Online Fraud Detection Using LGBM and Random Forest

Github Link : https://github.com/krutikapimple6/fraud-detection-with-LGB-and-RF

Krutika Kumar Pimple *20210459*
Shivani Bhat *20211408*
Sruthi Nair *20210531*
Vinay Jagdish Jethmalani *20210984*

***Abstract*—E-commerce has been on the exponential rise lately. The Covid pandemic has only accelerated the process of changing user behaviour of shopping towards online methods. One of the biggest challenges in processing such large number of transactions is the Credit Card Frauds. The major hurdle in detecting such transactions is the unavailability of sensitive or critical user information for analysis. Along with that, datasets are largely imbalanced. By handling imbalanced dataset effectively, we can reduce the false alarm at organizations and increase the rate of true positives. This paper focuses on one such skewed dataset that is handled by implementing oversampling of the minority class (Fraud = 1). Post this method, dataset is then subjected to LGBM classification model with K folds Cross Validation. Then Grid Search Cross Validation was performed for both Random Forest and LGBM classifiers. To judge the efficiency of this method, we have implemented the Area Under Curve (AUC) performance metric of ROC and PR Curve.**

***Index Terms*—fraud, imbalanced, oversampling, lgbm, k-folds, auc, random forest, precision, recall, ROC, Grid search, cross validation**

I. Introduction

The rate of fraud occurrence is increasing dramatically with the augmentation of technology and different communication pathways resulting in the loss of billions of dollars globally. Detection of fraud becomes a very important piece of work and with the advancement of human knowledge, statistics and machine learning provide effective technologies for fraud detection which have been applied successfully to detect activities such as money laundering and e-commerce credit card fraud. Fraud detection is a concept that accommodates different categories but we will focus on online credit card fraud detection. Herein,in this paper we performed a comparative analysis on LGBM and Random Forest to find the best suitable model for fraud detection. The feature engineering including data cleaning, label encoding and handling class imbalance is introduced in Data Mining Methodology section. Further, LGMB and RF based classifier models are discussed. The evaluation metrics used show the superiority of our models and is talked about in the Evaluation and Result section. Finally,the last section draws a conclusion of this paper and describes future scope.

II. Related Work

Fraud detection system functions on identifying general trends of suspicious transactions. Majority of the fraud detection systems use supervised learning to determine the target variable(is Fraud).Various evaluation and performance metrics are incorporated in the evaluation of fraudulent transactions. Some use Receiver Operating Characteristic (ROC) analysis (true positive rate versus false positive rate) while many use Area Under the Curve (AUC). In fraud detection, misclassification costs (false positive and false negative error costs) are unequal and uncertain differing from example to example, and can change over time. But its the false negative error that is usually more costly than a false positive error.When we consider fraudulent data, the volume of both fraud and legal classes will fluctuate independently of each other and it becomes very important to handle this ”class imbalance” issue as seen in [2]. In yet another research [3], work has been conducted on different techniques to overcome class imbalance such as Random oversampling, SOMO and SMOTE, through different classifiers and evaluation metrics.Most of the classifiers (Regression models, Support Vector Machine (SVM), Decision trees, or Neural Networks) present a poor performance when they are facing unbalanced data: they can have good accuracy for the majority class but poor results for the class of interest (in this case fraud class).This is further explained in [4] where detecting fraud in such a highly imbalanced data set is a huge challenge which could typically lead to predictions that favor the majority group, causing fraud to remain undetected. One such favourable sampling technique is the generation of synthetic instances which have been proven to be very successful in various business applications (e.g. credit scoring, churn prediction, and fraud detection). These methods add new information to the original data set by creating extra synthetic minority class samples based on the existing minority samples that are available in the data set. We have worked on over sampling our data set to control the issue of class imbalance

The common three issues of the real-world fraud datasets such as imbalanced data, non-availability of original features and pattern sparsity of fraud instances which are addressed in [5]. The study executes combinations of data

