Credit card Fraud detection

Ramaswamy Iyappan, George Mason University
Abhijeet Amitava Banerjee, George Mason University
Joel Sadanand Samson, George Mason University

1. Abstract

With the current trend of digital banking, which has simplified bank transactions, there has also been a considerable increase in the number of fraudulent transactions. To prevent customers from being charged for products they did not purchase, it is critical to find a way to identify fraudulent transactions. Also, legitimate transactions outnumber fraudulent ones by a large margin, making it incredibly challenging to distinguish between the two. To identify these scams as they take place and alert the relevant parties, this issue must be dealt with on a high-priority basis.

This research suggests utilizing machine learning techniques to automatically identify such fraudulent transactions and report them as they happen. We primarily concentrate on resolving the class imbalance issue (fraud and non-fraud) by comparing results from various over sampling and under sampling strategies, which enables machine learning models to learn from a more balanced distribution and generate precise predictions. We then compare how classifiers from ensemble learning and artificial neural networks perform on unseen test instances after being trained on the balanced training dataset.

2. Introduction

Credit card fraud is the act of using a credit card that has been suspended, cancelled, reported lost, or stolen with the aim of defrauding someone of something of value. Credit card fraud also includes using a credit card number without having the physical card. Another, more dangerous type of credit card fraud combines identity theft with credit card fraud by stealing someone's identity in order to obtain a credit card. The whole consumer credit industry is affected by the issue of credit card fraud. One of the fraud categories with the quickest growth rates is also one of the hardest to stop. During 2020, credit card and debit card gross fraud losses accounted for roughly 6.81 cents per $100 in total volume, up from 6.78 cents per $100 in 2019. In 2020, the United States accounted for 35.83 percent of global payment card fraud losses while generating only 22.40% of total volume. Over the next 10 years, card industry losses to fraud will collectively amount to $408.50 billion.

The detection and resolution of credit card fraud concerns are slowed significantly by using conventional procedures. Machine learning in banking has the potential to help all types of financial institutions find answers more quickly and accurately. AI-based fraud detection systems continuously and thoroughly scan consumer credit card statements. By identifying purchasing trends that are invisible to the human eye, AI's sophisticated pattern and anomaly identification comes into play. The technology raises a red flag, for instance, if a user doesn't often purchase online but starts doing so more frequently, according to their card statement. Similar to this, a red flag is raised if a user suddenly starts making purchases from locations that are far from where they are staying. Additionally, machine learning-based banking applications can distinguish between a single purchase and many purchases with the same pattern. This way, false positives in the detection of credit card fraud are prevented by using conventional procedures. Credit card companies or banks can send out their investigation team to investigate the situation after the

system raises red flags after scanning real-time purchases in order to address it before any serious card activity occurs.

The main focus of the project is to improve on the previous fraud detection models comparing various sampling techniques, classifiers, and performance metrics. We used a dataset generated by the transactions made by European card holders in September 2013. All the transactions were made in two days, so the dataset is highly skewed. To solve this issue, we used various sampling methods on the dataset since machine learning models would learn better from a balanced dataset. Here we also see why accuracy is not a good metric for our problem and explore other metrics that are widely used for dealing with unbalanced data such as precision, recall & AUC-PR.

3. Methodology

3.1 Dataset

The dataset used for this research was obtained from Kaggle and comprises of 284,807 credit card transactions made by European cardholders over the course of 2 days in September 2013. It is extremely unbalanced because there are just 492 fraud incidents (positive class), which represent only 0.172% of all transactions. Due to security and privacy concerns, all original features and context were concealed. All features are numerical input variables where V1, V2, …V28 were produced by applying PCA transformations to the original features. 'Amount' represents the transaction amount. "Time" was removed because it had no purpose in this study. The target variable 'Class' has the value 1 for fraudulent transactions and 0 for legal ones. Since the "Amount" feature has a varied range, it was scaled using StandardScalar(). Using train_test_split() and a test size of 20%, the dataset was divided into training and testing sets.

