Detecting Credit Card Fraud Using Ensemble Learning (Naïve Bayes and Random Forests)

Matthew Fisher
University of Colorado Boulder
mafi6631@colorado.edu

ABSTRACT

Detecting fraudulent credit card transactions is a difficult and important task for banks and credit card companies. They must monitor and track highly skewed data, where the majority of transactions are legitimate and only a small fraction are suspicious or fraudulent. Many researchers have employed various machine learning approaches to achieve a balance between precision and accuracy in detecting such activities. In this work, I present an ensemble learning approach that builds upon the work of other researchers. Classifiers such as Random Forests (RF) and Naïve Bayes (NB) have been utilized for detecting credit card fraud, and few researchers have combined such models and tested their performance. In this project, I worked with a European credit card dataset from the year 2013. An individual RF model was found to do slightly better than the ensemble model (RF + NB) where they had similar performance for recall, specificity, and accuracy but the RF model had slightly better precision (precision: 0.875 vs 0.8537, respectively) and a better Matthews Correlation Coefficient (MCC) (MCC: 0.8069, 0.7970, respectively).

KEYWORDS

Credit card fraud detection, ensemble machine learning, random forest, Naïve Bayes

ACM Reference format:

Matthew Fisher. 2023. Detecting Credit Card Fraud Using Ensemble Learning (Naïve Bayes and Random Forests). Boulder, CO, USA.

1 INTRODUCTION

Credit cards offer a convenient method for conducting transactions with a deferred payment period. The financial company provides a sum of money that can be used, ideally to be paid off in full the following month. In the current digital age, there are various ways an unauthorized individual can obtain your credit card information and make purchases. In 2022, credit card and debit card fraud resulted in losses of ~$34.5 billion globally and is projected to reach almost $50 billion by the year 2030 [8]. Detecting this fraudulent activity quickly and accurately is very important for these companies.

Financial companies have increasingly adopted machine learning techniques for fraud detection due to their speed and efficiency. These algorithms are tasked with distinguishing between a large number of legitimate transactions and a small number of fraudulent ones. Several algorithms, including Naïve Bayes (NB) [3], Random Forest [12], and Neural Networks [11], have been tested and generally perform well in classifying transactions as fraudulent or non-fraudulent. However, few companies have explored ensemble approaches for detecting credit card fraud. In this project, I explored an ensemble model by combining RF and NB via a voting classifier for credit card fraud detection.

1.1 RELATED STUDIES

Public availability of credit card fraud datasets is limited due to the sensitive nature of the information they contain, such as names, addresses, and other private and personal details [10]. This poses a challenge in improving models for detecting credit card fraud. Nevertheless, some researchers have managed to artificially create datasets and test machine learning approaches [1].

Given that most transactions are normal and only a small fraction are fraudulent, researchers have found that either oversampling the minority cases or undersampling the majority cases can enhance the performance of these models. Oversampling increases the number of minority class instances to match that of the majority class, while undersampling does the opposite. Both come with advantages and disadvantages. For instance, oversampling can lead to overfitting, while undersampling reduces the amount of training data. The Synthetic Minority Over-sampling TEchnique (SMOTE) is one method that has been shown to improve classifier performance. Instead of replicating the minority class, SMOTE creates artificial examples of the minority class [5].

Previous studies have explored different types of Random Forests. One group found that a Classification and Regression Trees (CART) Random Forest outperformed a Random-tree-based Random Forest (Accuracy: 96.77% vs 91.96%, respectively) [13]. Other studies have demonstrated the superiority of Naïve Bayes (NB) as a classifier for credit card fraud over k-nearest neighbors and logistic regression (Accuracy: 97.92%, 97.69%, and 54.86%, respectively) [2]. This study aims to investigate the combination of NB and RFs, as most studies tend to compare machine learning algorithms separately rather than in a multi-method manner.

2 PROPOSED METHOD

This work will be divided into four main parts. First, data was collected from Kaggle, which contained European credit card transactions for 2023 [7]. This data consisted of 50% normal and 50% fraud transactions which does not simulate real world data well. Credit card fraud data is supposed to be highly imbalanced, so I found a data set consisting of 2013 European cardholders where frauds were 0.172% of the data [9]. Second, the data will undergo a preprocessing stage, where any duplicates or missing values will be identified and addressed. Third, the NB and RF models will be trained separately (with a data split of training:90, testing:10) and then combined. This process will be conducted and then SMOTE will be applied. Finally, all of the models will be compared using precision, sensitivity, specificity, and accuracy.