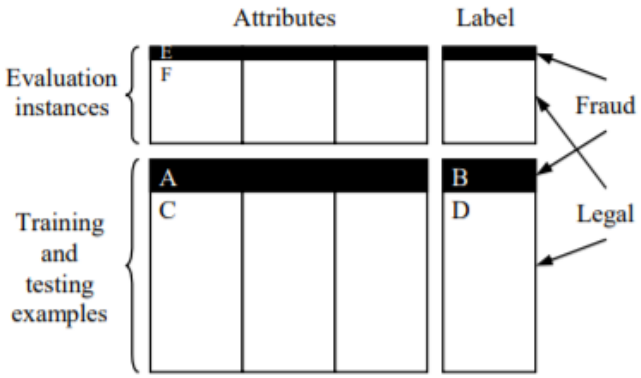

Fig. 1. Data analysis structure (taken from [2])

balancing techniques along with ensemble models such as Random Forest Classifier (RF), Extreme Gradient Boosting (XGBoost), Light Gradient Boosting (LGBoost). The results demonstrate that Random Forest performed the best predictive results when used with the SVM SMOTE technique.Similarly, [6] highlights on the importance of handling of unbalanced datasets. It implements a number of machine learning models like Logistic Regression, Random Forest Classifier and KNN on original unbalanced dataset and resampled datasets. Initially, unbalanced data is provided as input to machine learning models which results in the rise of false alarming on the genuine transactions. When data is resampled (Undersampling, Oversampling or SMOTE) accuracy of the model increases and also higher AUC scores as high as 0.91 and low false alarming rates. It is seen that most of the papers preferred oversampling techniques to tackle class imbalance but [7] experimented with a combination of Undersampling and oversampling techniques. Authors combined RUS (Random undersampling techniques) with oversampling techniques from SMOTE family (SMOTE, ADASYN, Borderline, SVM-SMOTE and ROS) to find the best results. The performance of all sampling combinations is determined after applying the Random forest classification model.They observed that individual sampling techniques didn't give high performance except for ROS.

Another important aspect of the accurately detecting fraud detection is the model selection. We can select from the Tree based framework such as LGB Model or XGB or traditional machine learning models like Logistic Regression or KNN. [8] chooses LGB Model over the XGB and the rest of models due to its ability to exclude features with small gradients (Gradient-based One-Side Sampling - GOSS) and feature bundling process (Exclusive Feature Bundling - EFB). These capabilities help to increase speed and accuracy of the whole system. Whereas in [9], authors compared data mining models like SVM, logistic Regression, Random Forest and XGBoost. AUC ROC score of XGBoost outperformed other models. SMOTE algorithm helped to overcome the concern of imbalance class. 5-fold RFECV (recursive feature elimination with cross-validation) method was used to select the top 100 important feature on the Xgboost model.

[10] proposed an intelligent approach for detecting fraud in credit card transactions by using an optimised Light Gradient Boosting Machine, which outperformed other ML algorithms including the linear Support Vector Machine, Random Forest and k -Nearest Neighbors. This approach achieved highest performance with the best Precision and AUC scores amongst other metrics. The study also highlights the effectiveness of the use of 5-fold Cross-Validation, that uses each fold for testing and training processes in reliably evaluating the model performance. [11] studied and compared three ways of handling unbalanced datasets – resampling methods (Undersampling and Oversampling), cost-sensitive training and tree algorithms. It was observed that oversampling with SMOTE techniques produced the best performance metrics. This technique can be easily implemented and has the added benefit of being independent of the underlying classifier. Additionally the study demonstrated that the AUC of the ROC curve isn't a good metric of the classifier performance for unbalanced datasets and that the AUC of PR curve gives an appropriate measure of the true output in such datasets. This is because the False Positive Rate does not decrease drastically when the total number of negative cases is huge. On the contrary, precision score is highly sensitive to false positives.

In our study, we will be using the oversampling technique to feed data to the model. Although, as per previous literature, it has been observed that oversampling using SMOTE is highly effective in dealing class imbalance, it has also been noticed that it tends to introduce noise while generating synthetic samples [12]. Additionally, it does not seem to be very practical for high-dimensional data.

III. Data Mining Methodology

The process of data mining plays a major role in decision making to analyse trends, behaviour and further extract hidden patterns. It becomes vital to turn raw data into useful insights. One such process to achieve this is **CRISP-DM** (CRoss Industry Standard Process for Data Mining).It is a process model with six phases that naturally describes the data science life cycle described as follows

A. Business Understanding

Fraud prevention system, though cumbersome, saves consumers millions of dollars per year. Researchers from the IEEE Computational Intelligence Society (IEEE-CIS) are partnering with the world's leading payment service company, Vesta Corporation, seeking the best solutions for fraud prevention industry. This process of business understanding emphasises on defining and framing the business problem. This research aims in identifying fraudulent transactions involving a wide range of features from device type to product features. (Select technologies and tools and define detailed plans for each project phase.)