3.2 Dealing with class imbalance using Sampling techniques

As we know, the dataset contains just 492 fraudulent cases out of 284,807 transactions, implying that the positive class (fraud cases) accounts for only roughly 0.172% of the total dataset. Therefore, even if a classifier predicts every example as fraudulent (majority class), it will still have a misleadingly high accuracy of 99.82%. This is because the classifier only predicted the majority class, which makes up roughly 99% of the dataset, and completely ignored the minority class (fraud), which is the primary goal. Therefore, machine learning models trained on unbalanced class distributions may not yield meaningful findings, which is why we need to use sampling strategies to change the dataset into one that is more evenly distributed.

By integrating some of the available sampling techniques, we can under sample the majority class (non-fraud) to reduce it to a distribution that is similar to the minority class, oversample the minority class (fraud) to become a large amount of the dataset almost like the majority class, or even combine both at once. In this project, we tried exploring a few that includes:

Figure 1: Sampling Techniques

Oversampling Minority class (Fraud)	Undersampling Majority class (non-fraud)	Combination of Oversampling and Undersampling	
RandomOverSampler (ROS)	RandomUnderSampler (RUS)	ROS+RUS	ROS+NearMiss
Synthetic Minority Oversampling Technique (SMOTE)	NearMiss	SMOTE+RUS	SMOTE+NearMiss

Note: The sampling techniques were tried only on the training data, because we need similar class distributions to train the classifiers. Resampling on the test set modifies the information from actual transactions. Therefore, it is important to avoid making any changes to the test set, which simulates a real-world scenario, when evaluating the effectiveness of trained models.

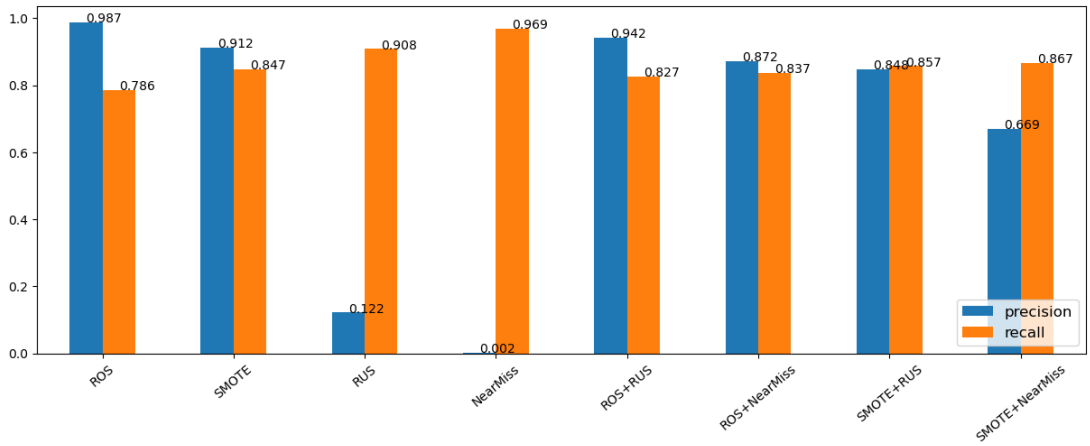

Figure 2: Bar-plot of Precision-Recall by different Sampling techniques

The bar graph above displays the precision and recall values for several sampling techniques trained on a Random Forest Classifier. It is evident from the graphs that (1) employing SMOTE to oversample only the minority class and (2) combining ROS+RUS produce better outcomes (both precision & recall). So, both strategies were chosen to train and compare in order to identify a better model.

3.3 Proposed Models

For this research, we carried out experiments using ensemble learning models like bagging, boosting, voting, and artificial neural networks along with oversampling (SMOTE) and combination of oversampling and undersampling (ROS+RUS) to learn the dataset and identify the best learning model among them. More specifically, we used:

(1) Random Forest Classifier with 100 estimators (Decision trees) as an instance of Bagging that uses bootstrap samples of the training dataset.
(2) XG Boost Classifier with a maximum depth of 3 for each tree as an instance of Boosting.
(3) Combination of base models such as Logistic Regression (with L2 penalty), Decision Tree Classifier, Gaussian Naïve Bayes, and K Nearest Neighbors with K=5 neighbors as an instance of Voting Classifier.
(4) Multi-Layer Perceptron (MLP) as an instance of ANN that consists of two hidden layers of size (100, 50), which uses Cross Entropy Loss and ReLu activation function.