The Naïve Bayes technique involves calculating the probability of a particular hypothesis being true given observed data. In this case, we are attempting to classify whether a transaction is normal or fraudulent. This technique employs the naive assumption that all attributes are independent, simplifying the calculations and making this technique efficient for large datasets [6]. Initially, a Gaussian NB was applied assuming it would be appropriate to model the data, but the method is generally used for continuous data and did not perform well. Given the highly imbalanced nature of the data and the binary class, a Complement NB was applied instead. Complement NBs calculate the probability of a given item (in this case, a transaction) belonging to a class.

Complement Naïve Bayes:

$$argmin\, p(y) \ \cdot \ \Pi \frac{1}{p(w|\hat{y})^{f_i}}$$

Random Forests consist of multiple decision trees that use different subsets of the data and identify the most popular class. RFs tend to reduce overfitting compared to decision trees because they involve multiple decision trees, reducing the number of correlated trees. They are also interpretable, allowing us to identify which features in the data are important [4]. The random forest classifier used is the CART classifier, which uses the Gini impurity measure.

Gini impurity:

$$Gini(Node) \ = \ 1 \ - \ \sum_{k=1}^{C} p_k^2$$

The NB technique is efficient and performs well with large datasets, and it tends to be more specific when working with credit card fraud data. On the other hand, RFs are accurate and capable of identifying important features [12]. By combining these two algorithms, the aim is to produce a model that is not only efficient but also more accurate than using either of these techniques individually.

To combine the two models, a voting classifier is used. The voting classifier calculates the argmax of the sum of predicted probabilities for each class (fraudulent or normal). The class with the higher weighted average for a given input is assigned as the output. Finally, to explore better performance of the models, a parameter search was conducted. This is a method to test and find the best multiple parameter combinations.

$$\hat{y} \ = \ argmax_i \sum_{j=1}^{m} w_j p_{ij}$$

2.1 EVALUATION METHOD

To evaluate the data, it will be split into 90% for training data and 10% for testing data. Further testing occurred with splitting ratios of 80:20 and 50:50. Given the sparse nature of fraudulent transactions in the data, it is anticipated that the detection rate for fraudulent transactions will be low. Therefore, it will be crucial to incorporate an oversampling or undersampling method such as SMOTE. Additionally, it is important to try to keep the same ratio when splitting the data into training and testing dataset.

To compare the different methods, a confusion matrix will be used and then the precision, recall, specificity, and accuracy will be calculated. Given the highly imbalanced nature of the data, accuracy tends to be high regardless. If the model were to classify every transaction as normal, it would be correct most of the time. For this project, it is important to include all true positives, true negatives, false positives, and false negatives. To do this, the Matthews Correlation Coefficient (MCC) will be used to compare the models too.

Table 1: Confusion matrix

	Predicted Negative (Normal)	Predicted Positive (Fraud)
Actual Negative (Normal)	True Negative (TN)	False Positive (FP)
Actual Positive (Fraud)	False Negative (FN)	True Positive (TP)

Precision: Rate of correct positive cases.

$$Precision \ = \ \frac{TP}{(TP+FP)}$$

Recall: Rate of fraud cases.

$$Recall \ = \ \frac{TP}{(TP+FN)}$$

Specificity: Rate of non-fraud cases.

$$Specificity \ = \ \frac{TP}{(FN+TN)}$$

Accuracy: Rate of correctly identified cases over all cases.

$$Accuracy \ = \ \frac{(TP+TN)}{(TP+TN+FP+FN)}$$

Matthews Correlation Coefficient: Quality measurement

$$MCC \ = \ \frac{TP\ X\ TN - FP\ X\ FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}}$$

3 EXPERIMENTAL ANALYSIS OF PROPOSED METHOD

The data consists of 492 fraudulent transactions out of 284,807 transactions (about 0.173% frauds) from September 2013 European cardholders (Figure 1). There are 30 features which include V1-V28 (principle components to protect identity), 'Time', 'Amount', and 'Class'.

