROC) is used in evaluating the model performance on the different sampling methods.

From the results, it shows that RUS performs well across all the classifiers with ROC scores at least at 0.8 and SMOTE with Logistic Regression recorded the highest score of AUC of 0.82 amongst the rest. Therefore, by applying sampling methods onto an imbalanced dataset, the accuracy of the classifiers can increase significantly [4].

2.2.3 Insurance Fraud Detection System

The authors in [1] has developed an insurance fraud detection system that was conducted on an imbalanced dataset. To overcome the imbalance class distribution, the authors has suggested the use of SMOTE to oversample the minority class. In addition to the sampling technique, they have also implemented extreme outlier detection using k Reverse Nearest Neighbors (kRNNs) approach. The outlier detection was intended
to remove any outliers within the minority class to increase the detection rate.

To evaluate the effectiveness of the detection system, the outlier detection system is combined with SMOTE and compared to SMOTE alone. In this comparison values of True Positive (fraud catching rate) and True Negative (non-fraud catching rate) are used as performance metrics. In addition, classifiers such as C4.5, Naïve Bayes, k-NN and Radial Basis Function (RBF) networks were also used in the evaluation process by comparing the performance of the model on both methods.

From the results it shows that C4.5 with extreme outlier detection has improved in both fraud and non-fraud catching rate with both scores at least 99%. In addition, C4.5 results indicated a higher detection of fraud with the proposed method against SMOTE alone [1].

Table 1 shows the summary of the previous studies that was discussed in this section.

