1. Task Description

Hi,

Now that you have finished your analysis for Credit One, I would like to propose a different project for you. Employers like to see candidates for Data Science positions demonstrate that they can solve a complex problem on their own and in a relatively short amount of time. An ideal candidate might need to demonstrate the following knowledge and skills:

GitHub
Python and all libraries needed to solve the problem
Exploratory Data Analysis
Data collection, pre-processing and feature engineering
Data Visualization
Data science process: Best practices
Predictive Modeling and Evaluation (the whole process)
Model selection
Cross validation

For this task, you'll need to choose your own data and design a Data Science project that uncovers three or more relevant findings in the data that address goals you have defined beforehand.

Your finished project should be accessible in the form of a link to your GitHub account. You will probably want to include work from earlier courses in your account so someone reviewing it won't just see one project and will be able to fully understand the potential I know you can bring to a Data Science role.

Since this is a capstone project of your own choosing and design there is no plan of attack for this task.

Best of luck,

Your Mentor

2. Project Overview

2.1 Project Justification

A typical organization loses an estimated 5% of its yearly revenue to fraud. The financial services industry and the industries that involve financial transactions are suffering from fraud-related losses and damages. 2016 was a banner year for financial scammers. In the US alone, the number of customers who experienced fraud hit a record 15.4 million people, which is 16 percent higher than 2015.

Fraudsters stole about $6 billion from banks last year. A shift to the digital space opens new channels for financial services distribution. It also created a rich environment for fraudsters.

Customer loyalty and conversions are affected in both environments, the digital and the physical. According to Javelin Strategy & Research, it takes 40+ days to detect fraud for brick-and-mortar financial institutions.

Fraud also impacts banks that provide online payments service. For instance, 20 percent of customers change their banks after experiencing scams. So, the challenge for industry players is to implement real-time claim assessment and improve the accuracy of fraud detection.

The machine learning (ML) approach to fraud detection has received a lot of publicity in recent years and shifted industry interest from rule-based fraud detection systems to ML-based solutions. The main differences between these two methods are:

Rule-based vs ML-based Fraud Detection Systems

Rule-based fraud detection	ML-based fraud detection
Catching obvious fraudulent scenarios	Finding hidden and implicit correlations in data
Requires much manual work to enumerate all possible detection rules	Automatic detection of possible fraud scenarios
Multiple verification steps that harm user experience	The reduced number of verification measures
Long-term processing	Real-time processing

Figure 1 – Differences of Rule-based vs ML-based fraud detection systems

The rule-based approach: Fraudulent activities in finance can be detected by looking at on-surface and evident signals. Unusually, large transactions or the ones that happen in atypical locations obviously deserve additional verification. Purely rule-based systems entail using algorithms that perform several fraud detection scenarios, manually written by fraud analysts.

Today, legacy systems apply about 300 different rules on average to approve a transaction. That's why rule-based systems remain too straightforward. They require adding/adjusting scenarios manually and can hardly detect implicit correlations.

ML-based fraud detection: However, there are also subtle and hidden events in user behavior that may not be evident, but still signal possible fraud. Machine learning allows for creating algorithms that process large datasets with many variables and help find these hidden correlations between user behavior and the likelihood of fraudulent actions. Another strength of machine learning system compared to rule-based ones

is faster data processing and less manual work. Machine learning detection is divided in common (anomaly detection) and advanced detection systems.

Anomaly detection: It's one of the common anti-fraud approaches in data science. It is based on classifying all objects in the available data into two groups: normal distribution and outliers. Outliers, in this case, are the objects (e.g. transactions) that deviate from normal ones and are considered potentially fraudulent.

Advanced fraud detection: Advanced systems aren't limited to finding anomalies but, in many cases, can recognize existing patterns that signal specific fraud scenarios. There are two types of machine learning approaches: unsupervised and supervised machine learning. They can be used independently or be combined to build more sophisticated anomaly detection algorithms.

Supervised learning entails training an algorithm using labeled historical data. In this case, existing datasets already have target variables marked, and the goal of training is to make the system predict these variables in future data.

Figure 2 –ML-based fraud detection systems (Supervise and Unsupervised algorithms)

Unsupervised learning models process unlabeled data and classify it into different clusters detecting hidden relations between variables in data items.

2.2 Project Description

The project will use machine learning algorithms to determine the pros and cons of supervised and unsupervised fraud detection methods.

One data (creditcard_sampledata_2.csv) set will be used to develop the model with supervised learning approach and two data sets (banksim.csv and banksim_adj.csv) will be used to developed the unsupervised learning model. Both models will be enhanced and then compared, some quality metrics (precision, recall and ROC curve) will be used to determine the most suitable model or combination of models.

3. Task Solution

3.1 Fraud Detection with supervised learning

This section will describe the sequence of steps and the results of the different algorithms used to detect fraud. A data set containing 31 columns and 7300 rows.

3.1.1 Models configuration:

Each model contains the following processes and parameters:

- Pre-processing
- Data Partition (70/30)
- Performance evaluation
- Ensemble

3.1.2 Data preprocessing, modelling and evaluation

The flow chart listed below (Figure 3) shows the preprocessing steps defined to pre-process the data, build and tune all the algorithms, the model evaluation is used to determine the best performer algorithm.