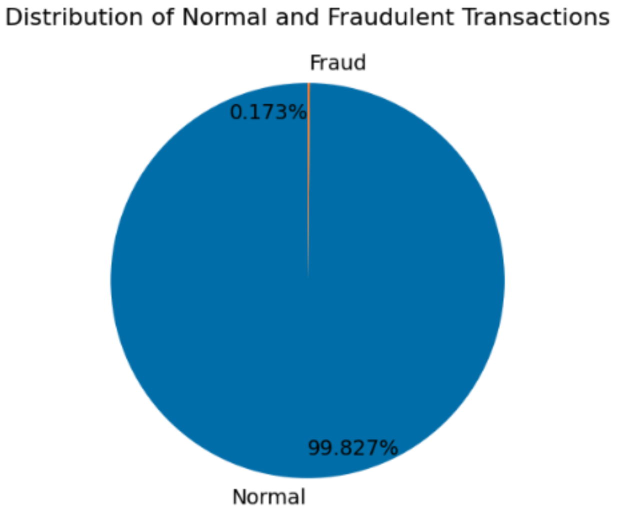

Figure 1: Pie chart of distributions of transactions.

3.1 DATA PREPROCESSING

The data underwent a preprocessing step to prepare it for modeling. This step involved checking for duplicate rows and null values. A total of 1081 duplicate rows were identified and subsequently excluded. No null values were found within the dataset. With these adjustments, the data was ready to be partitioned into training and testing datasets.

3.2 DATA SPLITTING AND OVERSAMPLING

No data warehousing occurred, the preprocessed CVS file was used to split the data into 90% training and 10% testing datasets. An option was used when splitting the datasets to keep the same ratio of fraudulent cases in both sets. Additional split ratios were tested at 80% training and 20% testing, and 50% training and 50% testing datasets.

Once the data was split, the training data could be oversampled. SMOTE was applied to the training data which resampled the dataset and brought up the fraud class to the same level as the normal class.

3.3 DATA MODELING

For this project, the metrics that matter most are recall and FNs. Recall lets one see the ratio of correctly calling the fraud cases and FNs are the cases which the model predicts is normal but it really is a fraud case. A good performing model would be one which has a high recall and low FNs. Other metrics are important too but these would be the ones which we care about the most in a real world setting because these are the cases the model would have missed. FPs are the cases where the model predicts fraud when it was actually a normal case. In a real world scenario, the bank or credit card company could contact the account hold and confirm if they actually made that transaction. Obviously a company would want these cases low too because a bank wouldn't want to continually contact a customer about potential fraud. This could lead to a low retention rate.

Initially, the Gaussian NB model was run, which was suboptimal. A Gaussian NB performs suboptimally in predicting an imbalanced data set, however other NB models can handle them better. The correlation matrix contained 194 FPs, 14 FNs, 33 TPs, and 28,132 TNs (Figure 2). When the Complement NB was run, the correlation matrix contained 6 FPs, 16 FNs, 31 TPs, and 28,320 TNs (Figure 3). This Complement NB model performed much better than the Gaussian NB model since there were a lot less FPs (6 vs 194, respectively).

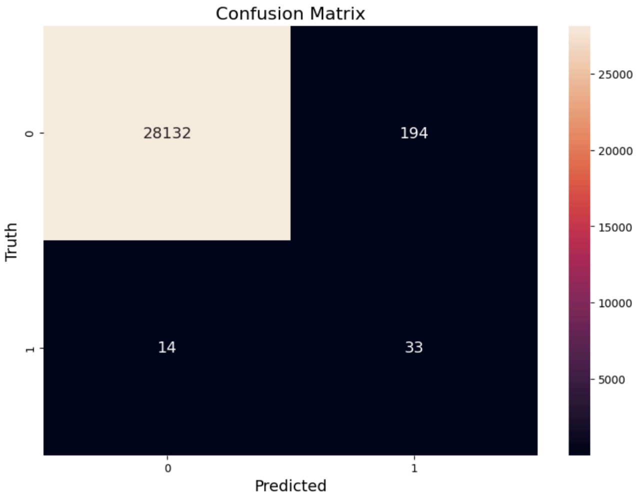

Figure 2: Confusion Matrix for Gaussian NB.