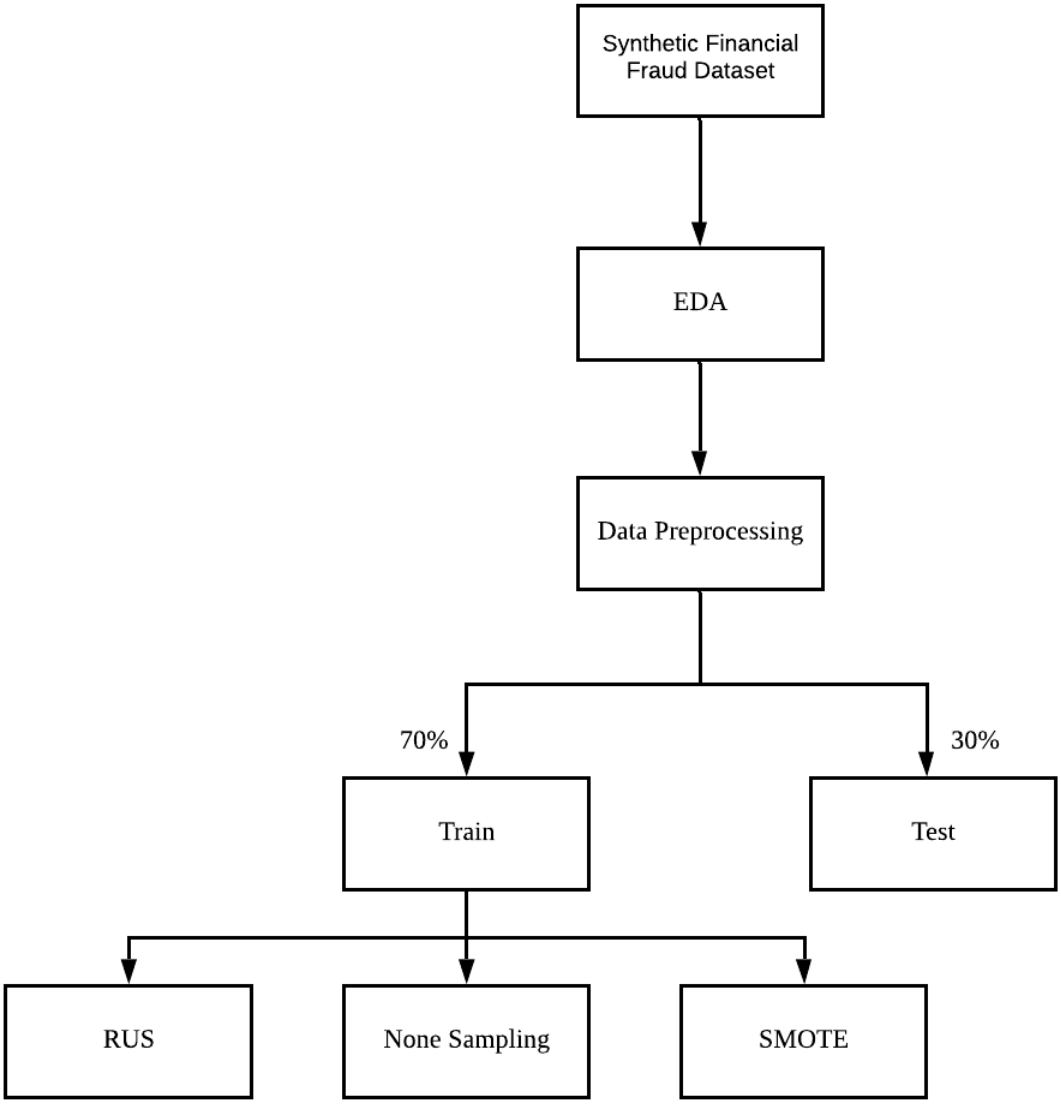
Table 1 Summary of previous studies

Author(s), Year	Type of Detection	Sampling Techniques	Classifiers	Experimental Results
Hodri, Yuhaniz and Azmi, 2018 [10]	Credit Card Frauds	• RUS • ROS • SMOTE	• Naïve Bayes • Linear Regression • Random Forest • Multilayer Perceptron	RF performs very well across all the sampling techniques with scores at least at 0.97. ROS performs slightly better than SMOTE in terms of accuracy and faster speed.
Bauder, Khoshgoftaar, and Hasanin, 2018 [4]	Provider Claim Frauds	• RUS • ROS • SMOTE • Borderline SMOTE ADASYN	• Logistic Regression • Random Forest Trees • Gradient Boosted Trees	SMOTE performs better across all the classifiers with an improved AUC score.
Padmaja, Dhuipalla, Bapi and Krishna, 2007 [1]	Insurance Frauds	• SMOTE • Extreme outlier elimination	• C4.5 • Naïve Bayes • k-NN • Radial Basis Function (RBF) networks	SMOTE with Extreme outlier elimination using C.5 has a higher True Positive and True Negative rate compared to SMOTE alone

3 Methodology

In this section will be discussing the methodology of the project. The following figure is a plan of the proposed fraud detection system.

Figure 1 Proposed fraud detection system



3.1 Dataset Retrieval

The data that we have acquired for this experiment is the Synthetic Financial Fraud Dataset from Kaggle. As mentioned previously, financial datasets are kept confidential by banks and are difficult to obtain due to its information policy that prevents exposing customer's private details. Therefore, a group of researchers have created a synthetic dataset that was generated through a simulator called Paysim as a solution. Paysim generates synthetic transactions by using aggregated data from private datasets. The result is a synthetic dataset that closely resembles existing transactions operations and is optimal for the study of fraud detection systems [11].

3.1.1 Exploratory Data Analysis (EDA)

The libraries that we will be using in this process are Pandas, Matplotlib and Seaborn. The purpose of EDA is to provide better understanding regarding content within the dataset. In this section we have uncovered that:

- There are 8213 transactions that are label ‘isFraud’ and only 16 label ‘isFlaggedFraud’ within the dataset.
- Transactions that are fraud occurs in the transaction type of ‘CASH_OUT’ and ‘TRANSFER’.
- Transactions that are flagged occurs in the transaction type of ‘TRANSFER’.
- There are empty values in columns ‘newbalanceOrig’, ‘oldbalanceOrg’, ‘newbalanceDest’ and ‘oldbalanceDest’ which are represented by ‘0’.

From the observation above we can conclude that the dataset is highly imbalance. Although the label ‘isFlaggedFraud’ exist, we will be dropping the column due low occurrence within the dataset. Finally, there are no empty or negative values, however there are some errors regarding the calculations in the columns for ‘newbalanceOrig’, ‘oldbalanceOrg’, ‘oldbalanceDest’ and ‘newbalanceDest’.

3.2 Data Cleaning and Data pre-processing

3.2.1 Data Cleaning

In this section we conduct cleaning of the contents of data based on the EDA.
Based on the calculation error mentioned in the previous section, we have solved the issue by creating two new columns which are ‘errorBalanceOrg’ and ‘errorBalanceDest'. Inside of the two columns the corrected values will be inputted by using the calculations below:

1. $errorBalanceOrg = newbalanceOrig + amount - oldbalanceOrg$
2. $errorBalanceDest = oldbalanceDest + amount - newbalanceDest$

Finally, columns such as ‘step’, 'isFlaggedFraud', 'nameDest' and 'nameOrig' will be removed because it does not contribute to the final output.

3.2.2 Standard Scaler

Standard scaler is downloaded from the scikit-learn python library. It is used to scale each feature column to a zero mean and a standardization of one. Standard scaler is useful in the case of classifications of frauds where it arranges the data in a standard normal distribution which will provide better result. This method is applied on 'amount', 'oldbalanceOrg', 'newbalanceOrig',' oldbalanceDest', 'newbalanceDest', 'errorBalanceOrg' and ‘errorBalanceDest'.

3.2.3 One Hot Encoding

Get dummies is downloaded from the Pandas python library. The method, Get Dummies or one hot encoding provides binary encoding to categorical columns for better classification. It converts the selected columns to binary columns (1 or 0) to indicate whether it is active or inactive [12]. In this study, one hot encoding is applied to the column 'type' for 'CASH-IN', 'CASH-OUT', 'DEBIT', 'PAYMENT' and 'TRANSFER'.

3.2.4 Train/Test Split

In this experiment we have split the data into testing set of 30% and training set of 70%. The training set will be used to train the classifiers and build models on the different sampling methods whereas the test set will be used for evaluation of the classifier model.

