Finding Fraud Faster

Macon Barker – Section B – BAN 6053 – 2/23/2024

# Executive Summary

Our analytics team was hired by Deacon Bank to develop a model that can help predict and identify transactional fraud. Given a data set with 26 variables, we built 3 different types of classification models using Logistic Regression, Random Forest, and Gradient Boosting. Our best model was a **Gradient Boosting** model which had an accuracy of 97.36%. The features that had the largest impact were the adjusted transaction amount, transaction type, and account age in days.

Below we will walk through our initial analysis, methodology and model selection process, and our recommendations moving forward.

## Analysis

### Exploratory Analysis

**Distribution of Target Variable**

Our data set contains 94.57% legit transactions and 5.43% fraud transactions.

A screenshot of a computer

Description automatically generated

**Missing Values**

Out of the 26 columns, 21 contained at least 1 missing value. However, the amount of missing values was still pretty insignificant. During our exploratory analysis and for model training we will fill in the missing values one of two ways: **numeric columns** will be filled with the **mean** and **categorical columns** will be filled with the **mode**.

**Exploratory Data Analysis**

For an exploratory data analysis, we looked at the distribution of fraud vs. legit transactions for a number of the variables. The 3 variables that stuck out the most were the **Transaction Environment**, **Account Age in Days**, and **Adjusted Transaction Amount**.

Below, we can see that there are quite a few transaction environments that contain a high percentage of fraudulent transactions. Transactions made in environments A, P, and Y were **all** fraudulent ones. This insight is valuable because if we see a transaction occur in one of these environments, there is a very high chance that it is a fraudulent one.

A graph of a graph of numbers and letters

Description automatically generated with medium confidence

The distribution of account age in days shows a slightly higher average for fraudulent transactions. This is interesting because it might be thought to be the opposite.

A graph of a distribution of account age days with Ryugyong Hotel in the background

Description automatically generated

Comparing the transaction amount with the adjusted transaction amount provides some valuable information for tracking fraud. We can see that for fraudulent transactions, the initial amount is slightly higher on average than legitimate transactions while the adjusted amount is typically lower on average.

A diagram of a distribution of a number of transactions with Ryugyong Hotel in the background

Description automatically generated with medium confidenceA graph of a distribution of a product

Description automatically generated with medium confidence

**The Firm believes that email domain and billing postal code are important predictors, your write-up should discuss why or why not.**

To test the importance of email domain and billing postal code, we took the top 10 from each column. There are thousands of different domains and postal codes, but we think that analyzing the 10 most common will give us insight as to how important the variable is as a whole. We can see that none of the top postal codes or email domains have a strong influence on the legitimacy of the transaction. Therefore, we conclude that email domain and billing postal code are not important predictors for fraud.

A graph of a bar chart

Description automatically generated with medium confidenceA graph of a bar chart

Description automatically generated with medium confidence

A Standard Scaler will be used for numeric variables and a One Hot Encoder for categorical.

Missing values for numeric variables are imputed with the mean and categorical variables will use the most frequent (mode).

A screen shot of a computer program

Description automatically generated

## METHODOLOGY

### Model Development

To help classify our transactions as legit or fraudulent, we built 3 different types of models: Logistic Regression, Random Forest, and Gradient Boosting. Before training our models, we decided to resample our data with a technique called SMOTE. Many classification methods are prone to performing well on the training data set, but not working as well on new data sets. Because the distribution of our target variable was so skewed one way (95% of transactions were legit), we decided to resample the data to where 67% were legit and 33% were fraud. We believe that this will produce models that will better perform on new data sets.

**Parameter Tuning**

Tuning involves adjusting the parameters used to build the model to find the best-performing model. We used a method called GridSearch, which tests various combinations of parameters and gives us the best results to use.

**Feature Selection**

After some exploratory analysis, we decided to proceed with the following 8 variables for our model-building process. These were chosen because they seem to have an effect on the distribution of legit versus fraud.

Numeric Columns:

* account\_age\_days
* transaction\_amt
* transaction\_adj\_amt
* historic\_velocity
* initial\_amount

Categorical Columns:

* transaction\_initiate
* transaction\_type
* transaction\_env

**Model Evaluation**

**Performance Metrics**

To select our best-performing model, we looked at 5 different summary statistics: Accuracy, AUC, Precision, Recall, and F1 Score. Below is a brief explanation of each metric as it pertains to our use case.