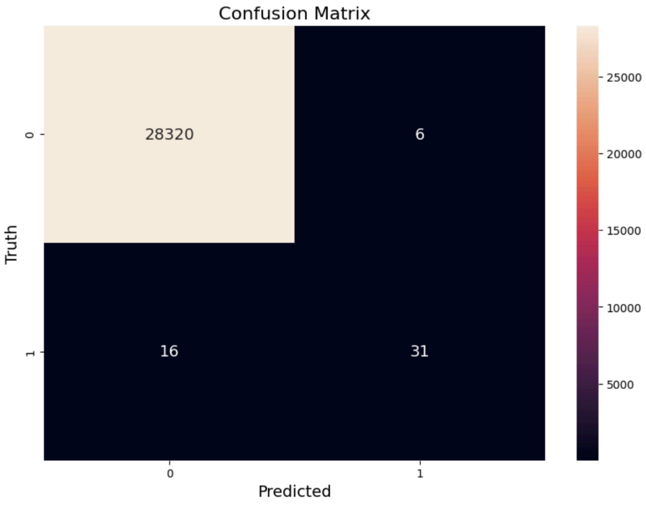

Figure 3: Confusion Matrix for Complement NB.

The RF model performed better than the NB models with a correlation matrix of 5 FPs, 12 FNs, 35 TPs, and 28,321 TNs (Figure 4). Not only did the RF model have lower FNs and FPs but it correctly called a few more normal and fraud transactions. Finally, the combined model (Complement NB and the RF) had similar results to the RF model (Figure 5). It contained the same TPs and FNs as the RF but it had 1 more FP.

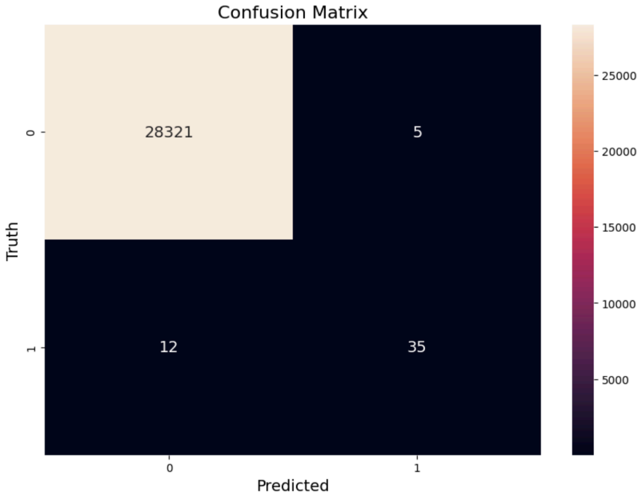

Figure 4: Confusion Matrix for RF.

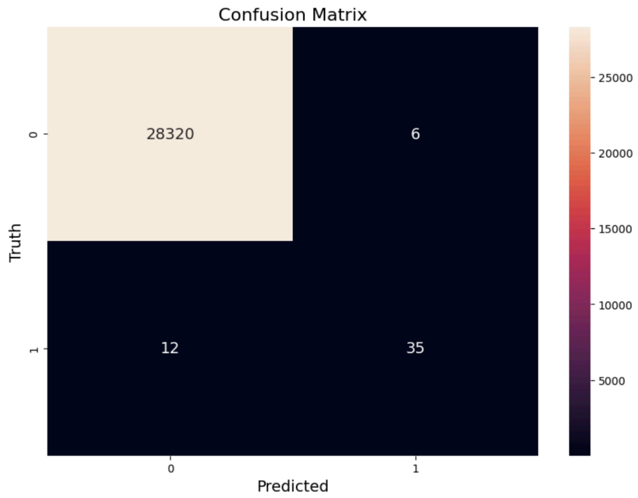

Figure 5: Confusion Matrix for Ensemble model.

Performance of the models can be seen in Table 1 for a 90% train and 10% test dataset. As predicted, all of the models had good accuracy. Once again, the most important metric is the recall. The RF and ensemble model had essentially the same values for recall, sensitivity, and accuracy, but the RF did slightly better for precision and MCC.

Table 1: Performance of models with 90:10 split.

	Precision	Recall	Sensitivity	Accuracy	MCC
Gaussian NB	0.1454	0.7021	0.9932	0.9927	0.3174
Complement NB	0.8378	0.6596	0.9998	0.9992	0.7430
RF	0.875	0.7447	0.9998	0.9994	0.8069
Ensemble	0.8537	0.7447	0.9998	0.9994	0.7970

GridSearchCV was used to test different parameters for the Complement NB, RF, and ensemble models but all of the statistics remained the same. Also, feature selection was tested for each model by looking at and selecting important features within the RF but the statistics did not improve. The statistics were also calculated when splitting the data into different ratios, 80:20 and 50:50. After splitting, SMOTE was applied, the model was run, and the statistics were calculated. The results for the 80:20 split can be seen in Table 2 and the results for the 50:50 split can be seen in Table 3.