3.3 Sampling Techniques

Sampling techniques allows the rebalance of class distribution to remove the effects of imbalance data and at the same time improve the accuracy of the classifiers to classify data [10]. There are two types of sampling techniques which are under sampling and oversampling. In this study we will be implementing both types of methods by using Random Under sampling (RUS) and SMOTE. Both techniques are retrieved from imblearn python library.

The simplest yet effective method to handle imbalance data is to perform under sampling of the majority class. RUS is an under-sampling technique where it removes data samples randomly from the majority class until it is balanced. This also improves the runtime of the algorithms due to the removal of data [10].

Oversampling is a technique where it increases the samples within the minority class dataset until the data is balanced. SMOTE is a type of oversampling technique where it generates synthetic data samples instead of copying existing samples within the class [4]. It uses a complex method which utilizes K-Nearest Neighbor (KNN) technique to extrapolate pre-existing minority to create new samples [10]. It has been highly regarded by past researchers based on its effectiveness in improving the accuracy of classification of the minority data [1] [10] [4].

Table 2 shows the amount of majority class and minority class before and after applied the sampling techniques.

Table 2 Data Resampling

Sampling Techniques	Number of Majority Class	Number of Minority Class
No sampling	6344407	8213
RUS	8213	8213
SMOTE	6344407	6344407

3.4 Classifiers

After the resampling process, the resampled data is trained by the classifiers using the training set to evaluate the different techniques. The task of classifying data transaction to fraudulent (1) or genuine (0) is a binary classification task. Therefore, in this experiment we will be deploying three different classification algorithms which are Logistic Regression (LR), Random Forest (RF) and Extreme Gradient Boost (XGB) that have been imported from the both sci-kit learn and xgboost libraries.

LR is a statistical regression classifier that is mainly used to perform binary classification predictions. It is very famous amongst researchers because of its simplicity and less intensive computational demand compared to other algorithms. LR is easy to train and commonly used as a performance benchmark for other complex classifiers [13].

RF is regarded as one of the popular classifiers amongst researchers due to its ability of performing both classification and regression. It is known to be very efficient and produced convincing results most of the time. This is due to its large collection of decorrelated decision trees which are trained by using the bagging method [14].

XGB is an ensemble decision tree method which uses gradient boosting framework. One of the highest regarded features of XGB is the ability to perform parallel processing which can increase the speed of the model computation process. It also has a regularization feature where it prevents over-fitting [15].

3.5 Performance Evaluation

The performance evaluation will be applied on the output model of the classifier. This is to measure the accuracy of classification for each of the sampling technique. As discussed in the previous section, we will be using precision, recall, F-measure, and PRC score as the performance metric. The metrices can be retrieved from sci-kit learn python packages.

The performance metric of precision, recall, and F-measure can be retrieved from the confusion matrix table. The table consists of True Positives (TP), True Negatives (TN), False Positives (FP) and False Negative (FN) which are used for further calculations to obtain the metrics in the above [5]. Table 3 shows the confusion matrix table along with the description of each values.

Table 3 Confusion Metrics

Metrics	Description
True Positive (TP)	Fraudulent transactions that are correctly predicted by the system as fraudulent.
True Negative (TN)	Genuine transactions that are correctly predicted by the system as genuine.
False Positive (FP)	Fraudulent transactions that are wrongly predicted by the system as genuine.
False Negative (FN)	Genuine transactions that are wrongly predicted by the system as fraudulent.

Performance metric calculations:

1. Precision: $\frac{TP}{TP+FP}$
2. Recall: $\frac{TP}{TP+FN}$
3. F-measure: $\frac{2*Precision*Recall}{Precision+Recall}$

By retrieving the values above, PRC can be calculated. PRC is famously used to evaluate the performance of the binary classifications on an imbalance dataset. It measures values of precision and recall and shows the trade-off between the two values for different thresholds. A high value of area under the curve suggests that both values of recall and precision are high. High recall indicates low false positive rate whereas high precision indicates low false negative rate [16].

4 Results and Discussion

This section discusses the results of our study. We have applied three different sampling methods which are regular sampling (Non), RUS and SMOTE on different classifiers on the testing dataset. Measurements such as PRC, precision, recall and F-measure are taken to evaluate the classification performance of the models on the different sampling methods. The results are shown below in Figure 2-7 and Table 4.

Figure 2-7 shows the PRC graphs of different classifiers with different sampling techniques. The graph's x-axis represents the recall score (false positive rate) whereas the y-axis represents the precision score (false negative rate).