4. Technical Results

As previously stated, using accuracy as a measure of a model's performance is useless because it is misleading in the context of the question we are trying to answer. Therefore, in order to evaluate models with unbalanced data, we used different performance metrics like Precision, Recall, and Area under the Precision-Recall Curve (AUC-PR).

4.1 Precision

$$Precision = \frac{True\ Positives}{True\ Positives + False\ Positives}$$

Precision measures the proportion of positive predictions that actually belong to the positive class. Here, precision defines the fraction of fraudulent predictions by a model that were observed as fraud.

Figure 3: Precision values by different models

MODEL	**SMOTE**	**ROS+RUS**
Random Forest Classifier	0.912	0.942
XGB Classifier	0.307	0.741
Voting Classifier	0.699	0.766
MLP	0.695	0.732

Using both sampling techniques, it is evident that Random Forest performs substantially better than other models, who barely perform above average. It appears that Random Forest doesn't really rely on any sampling methods. However, shows a higher precision value when using a combination of Oversampling and Undersampling (ROS+RUS: 0.942).

4.2 Recall

$$Recall = \frac{True\ Positives}{True\ Positives + False\ Negatives}$$

Recall measures the proportion of positive observations that were predicted correctly. Here, recall defines the fraction of fraud transactions that were correctly identified as fraud by a model. This is our main objective since it tells the amount of fraud transactions correctly identified by a model and must be optimized to find a relatively better performing classifier.

Figure 4: Recall values by different models

MODEL	**SMOTE**	**ROS+RUS**
Random Forest Classifier	0.847	0.826
XGB Classifier	0.888	0.878
Voting Classifier	0.878	0.867
MLP	0.816	0.837

The table shows that regardless of the sampling technique, recall values for all the models are essentially the same. In comparison to all, XG Boost classifier has the highest recall value (0.888).

4.3 Area under the Precision-Recall curve

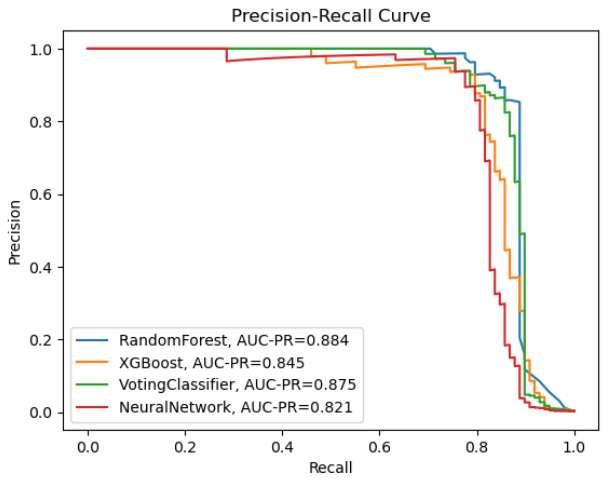

Figure 5: Precision-Recall curve using SMOTE

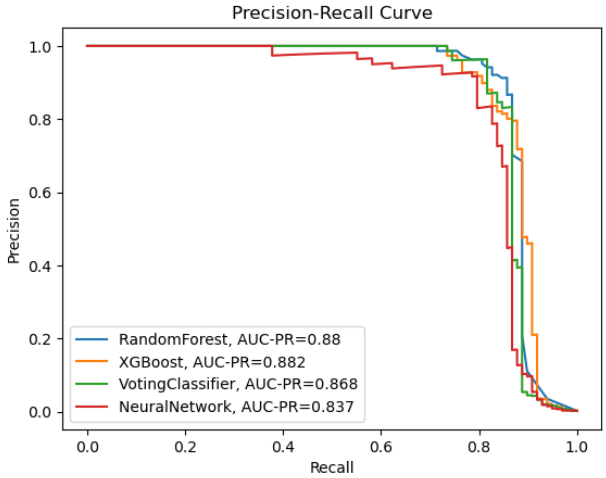

Figure 6: Precision-Recall curve using ROS+RUS

When interpreting a classifier's output for binary classification, the precision-recall curve is particularly helpful. The precision-recall plot and area under the curve (AUC) score are used to summarize a model's performance. The better the model, the closer the curve hugs the top right corner of the graph.