Table 2: Performance of models with 80:20 split.

	Precision	Recall	Sensitivity	Accuracy	MCC
Complement NB	0.8293	0.7158	0.9998	0.9993	0.7701
RF	0.8721	0.7895	0.9998	0.9995	0.8288
Ensemble	0.8929	0.7895	0.9998	0.9995	0.8393

Table 3: Performance of models with 50:50 split.

	Precision	Recall	Sensitivity	Accuracy	MCC
Complement NB	0.7830	0.7797	0.9996	0.9993	0.7810
RF	0.8829	0.8305	0.9998	0.9995	0.8560
Ensemble	0.8789	0.8305	0.9998	0.9995	0.8541

When comparing the models for an 80:20 split, the ensemble model had better performance than the complement NB and RF models. The recall, sensitivity, and accuracy between the ensemble and RF models are the same again, but the precision is higher (0.8929 vs 0.8721) and the MCC is higher (0.8393 vs 0.8288). As for the 50:50 split, the RF model outperforms the ensemble model with higher precision (0.8829 vs 0.8789) and higher MCC (0.8560 vs 0.8541).

When comparing the models between different splits, as we move closer to a 50:50 split, recall seems to be increasing. However, this could be due to the fact that there are not a lot of fraud cases and therefore the FNs will remain low but the amount of fraud cases in the test set are increasing. With enough correct calls and few FNs, the recall will increase.

4 DISCUSSION

For this project, I planned on starting the analysis November 11th, 2023 and plan to be finished on November 18th, 2023. From there, I plan to take two to three days to assess the results and finish this write up. This was completed. One potential challenge I saw working on this project is correctly bringing up the minority cases (fraudulent cases) using SMOTE. Another potential challenge for me would be correctly combining the two algorithms (RF and NB). I worked through those challenges and even decided to take a few days to try and add parameter selection, feature selection, and testing different data sets. This was completed and added to the report.

This project consisted of using a multi-method model (combination of RF and NB) to detect credit card fraud and see if it would perform better than the two models individually. In one instance (80:20 split), the ensemble model did outperform the individual models but in all other cases the RF model performed better. There was a focus on recall and FNs within the project.

Recall is the ratio of correctly calling fraud cases, and FNs occur when a fraudulent transaction is incorrectly predicted as normal. Ideally, recall would be very high, close to 1, and FNs would be low, close to 0. As for FPs, where the model predicts a transaction as fraudulent when it is actually normal, they are not a significant concern. In a real life scenario, the bank can contact the account holder (E.g., via text or email) and confirm if they made that transaction or not.

The recall, sensitivity, and accuracy between the RF model and the ensemble model was the same for a given data split (90:10, 80:20, 50:50). The precision and MCC for the RF model is slightly better for data splits 90:10 and 50:50 but higher for the ensemble model for the 80:20 split. The better ensemble model could be due to how the data was split. The recall seemed to increase as the splits got closer to 50:50 but this could be due to more fraud cases being in the test datasets.

Currently, the RF model is slightly better for this dataset in most cases. Future work should include feature selection, parameter optimization, and different ensemble models on a larger dataset. Though parameter optimization did not improve any metrics in this project, it could be important in other projects. Feature selection has been shown to improve performance of machine learning models in imbalanced datasets and as datasets grow, it could be important to speed up the models. Finally, larger datasets could improve the models by having more data to train the models on.

5 CONCLUSION

Credit card fraud detection is important in the digital world of today. With increasing rates of fraud, it is crucial to continually develop and improve credit card fraud detection methods. In this project, the goal was to combine the advantages of Naïve Bayes and Random Forests to create an efficient and accurate fraud detection method.

The RF and ensemble method (RF + NB) perform very similarly, with the RF model having slightly better precision and MCC in most cases, except for data split 80:20 where the ensemble model outperformed the RF model. The rest of the statistics are essentially the same when comparing the models with data splits of 90:10, 80:20, and 50:50. Currently, a standard RF model using scikit-learn performs the best out of the models tested in this project. It will be important to test other combinations of models on larger datasets for credit card fraud detection.