Figure 2

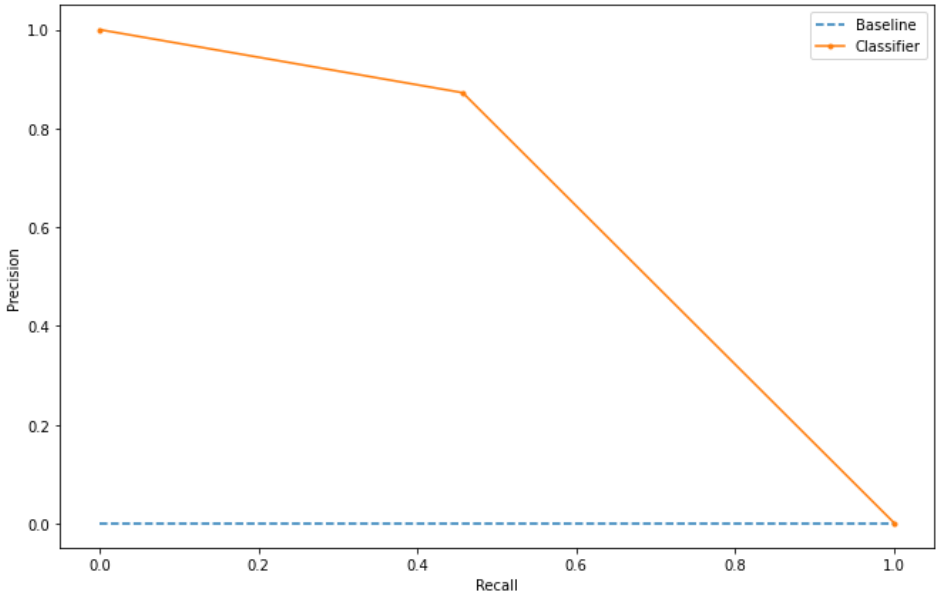


Figure 2 shows the PRC graph of Logistic Regression with regular sampling.

Figure 3

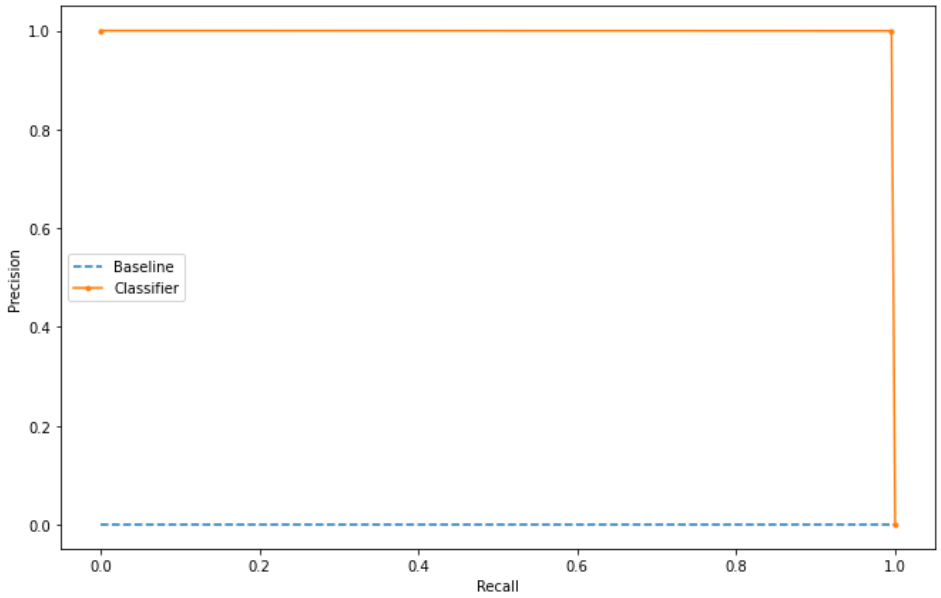


Figure 3 shows the PRC graph of Random Forest with regular sampling.

Figure 4

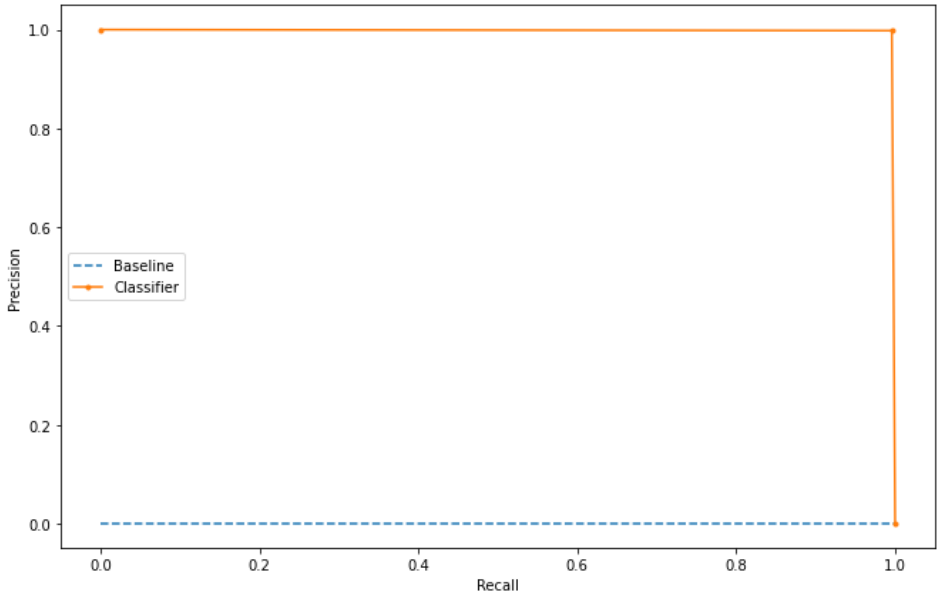


Figure 4 shows the PRC graph of XGBoost with regular sampling.

