**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. In order to prevent customers from being charged for products they did not purchase, it is critical that credit card issuers can identify fraudulent transactions. Also, legitimate transactions outnumber fraudulent ones by a large margin, making it incredibly challenging to distinguish between the two. In order to identify these scams as they take place and alert the relevant parties, it is extremely important that this issue 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 & non-Fraud) by comparing results from various Over sampling & 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.

1. **Introduction**

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₵ per $100 in total volume, up from 6.78₵ per $100 in 2019. In 2020, US accounted for 35.83% of the worldwide payment card fraud losses but generated only 22.40% of total volume. Over the next 10 years, card industry losses to fraud will collectively amount to $408.50 billion [1].

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. In these finance-based enterprises, the development of digitization in banking has brought up a number of cybersecurity-related challenges. 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. False positives in the detection of credit card fraud are prevented in this way. Credit card companies or banks can send out their investigation team to look into 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 the improve on the previous fraud detection models by using and comparing various sampling techniques, classifiers and performance metrics. We used a dataset generated by the transaction made by European card holders in September 2013, all of the transactions were made in two days, so the dataset is highly skewed and to solve this issue we have used Cross-validation technique on training data and sampled the data, because our model would learn better from a balanced dataset. Here we also see why accuracy is not a good metric for our problem.

1. **Methodology**
   1. **Dataset**

The dataset used in this research is taken from Kaggle which consists of 284,807 credit card transactions made by European card holders in 2 days of September 2013. There are 492 fraud cases (positive class) which accounts only for about 0.172% of all transactions making it highly unbalanced. All original features and background information were hidden due to security and privacy reasons. All features are numerical input variables where V1, V2, …V28 except ‘Time’ and ‘Amount’ were obtained by applying PCA dimensionality reduction to the original features. ‘Amount’ represents the transaction amount and ‘Time’ represents the seconds between each transaction. The target variable is ‘Class’ which consists of 1 in case of fraud and 0 for legitimate transactions.

‘Time’ feature was dropped since it was irrelevant for the purpose of this research. The ‘Amount’ feature was scaled using StandardScalar () because of having a different range. There was no case of a NULL or a NaN value making it easier for the learning models. The dataset was split into training and testing sets using train\_test\_split () with a test size of 20%.

* 1. **Dealing with class imbalance using Sampling techniques**

As we know, the dataset contains only 492 fraudulent cases out of 284,807 transactions, i.e., the positive class (fraud cases) account only for about 0.172% of the whole dataset. So, even if a classifier predicts everything as not fraud (majority class), it will show a very high accuracy of 99.82% which is misleading. This is because the classifier just predicts the majority class which is almost 99% of the dataset and just totally avoided the minority class (fraud) which is the main objective. If we split and check the class-wise correct predictions, then non fraud cases will have 100% accuracy and fraud cases will have 0% accuracy. Hence machine learning models might not produce significant results when trained on imbalanced class distributions, which is why we need to apply sampling techniques to transform it to a more balanced dataset.

By using some available sampling techniques, we can oversample the minority class (fraud) to become a significant proportion of the dataset almost like the majority class, or undersample the majority class (non-fraud) to bring it down to a similar distribution as the minority class, or even combine both simultaneously. 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. Performing resampling on test set, changes the data from real-world transactions. Hence, the test set (which represents an instance of reality) should be kept unseen and used to measure performance of trained models.

Chart, bar chart

Description automatically generated

The above bar plot shows the mean precision and recall values by different sampling strategies trained on a Random Forest Classifier. We can clearly see that plots from (1) Oversampling only the minority class using SMOTE and (2) combination of ROS+RUS, show better results (both precision & recall). So, both these techniques were selected to train and compare, for finding a better model.

* 1. **Proposed Models**

For this research, we ran experiments by applying both Oversampling (SMOTE) and combination of Oversampling and Undersampling (ROS+RUS) with Ensemble learning models such as Bagging, Boosting, Voting, and Artificial Neural Networks to learn the dataset and find a better learning model amongst them. More specifically, we used:

1. Random Forest Classifier as an instance of Bagging.
2. XG Boost Classifier as an instance of Boosting.
3. Combination of base models such as Logistic Regression, Decision Tree Classifier, Gaussian Naïve Bayes, and K Nearest Neighbors as an instance of Voting Classifier.
4. Multi-Layer Perceptron (MLP) as an instance of ANN.
5. **Technical Results**

As mentioned before, using accuracy won’t help us in evaluating a model’s performance as it is misleading for our subject of interest. So, we used other performance metrics such as Precision, Recall and Area under the Precision-Recall curve (AUC-PR) which are effectively used to evaluate models with imbalanced data.

* 1. **Precision**

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.

|  |  |  |
| --- | --- | --- |
| 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 |

We can clearly see that Random Forest perform relatively better while other models perform hardly just above average, using both the sampling techniques. Random Forest doesn’t seem to really depend on any sampling technique. However, shows better precision value while using a combination of Oversampling and Undersampling (ROS+RUS).

* 1. **Recall**

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 identified by the models and must be optimized to find a relatively better performing classifier.

|  |  |  |
| --- | --- | --- |
| 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 |

From the table, recall values for all the models are almost similar irrespective of the sampling technique. The XG Boost classifier shows the highest recall value amongst all the models.

* 1. **Area under the Precision-Recall curve**

Chart

Description automatically generatedDiagram

Description automatically generated

SMOTE ROS+RUS

1. **Conclusion**

**6. References**

1. https://wallethub.com/edu/cc/credit-debit-card-fraud-statistics/25725
2. https://journalofbigdata.springeropen.com/articles/10.1186/s40537-022-00573-8
3. https://machinelearningmastery.com/roc-curves-and-precision-recall-curves-for-imbalanced-classification/
4. https://www.kaggle.com/datasets/mlg-ulb/creditcardfraud
5. https://medium.com/@douglaspsteen/precision-recall-curves-d32e5b290248