B. Data Understanding

The main objective for this research if predicting the probability that an online transaction is fraudulent, as denoted

by the binary target is Fraud.The dataset is from IEEE-CIS competition [15] sourced from kaggle.The data set is broken into two files identity and transaction, which are joined by Transaction ID. Not all transactions have corresponding identity information.Few categorical features in the transaction table include ProductCD, card1 - card6, addr1, addr2, ,P_emaildomain, R_emaildomain and M1 - M9 whereas the ones in identity table are DeviceType, DeviceInfo, id_12 - id_38.It is very important to analyse and understand the data, explore data, describe data and verify data quality.The shape of the data set includes over 590k instances with around 434 features. But, the target variable is imbalanced with over 560k non-fraudulent transactions and just 20k fraudulent ones. As discussed, this poses the problem of "class imbalance" which will be discussed further.The data set is a combination of both categorical and discrete values(as seen in Figure 2) and it becomes important to understand the significance of these features. A lot of the data(For ex. credit card details) is masked to avoid leaking critical information.

C. Data Preparation

The dataset consists of 4 main tables transaction train and test tables, identity train and test tables. For convenience of coding, data exploration is done by merging the transaction and identity files on the transaction_id column using left outer join. Random sampling is performed on the merged table to avoid memory issues.

1. Transaction Tables

Variable	Description	Type
isFraud	Binary	Categorical
TransactionDT	transaction date	time
TransactionID	unique transaction ID	ID
TransactionAmt	transaction amount	numerical
addr1-addr6	address	categorical
card1-card6	card	Categorical
Email	Email	Categorical
C1-C14	anonymous features	numerical
D1-D15	anonymous features	numerical
V1-V339	anonymous features	numerical
M1-M9	anonymous features	Categorical

Fig. 2. Transaction Tables

Transaction table has presence of both categorical and numerical features (394 Features). A lot of data is masked which otherwise could give fraudsters critical information to evade detection [1].For efficient feature engineering, each columns correlation is checked with the target variable [isFraud]. The highly correlated features (For example D1 and D2 are heavily correlated; D4 is highly correlated with D10,D11,D15; D3 and D5 are highly correlated; D15 is highly correlated with D10,D11;D10 and D11 are highly correlated) are then removed because it can prevent our model from over fitting. Apart from that, attributes which have high rate of NaN, Null or missing values(For example D2,D5,D6,D7,D8,D9,D12,D13,D14 have more than 80% null values) are also discarded to reduce the memory usage by model. However, the features which were masked due to confidentiality and had lesser null values were imputed with various approaches(Mode/mean/grouped as "others")based on the statistical knowledge of the attribute but a few which were difficult to analyse were retained.

2. Identity Tables

Variable	Description	Type
TransactionID	unique transaction ID	ID
DeviceType	device type	categorical
DeviceInfo	device info	categorical
id01-id11	data for identification	numerical
id12-id38	data for identification	categorical

Fig. 3. Identity Tables

Identity Tables has total of 41 features which includes the Device Information, device type and Transaction Id. Similar to handling of null values in transaction tables, attributes with higher rate of null values are discarded.

3. Label Encoding

Label Encoding is one of data preprocessing step in our implementation. Most machine learning algorithms and frameworks works faster and greater accuracy with numerical input. Label Encoding process will convert our categorical data into numericals. All the attributes present after data cleansing process which are categorical except isFraud are label encoded.

4. Class Imbalance

The problem of solving class-imbalance, as already discussed in the previous sections, needs attention to achieve unbiased results. The instances of fraudulent class has comparatively lesser data than those of non-fraud which can result to incorrect and skewed predictions.

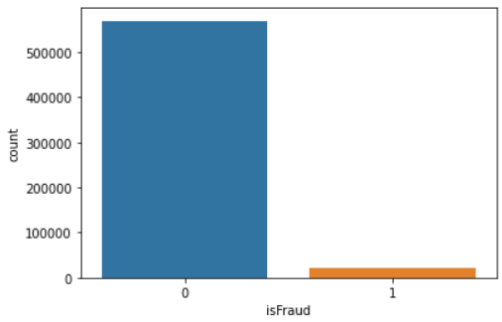

Fig. 4. Imbalanced data

On research, we came across various methods to handle this few being under sampling, Oversampling, and SMOTE. SMOTE has found a lot of advantage over the other techniques and has proven to achieve higher accuracy. But, it does not perform well on high dimensional data(as in our case) and could introduce noise [12].With this background, we moved ahead with oversampling which tries to populate the minority class(Fraud) to match the records of the majority class(non-Fraud) as seen in fig 5.