Figure 5

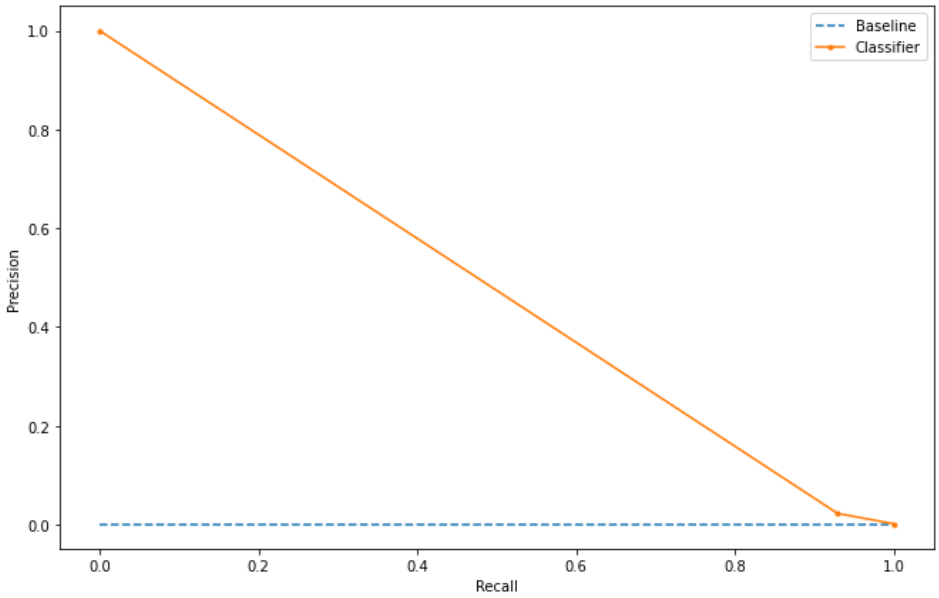


Figure 5 shows the PRC graph of Logistic Regression with Random Under sampling.

Figure 6

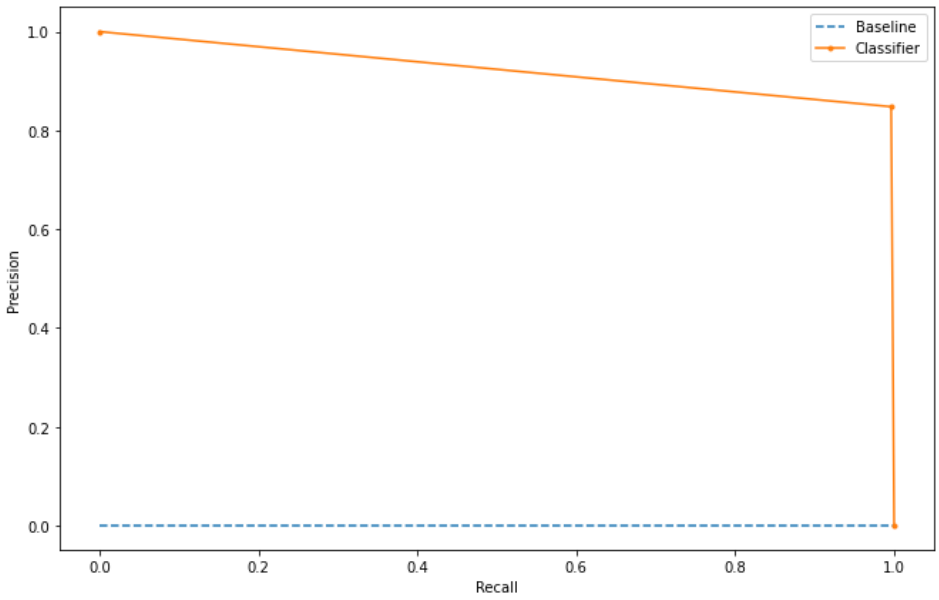


Figure 6 shows the PRC graph of Random Forest with Random Under sampling.