REFERENCES

[1] E. Aleskerov, B. Freisleben, and B. Rao. 1997. CARDWATCH: a neural network based database mining system for credit card fraud detection. In *Proceedings of the IEEE/IAFE 1997 Computational Intelligence for Financial Engineering (CIFEr)*, IEEE, 220–226. DOI:https://doi.org/10.1109/CIFER.1997.618940

[2] John O. Awoyemi, Adebayo O. Adetunmbi, and Samuel A. Oluwadare. 2017. Credit card fraud detection using machine learning techniques: A comparative analysis. In *2017 International Conference on Computing Networking and Informatics (ICCNI)*, IEEE, 1–9. DOI:https://doi.org/10.1109/ICCNI.2017.8123782

[3] Alejandro Correa Bahnsen, Aleksandar Stojanovic, Djamila Aouada, and Björn Ottersten. 2014. Improving Credit Card Fraud Detection with Calibrated Probabilities. In *Proceedings of the 2014 SIAM International Conference on Data Mining (SDM)*. Society for Industrial and Applied Mathematics, 677–685. DOI:https://doi.org/10.1137/1.9781611973440.78

[4] Leo Breiman. 2001. Random Forests. *Mach. Learn.* 45, 1 (October 2001), 5–32. DOI:https://doi.org/10.1023/A:1010933404324

[5] Nitech V. Chawla, Kevin W. Bowyer, Lawrence O. Hall, and W. Philip Kegelmeyer. 2011. SMOTE: Synthetic Minority Over-sampling Technique. *arXiv [cs.AI]*. Retrieved from http://arxiv.org/abs/1106.1813

[6] Ivo D. Dinov. 2018. Probabilistic Learning: Classification Using Naive Bayes. In *Data Science and Predictive Analytics: Biomedical and Health Applications using R*, Ivo D. Dinov (ed.). Springer International Publishing, Cham, 289–305. DOI:https://doi.org/10.1007/978-3-319-72347-1_8

[7] Nidula Elgiriyewithana. 2023. Credit Card Fraud Detection dataset 2023. Retrieved November 10, 2023 from https://www.kaggle.com/datasets/nelgiriyewithana/credit-card-fraud-detection-dataset-2023/data

[8] John S. Kiernan. 2023. Credit card fraud statistics. *WalletHub*. Retrieved November 9, 2023 from https://wallethub.com/edu/cc/credit-card-fraud-statistics/25725

[9] MACHINE LEARNING GROUP-ULB. 2018. Credit Card Fraud Detection. Retrieved November 15, 2023 from https://www.kaggle.com/datasets/mlg-ulb/creditcardfraud

[10] Sam Maes, Karl Tuyls, Bram Vanschoenwinkel, Bernard Manderick. Credit card fraud detection using Bayesian and neural networks. Retrieved November 9, 2023 from https://www.researchgate.net/profile/Karl-Tuyls/publication/254198382_Machine_Learning_Techniques_for_Fraud_Detection/links/555f695508ae6f4dcc926e88/Machine-Learning-Techniques-for-Fraud-Detection.pdf

[11] E. W. T. Ngai, Yong Hu, Y. H. Wong, Yijun Chen, and Xin Sun. 2011. The application of data mining techniques in financial fraud detection: A classification framework and an academic review of literature. *Decis. Support Syst.* 50, 3 (February 2011), 559–569. DOI:https://doi.org/10.1016/j.dss.2010.08.006

[12] Yakub K. Saheed, Moshood A. Hambali, Micheal O. Arowolo, and Yinusa A. Olasupo. 2020. Application of GA Feature Selection on Naive Bayes, Random Forest and SVM for Credit Card Fraud Detection. In *2020 International Conference on Decision Aid Sciences and Application (DASA)*, IEEE, 1091–1097. DOI:https://doi.org/10.1109/DASA51403.2020.9317228

[13] Shiyang Xuan, Guanjun Liu, Zhenchuan Li, Lutao Zheng, Shuo Wang, and Changjun Jiang. 2018. Random forest for credit card fraud detection. In *2018 IEEE 15th International Conference on Networking,*

Sensing and Control (ICNSC), IEEE, 1–6. DOI:https://doi.org/10.1109/ICNSC.2018.8361343