Fig. 5. Oversampling technique [11]

D. Data Modeling

1. K-Fold Cross Validation

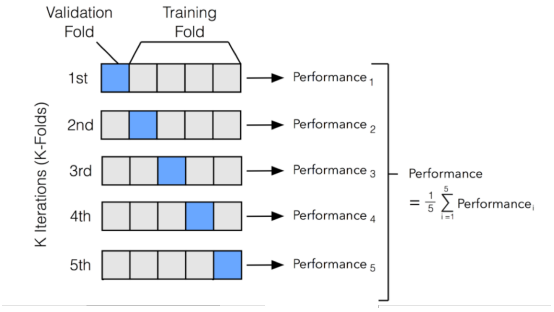

Fig. 6. Overview of Kfolds

Dataset which is under consideration has been collected over the certain time period. TransactionDT column has numerical data which is a reference to a certain timestamp. The value of this feature is a timedelta. Due to this reason, splitting the data into training and testing subsets has been carried out using TimeSeriesSplit function. The dataset is randomly sampled into 'K' number of groups. In our case, the number of Folds has been set to 5.

2. Implementation of Light Gradient Boosting(LGB) Model

In this section, we will briefly discuss about our LGB model and the respective parameters used in it. The LGB is a histogram based algorithm that generates highly complex trees and is scalable when the number of records is high. By making use of Gradient-based One-Side Sampling(GOSS) optimization techniques, it excludes feature value with small gradients. For optimising the parameters after the data was split into training and test sets, we have implemented hyper-parameter tuning using Grid Search Cross Validation with 3 folds (this was chosen to obtain appropriate parameters with reduced computational time, as opposed to 5 folds). We used Grid Search CV as it evaluates all combinations of parameters that we specify unlike random search that samples randomly from that distribution. The maximum depth was chosen as 15 and number of leaves as 256 to ensure sufficient complexity of the tree model, while also restricting the tree depth to avoid over-fitting. The learning rate of 0.01 was too slow and 0.02 was chosen as optimal for convergence.

Parameter	Description of Parameter	Value
n_estimators	count of estimators	5000
Learning_rate	learning rate	0.02
bagging_function	rate of rows	0.8
feature fraction	sample rate (columns)	0.5
max depth	depth of tree	15
boosting_type	Type of boosting	gbdt, goss
min_child_samples	Minimum child samples	79
num_leaves	Number of leaves	256

Fig. 7. LGBM Parameters

3. Implementation of Random Forest

Random Forest classifier is considered as one of the top classifiers which is particularly useful in detecting fraudulent transactions. This involves developing a collection of decision trees by focusing on the records and features of the training data, and classifying records based on the maximum votes for each class. This method minimises over-fitting in decision trees and helps improve the accuracy. However, the process is quite slow, as it combines many trees for determining the class. Another drawback is that the training of this classifier requires complete data with no missing values. Therefore, we have used random sampling imputation for columns with null values so that these values could be replaced with random samples of the respective columns. Furthermore, in order to optimise the model we have performed hyper-parameter tuning with the use of Grid Search Cross Validation with 3 folds, as was used in LGB. This was done after the data was split into training and test sets. Among other parameters, the number of estimators that corresponds to the number of decision trees was set to 50 with the minimum number of data points allowed at each leaf node as 2. Using this method, different hyper-parameters were tested on the training data and after the optimum parameters were selected, the Random Forest Classifier was tested on the test set. This resulted in improved performance.

Parameter	Description of Parameter	Value
n_estimators	count of estimators	50
criterion	Criteria	entropy
max_features	Number of features	auto
min_samples_leaf	sample rate (columns)	2

Fig. 8. Random Forest Parameters

IV. Evaluation and Results

Evaluating results in a data mining problem is a fundamental aspect and can differ between models. The ROC (Receiver Operating characteristic) curve summarises all the confusion matrices generated by different thresholds used by the classifier. The ROC's Area Under the Curve (AUC) could be

utilized as an evaluation metric when there are roughly equal numbers of observations for each class. For highly imbalanced datasets with a large amount of true negatives, the False Positive Rate (FPR) reduces to low values that pushes the ROC towards the left, making the AUC of the ROC nearly 1. This could be misleading as it falsely suggests that the classifier is performing well [16]. In such cases, the Precision-Recall (PR) curve is a better evaluation metric as calculating both the Precision and Recall does not involve true negatives and the curve is therefore, unaffected by the imbalance in the data [14]. This curve can also provide the viewer with an accurate prediction of future classification performance due to the fact that they evaluate the fraction of true positives among positive predictions [13].