Figure 7

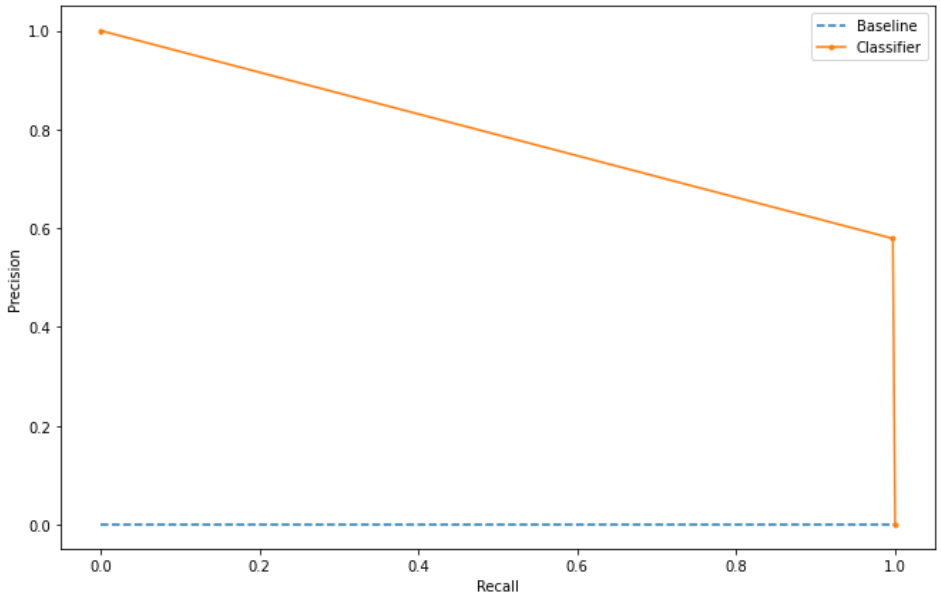


Figure 7 shows the PRC graph of XGBoost with Random Under sampling.

Figure 8

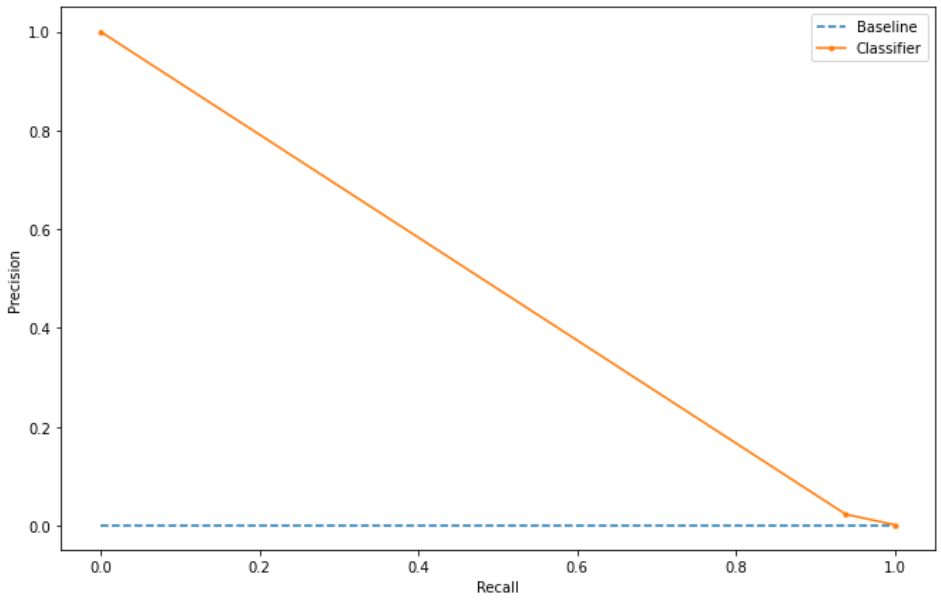


Figure 8 shows the PRC graph of Logistic Regression with SMOTE.

Figure 9

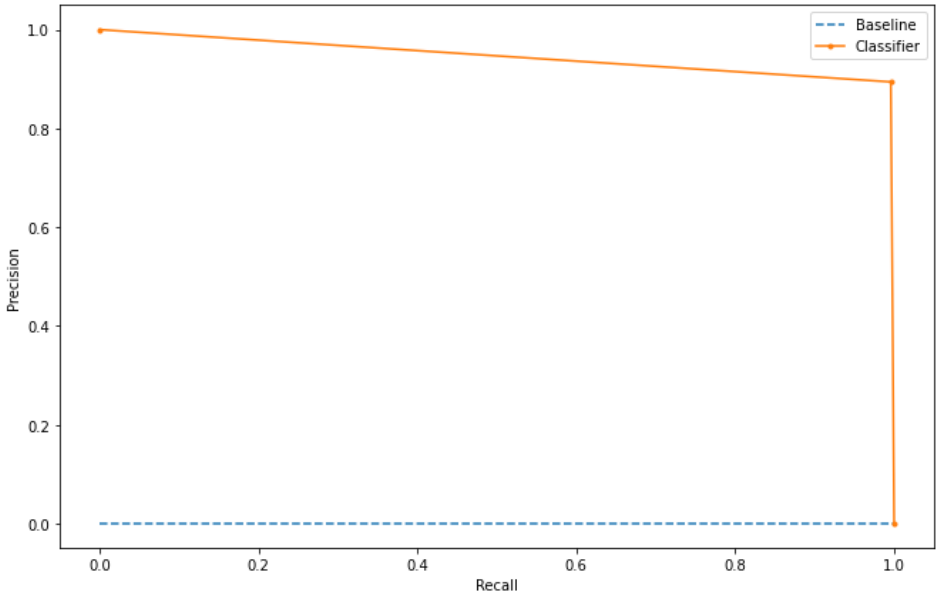


Figure 9 shows the PRC graph of Random Forest with SMOTE.