Here, scores of all classifiers on the two plots are remarkably similar. However, the Random Forest Classifier has a higher AUC-PR (0.884 & 0.88) and excellent recall vs precision plot (blue curve).

4.4 Towards a better model

The aforementioned table and graphs demonstrate that the AUC-PR and Recall scores of each model only vary by a small amount. However, when examining the precision table, classifiers MLP, XGB, and Voting show almost average precision with SMOTE, implying that they are merely guessing and frequently accuse customers who make legitimate transactions as fraud. This is a worst-case scenario in which these no-skill models fail to identify fraudulent transactions (low recall) and instead accuse legitimate transactions of being fraudulent (low precision). Therefore, we must try to obtain a model that accurately predicts most fraud cases while also producing very few false positive predictions that do not heavily implicate customers (better recall).

According to the above experiments and technical results, the Random Forest Classifier achieves significantly higher precision, recall, and AUC-PR scores when used with both sampling techniques. However, there is a trade-off between precision and recall, with better recall when used with SMOTE and better precision when used with ROS+RUS. But since using SMOTE involves duplicating existing examples, it is likely to overfit and require more learning time because it adds more training examples. As a result, a highly effective method for detecting fraudulent transactions without accusing the customers is to combine the RandomOverSampler and RandomUnderSampler on the training data and learn it using a Random Forest Classifier.

5. Conclusion

Through this study, we were able to explore different Sampling methods for handling unbalanced dataset, use Ensemble modeling for reducing generalization error of classification models, and understand how evaluation metrics like Precision, Recall, AUC of recall vs precision, tell us how a model performs on an unbalanced distribution.

There are some situations in the actual world that cannot be avoided for a variety of reasons, much like class inequality. There are still additional machine learning techniques that may be used to provide predictions that are more accurate and realistic, such as cost-sensitive learning, various learning algorithms, anomaly detection, and the use of measures like false positives, true positives, F1-score, etc. Imposing an effective approach for significantly identifying fraudulent transactions without blaming customers is critical to resolving this slowly prevalent global challenge.

6. References

(1) Emmanuel Ileberi, Yanxia Sun, Zenghui Wang. A machine learning based credit card fraud detection using the GA algorithm for feature selection. In: Journal of Big Data 9: 2022, Article: 24.

(2) Varmedja D, Karanovic M, Sladojevic S, Arsenovic M, Anderla A. Credit card fraud detection-machine learning methods. In: 18th international symposium INFOTEH-JAHORINA (INFOTEH); 2019. p. 1-5.

(3) Campus K. Credit card fraud detection using machine learning models and collating machine learning models. Int J Pure Appl Math. 2018;118(20):825–38.
(4) https://wallethub.com/edu/cc/credit-debit-card-fraud-statistics/25725
(5) https://journalofbigdata.springeropen.com/articles/10.1186/s40537-022-00573-8
(6) https://machinelearningmastery.com/roc-curves-and-precision-recall-curves-for-imbalanced-classification/
(7) https://www.kaggle.com/datasets/mlg-ulb/creditcardfraud
(8) https://medium.com/@douglaspsteen/precision-recall-curves-d32e5b290248
(9) https://towardsdatascience.com/ensemble-methods-bagging-boosting-and-stacking-c9214a10a205
(10) https://machinelearningmastery.com/combine-oversampling-and-undersampling-for-imbalanced-classification/
(11) https://www.kaggle.com/code/rafjaa/resampling-strategies-for-imbalanced-datasets/notebook
(12) https://machinelearningmastery.com/tour-of-evaluation-metrics-for-imbalanced-classification/
(13) https://velog.io/@jiselectric/Ensemble-Learning-Voting-and-Bagging-at6219ae
(14) https://machinelearningmastery.com/xgboost-for-imbalanced-classification/
(15) https://towardsdatascience.com/ensemble-learning-bagging-boosting-3098079e5422