For our classification problem, we have tackled the issue of class imbalance with the use of oversampling. Therefore, for evaluating and comparing the performance of our models, we have considered the evaluation metrics as the AUC of the ROC and P-R curves. Here, we have computed the Average Precision score which corresponds to the AUC of the P-R curve.

	Actual Values	
Predicted Values	**Postive**	**Negative**
Postive	57792	90
Negative	978	1140

Fig. 9. Confusion Matrix - Random Forest

	Actual Values	
Predicted Values	**Postive**	**Negative**
Postive	56281	1601
Negative	561	1557

Fig. 10. Confusion Matrix - LGBM

On comparing the performance of the Light Gradient Boost and Random Forest models for classifying the transactions, it was observed that the precision for Random Forest was better but the recall for LGB model seemed higher indicating greater sensitivity levels for LGB. The f1-score for Random Forest (0.84) was higher than LGB (0.76). The AUC of ROC for both classifiers were almost similar (nearly 0.94). The number of False Positives was considerably low for the Random Forest Model (90) while this number was found to be relatively high for LGB (1601). This shows the effectiveness of our Random Forest Model in mitigating the problem of incorrectly classifying genuine transactions as fraud. Additionally, a higher value for AUC of the P-R curve was observed for Random Forest (0.75) as compared to LG Boost (0.7). We conclude that for the set of parameters chosen after Hyper-parameter Tuning for each classifier using Grid Search Cross Validation, the performance of Random Forest Classifier was marginally better as compared to the LGB model.

Model	Precision	Recall	f1-score	AUC for P-R curve	AUC for ROC curve
Light GB	0.74	0.85	0.79	0.7	0.95
Random Forest	0.96	0.77	0.84	0.75	0.94

Fig. 11. Summarized Evaluation Results

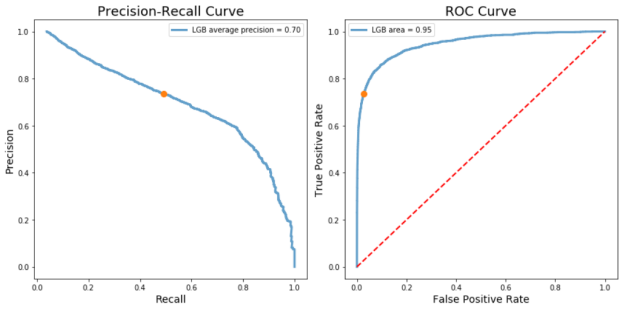

Fig. 12. Classification Report, Precision-Recall and ROC Curves LGBM

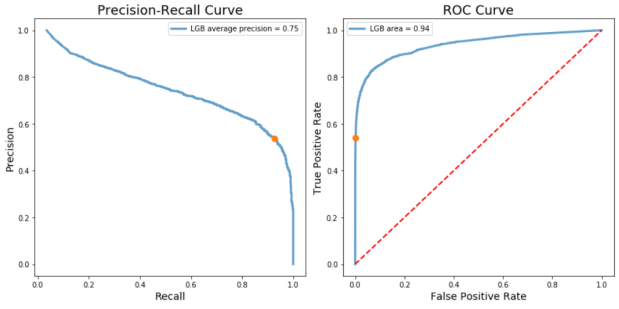

Fig. 13. Classification Report,Precision-Recall and ROC Curves-RF Classifier

V. Conclusion and Future Work

The primary focus of this research is to predict the fraudulent transactions in a given data set. Additionally, work is conducted on comparing the predictive accuracy between LGBM and Random Forest.The work aimed at reducing false positive predictions and false alarm of fraudulent transactions which was achieved by Random Forest Classifier which,with a set of hyper-parameters, performed marginally better than LGBM.

However, the idea of using Deep learning to predict the class with higher accuracy is still a possibility that could be analyzed in the future. Additionally, applying SMOTE to handle class imbalance effectively can also be researched upon.

References

[1] R. J. Bolton and D. J. Hand, "Statistical Fraud Detection: A Review," Statistical Science, vol. 17, no. 3, pp. 235–255, Aug. 2002, doi: 10.1214/ss/1042727940.