Figure 10

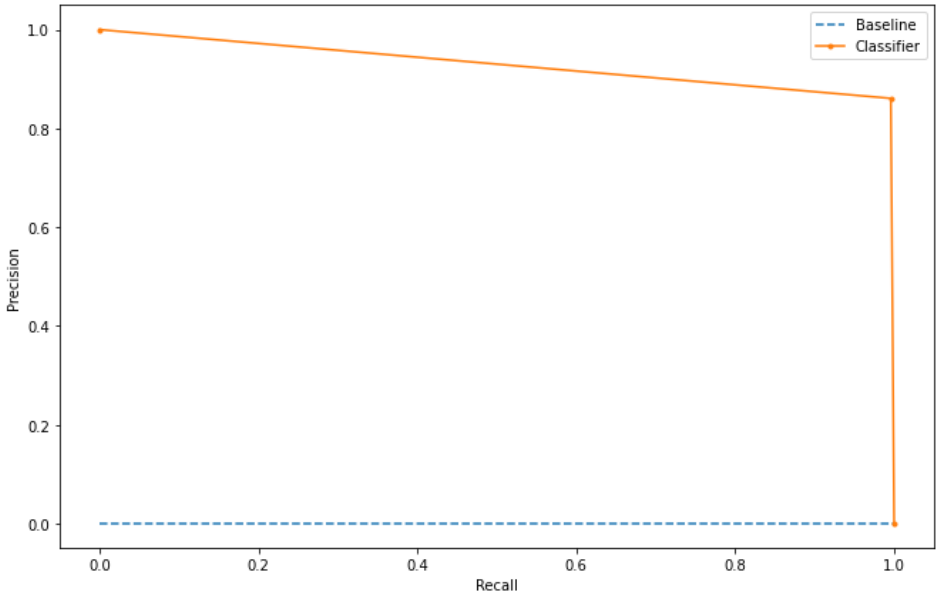


Figure 10 shows the PRC graph of XGBoost with SMOTE.

Table 4 Fraud detection results

		Performance Metric			
Classifiers	Sampling	Precision	Recall	F-measure	PRC
LR	Non	0.87	0.46	0.60	0.63
	RUS	0.02	0.93	0.04	0.55
	SMOTE	0.02	0.94	0.04	0.55
RF	Non	1.0	1.0	1.0	1.0
	RUS	0.85	1.0	0.92	1.0
	SMOTE	0.89	1.0	0.94	0.99
XGB	Non	1.0	1.0	1.0	1.0
	RUS	0.58	1.0	0.73	0.99
	SMOTE	0.86	1.0	0.92	1.0

As seen in Figure 2-7 and Table 4, the results of XGB and RF performed very well with perfect results with no sampling applied. On the other hand, LR performed poorer compared to XGB and RF with a PRC score of 0.63. An imbalanced dataset can have an influence over the accuracy of the model due to the lack of information in the minority class (fraud). Therefore, to compensate the classifier will have to make assumptions based on the majority class (genuine). Despite the convincing classifiers results as seen on the sampling of Non, it will be withdrawn from any further comparison due to its biased results.

Overall observations that can be obtained from the results is that SMOTE performs better than RUS. In addition, results from Figure 7 and 10 suggested that XGB runs significantly better on SMOTE rather than RUS. This can also be seen in table 4 where a significant increase in both precision and F-measure scores from 0.58 and 0.73 to 0.86 and 0.92. For RF both precision and F-measure score increase slightly from RUS to SMOTE from 0.85 and 0.92 to 0.89 and 0.94. However, LR performs the worst amongst the rest. In maintains its values as well as the shape of the curve for both sampling technique with the lowest score of precision at 0.02.

SMOTE showed greater performance over RUS in terms of precision and F-measure. The reason of the poor performance of classifiers when using RUS is because of the removal of data. RUS under samples the data by randomly removing samples from the majority class to balance the class distribution. By removing data, it eliminates crucial information that could contribute to the classifier's performance in the classification of transactions [4]. Therefore, RUS underperforms especially when tested on a larger data sample.

5 Conclusion and Future Work

5.1 Conclusion

In this paper we have focused on the detection of fraud within a Synthetic Financial Fraud Dataset which suffers from severe data imbalance. An imbalance of data occurs when there is an unequal distribution between the classes within the dataset. To solve the issue, we introduce two sampling techniques (RUS and SMOTE) on three different supervised machine learning classifiers (LR, RF and XGB). To evaluate the performance of the techniques, we adopted a different evaluation metrics which uses the precision, recall, f-measure, and PRC. In addition, from the values obtained we plotted the scores on the PRC curve. A high PRC score indicates high recall (low false positive rate) as well as high precision (low false negative rate) [16]. In addition, we included regular sampling of data (Non) to indicate biasness of majority class (genuine). We tested the results of all the classifiers on the same test for a fair comparison.