* Accuracy: the percentage of predictions made correctly; for both fraud and legit transactions
* ROC AUC: ‘Area Under the Curve’ with 1 being perfect. X-axis is the false positive rate (actual legit transactions classified as fraud) and Y-axis is the true positive rate (actual fraud transactions classified as fraud). Indicates how well the model is able to distinguish between fraud and legit.
* Precision: out of the transactions predicted as fraud, the percentage that were actually fraudulent
* Recall: (True Positive Rate) out of all the actual fraudulent transactions, the percentage that were predicted correctly as fraudulent by the model
* F1 Score: combines precision and recall as a sort of average; typically when recall increases, precision will decrease; gives a single value that describes the performance of the model identifying fraudulent transactions correctly

**Model Comparison**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Model | Accuracy | AUC | Precision | Recall | F1 Score |
| Logistic | 0.9391 | 0.9336 | 0.4684 | 0.7990 | 0.5906 |
| Random Forest | 0.9709 | 0.9343 | 0.7581 | 0.6893 | 0.7221 |
| XGBoost | 0.9736 | 0.9396 | 0.8150 | 0.6714 | 0.7362 |

**Best Model**: XGBoost

We selected the Gradient Boosting model to move forward with as our best-performing model. It had the highest results in four out of the five metrics. The Logistic Regression model had a very high recall, however, it did not perform as well in some of the other metrics. The Random Forest model was pretty comparable to the Gradient Boosting in each of the five statistics, which makes sense as both use decision trees in the fitting process. We are more confident in the XGBoost model because Gradient Boosting applies an error correction where each subsequent tree tries to learn from the misclassifications made by the previous one. This process makes for a stronger predictive model.

**Feature Importance Analysis**:

Below are the top ten features for each model. The higher the score, the larger the effect that variable has on predicting the target variable, in our case fraudulent transactions. We can see that ‘transaction\_adj\_amt’ and ‘account\_age\_days’ were in the top five for both the Random Forest and Gradient Boosting models.

**Logistic Regression**

A graph of a bar graph

Description automatically generated

**Random Forest**

A graph with blue squares

Description automatically generated

**Gradient Boosting**

A graph with blue and white bars

Description automatically generated

**Model FPR/TPR/Threshold Table**

|  |  |  |
| --- | --- | --- |
| Target False Positive Rate | True Positive Rate (TPR) | Prob Threshold |
| 1% | 68.45% | 0.4712 |
| 2% | 73.98% | 0.2935 |
| 3% | 77.72% | 0.1932 |
| 4% | 80.34% | 0.1380 |
| 5% | 82.18% | 0.1059 |
| 6% | 83.54% | 0.0843 |

**ROC Charts for each model on Test Set.**

A graph of a curve

Description automatically generated

**Random Forest vs. GBM/XGBoost**:

Random Forest and Gradient Boosting both build multiple decision trees to help classify data. The difference comes with how the trees are built. With Random Forest, the trees are built separately and then the prediction is made by the majority class of all individual trees. Gradient boosting builds its decision trees sequentially, where each subsequent tree tries to correct the misclassifications of the previous.

## Recommendations

### Understanding False Positive Rate

A **5% False Positive Rate** means that out of all the transactions that are actually legit, 5% of them are incorrectly classified as fraudulent. In other words, of all the transactions predicted fraudulent, 95% were actually fraudulent. This means that the model would have a precision of 95% because precision is calculated as 1 – FPR. When precision is this high, the recall (otherwise known as True Positive Rate) tends to suffer. In a business case where the objective is to catch fraudulent transactions, recall is the more important statistic to focus on because recall tells us of all the transactions that were fraudulent, what percentage did we catch or correctly identify.

The lower the False Positive Rate, the less likely it is to incorrectly flag a legitimate transaction as fraudulent, therefore improving the customer experience. No one wants the misfortune of being assumed fraudulent when he/she has done nothing wrong. However, as mentioned above, this comes with a trade-off that some fraudulent transactions may slip through the identification process and lose the company money. The business should decide the balance they would like to have between detecting as much fraud as possible without upsetting customers by misclassifying their transactions as fraudulent.

**Operational Strategy at 5% FPR**:

To operate at a 5% FPR, we need to consider the probability decision threshold of identifying a transaction as fraudulent. As the false positive rate decreases, the threshold increases because we want it to be harder to misclassify a legitimate transaction as fraudulent. Based on our best-performing model, our decision threshold is 0.1059 meaning that any transaction with a predicted probability higher than 0.1059 will be classified as fraudulent. While this is a low threshold, the data used to build the model did not contain many fraudulent transactions, so the occurrence of fraud was rare.

**Statistics at 5% False Positive Rate**

Threshold: 0.1059

Recall: 0.8218

Precision: 0.95

Because it is desired to operate at a 5% false positive rate and we want to be wary of misclassifying legit transactions as fraud, we recommend implementing a **more thorough review process for transactions**. The most telling variables were ones pertaining to the transactions themselves, such as the adjusted transaction amount, transaction environment, transaction type, and transaction initiate. There were quite a few transaction environments where all of the transactions were fraudulent. We recommend setting up an **alert system that notifies when those** **transaction environments occur**.