[2] C. Phua, V. Lee, K. Smith, and R. Gayler, “A Comprehensive Survey of Data Mining-based Fraud Detection Research,” Sep. 2010, doi: 10.1016/j.chb.2012.01.002.

[3] M. F. O. Moreno, “Comparing the performance of oversampling techniques for imbalanced learning in insurance fraud detection,” Mar. 2018, Accessed: Apr. 12, 2021. [Online]. Available: https://run.unl.pt/handle/10362/33863.

[4] B. Baesens, S. Höppner, I. Ortner, and T. Verdonck, “robROSE: A robust approach for dealing with imbalanced data in fraud detection,” arXiv:2003.11915 [cs, stat], Mar. 2020, Accessed: Apr. 12, 2021. [Online]. Available: http://arxiv.org/abs/2003.11915

[5] S. Taneja, B. Suri and C. Kothari, ”Application of Balancing Techniques with Ensemble Approach for Credit Card Fraud Detection,” 2019 International Conference on Computing, Power and Communication Technologies (GUCON), New Delhi, India, 2019, pp. 753-758.

[6] P. Mrozek, J. Panneerselvam and O. Bagdasar, ”Efficient Resampling for Fraud Detection During Anonymised Credit Card Transactions with Unbalanced Datasets,” 2020 IEEE/ACM 13th International Conference on Utility and Cloud Computing (UCC), Leicester, UK, 2020, pp. 426-433, doi: 10.1109/UCC48980.2020.00067.

[7] H. Shamsudin, U. Yusof, A. Jayalakshmi and M. Akmal Khalid, ”Combining oversampling and undersampling techniques for imbalanced classification: A comparative study using credit card fraudulent transaction dataset”, 2020 IEEE 16th International Conference on Control & Automation (ICCA), 2020. Available: 10.1109/icca51439.2020.9264517 [Accessed 14 April 2021].

[8] D. G e, J. Gu, S. Chang and J. Cai, ”Credit Card Fraud Detection Using Lightgbm Model,” 2020 International Conference on E-Commerce and Internet Technology (ECIT), Zhangjiajie, China, 2020, pp. 232-236, doi: 10.1109/ECIT50008.2020.00060.

[9] Y. Zhang, J. Tong, Z. Wang and F. Gao, ”Customer Transaction Fraud Detection Using Xgboost Model”, 2020 International Conference on Computer Engineering and Application (ICCEA), 2020. Available: 10.1109/iccea50009.2020.00122 [Accessed 14 April 2021].

[10] A. A. Taha and S. J. Malebary, “An Intelligent Approach to Credit Card Fraud Detection Using an Optimized Light Gradient Boosting Machine,” vol. 8, p. 9, 2020.

[11] E.-A. Mînăstireanu and G. Meșniță, “Methods of Handling Unbalanced Datasets in Credit Card Fraud Detection,” BRAIN BROAD Res. Artif. Intell. Neurosci., vol. 11, no. 1, pp. 131–143, Mar. 2020, doi: 10.18662/brain/11.1/19.

[12] K. Cheng, C. Zhang, H. Yu, X. Yang, H. Zou and S. Gao, ”Grouped SMOTE With Noise Filtering Mechanism for Classifying Imbalanced Data,” in IEEE Access, vol. 7, pp. 170668-170681, 2019, doi: 10.1109/ACCESS.2019.2955086.

[13] T. Saito and M. Rehmsmeier, “The Precision-Recall Plot Is More Informative than the ROC Plot When Evaluating Binary Classifiers on Imbalanced Datasets,” PLoS One, vol. 10, no. 3, Mar. 2015, doi: 10.1371/journal.pone.0118432.

[14] J. Davis and M. Goadrich, “The relationship between Precision-Recall and ROC curves,” in Proceedings of the 23rd international conference on Machine learning, New York, NY, USA, Jun. 2006, pp. 233–240, doi: 10.1145/1143844.1143874.

[15] ”IEEE-CIS Fraud Detection — Kaggle”, Kaggle.com, 2021. [Online]. Available: https://www.kaggle.com/c/ieee-fraud-detection/data. [Accessed: 17- Apr- 2021].

[16] J. Brownlee, “How to Use ROC Curves and Precision-Recall Curves for Classification in Python,” Machine Learning Mastery, Aug. 30, 2018. https://machinelearningmastery.com/roc-curves-and-precision-recall-curves-for-classification-in-python/ (accessed Apr. 17, 2021).