From the experimental result, as predicted classifiers with Non displayed near perfect results despite the imbalanced of data. It is believed that the contribution to their success is the biasness of the algorithm favoring the majority class (genuine) due to the lack of information in the minority class (fraudulent). Both XGB and SMOTE performed as expected where it returns an accurate reading on the minority class. SMOTE performed well across all the classifiers with XGB being the highest amongst the rest. This is due to the oversampling technique which utilizes KNN that delivers an improvement of the classification accuracy on the minority class [10]. In contrast, RUS performed slightly worse than SMOTE which can be seen in scores such as precision and F-measure. However, LR being the lowest amongst the rest where it maintains the same performance scores despite the change of sampling method. Therefore, SMOTE is regarded as the suitable sampling method in this dataset due to its improvement of accuracy in terms of precision and recall scores.

5.2 Further work

Fraud is an ever-growing problem therefore the study should not end here. Fraudsters may find new ways to cheat the system by exploiting its weaknesses. In addition, the methods that are discussed might not performed similarly when tested on different datasets. Therefore, we suggest introducing new classifiers such as Artificial Neural Network and Bayesian Network.

In addition, further exploration on different sampling techniques is highly recommended. Transaction data in most cases are highly imbalanced therefore manipulation of data through sampling is needed to obtain accurate results. Recommendation of new sampling methods to be implemented are Random Oversampling and Adaptive Synthetic (ADASYN). By deploying more methods, it will provide the system with more variety of input to evaluate which is key for development in this field of study.

6 References

[1] T. M. Padmaja, N. Dhulipalla, R. S. Bapi, and P. R. Krishna, "Unbalanced data classification using extreme outlier elimination and sampling techniques for fraud detection," in *15th International Conference on Advanced Computing and Communications (ADCOM 2007)*, Guwahati, Assam, India, Dec. 2007, pp. 511–516, doi: 10.1109/ADCOM.2007.74.

[2] R. J. Bolton and D. J. Hand, "Statistical Fraud Detection: A Review," p. 54.

[3] B. Wiese and C. Omlin, "Credit Card Transactions, Fraud Detection, and Machine Learning: Modelling Time with LSTM Recurrent Neural Networks," in *Innovations in Neural Information Paradigms and Applications*, M. Bianchini, M. Maggini, F. Scarselli, and L. C. Jain, Eds. Berlin, Heidelberg: Springer, 2009, pp. 231–268.

[4] R. A. Bauder, T. M. Khoshgoftaar, and T. Hasanin, "Data Sampling Approaches with Severely Imbalanced Big Data for Medicare Fraud Detection," in *2018 IEEE 30th International Conference on Tools with Artificial Intelligence (ICTAI)*, Volos, Greece, Nov. 2018, pp. 137–142, doi: 10.1109/ICTAI.2018.00030.

[5] Y. Jain and S. Jain, "A Comparative Analysis of Various Credit Card Fraud Detection Techniques," vol. 7, no. 5, p. 6, 2019.

[6] T. P. Bhatla, V. Prabhu, and A. Dua, "Understanding Credit Card Frauds," p. 17.

[7] F. Carcillo, A. Dal Pozzolo, Y.-A. Le Borgne, O. Caelen, Y. Mazzer, and G. Bontempi, "SCARFF: A scalable framework for streaming credit card fraud detection with spark," *Inf. Fusion*, vol. 41, pp. 182–194, May 2018, doi: 10.1016/j.inffus.2017.09.005.

[8] N. Laleh and M. Abdollahi Azgomi, "A Taxonomy of Frauds and Fraud Detection Techniques," in *Information Systems, Technology and Management*, Berlin, Heidelberg, 2009, pp. 256–267, doi: 10.1007/978-3-642-00405-6_28.

[9] C. Phua, D. Alahakoon, and V. Lee, "Minority Report in Fraud Detection: Classification of Skewed Data," p. 10.

[10] N. F. Hordri, S. Sophiayati, N. Firdaus, and S. Mariyam, "Handling Class Imbalance in Credit Card Fraud using Resampling Methods," *Int. J. Adv. Comput. Sci. Appl.*, vol. 9, no. 11, 2018, doi: 10.14569/IJACSA.2018.091155.

[11] E. A. Lopez-Rojas, S. Axelsson, and D. Baca, "Analysis of Fraud Controls Using the PaySim Financial Simulator," p. 10.

[12] D. (DJ) Sarkar, "Categorical Data," *Medium*, Mar. 27, 2019. https://towardsdatascience.com/understanding-feature-engineering-part-2-categorical-data-f54324193e63 (accessed Nov. 14, 2020).

[13] "The Logistic Regression Algorithm – machinelearning-blog.com." https://machinelearning-blog.com/2018/04/23/logistic-regression-101/ (accessed Nov. 13, 2020).

[14] "The Random Forest Algorithm: A Complete Guide | Built In." https://builtin.com/data-science/random-forest-algorithm (accessed Nov. 13, 2020).

[15] S. Mishra, "Handling Imbalanced Data: SMOTE vs. Random Undersampling," vol. 04, no. 08, p. 4, 2017.

[16] "Precision-Recall — scikit-learn 0.23.2 documentation." https://scikit-learn.org/stable/auto_examples/model_selection/plot_precision_recall.html (accessed Nov. 13, 2020).