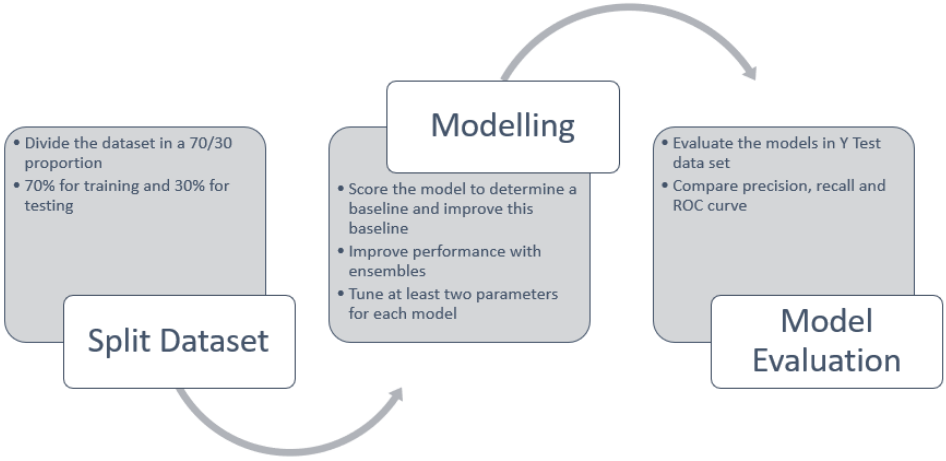


Figure 3 – Workflow for Credit One Dataset

3.1.3 Results

The exploratory analysis (Left image of figure 4) shows that 7000 transactions were not flagged as fraud, while 300 transactions were flagged as fraud. In the right image, the minority class (Class 1) is heavily imbalanced, against class 0 (Non-Fraud).

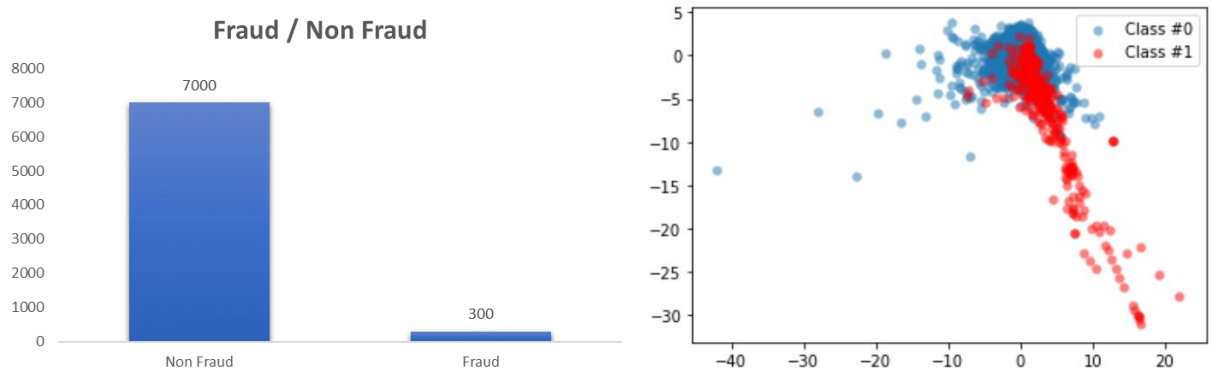
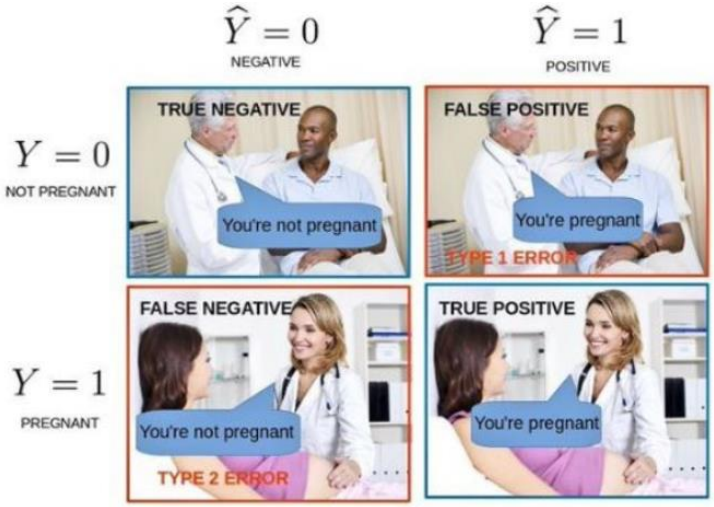


Figure 4 – Count of Fraud (class 1) / Non fraud (Class 0) transactions

The fraud rate is 4,11% against 95,89% of non-fraud transactions. Highly imbalanced data cannot be measured with accuracy. Therefore, confusion matrix, precision, recall and ROC curve are used to evaluate this test case.

The confusion matrix for the models is listed in Figure 5. There is no good or bad algorithm, to decide which final model is best, you need to take into account how bad it is not to catch fraudsters, versus how many false positives we can deal with.



	Predicted positive	Predicted negative
Actual positive	True positive (TP)	False negative (FN)
Actual negative	False positive (FP)	True negative (TN)

Model	True Positive	True Negative	False Positive	False Negative
Random Forest	2097	73	18	2
Tuned Random Forest	2095	76	15	4
Logistics Regression	2052	80	11	47
Voting Classifier	2090	78	13	9
Adjusting weights	2094	77	14	5

Figure 5 – Confusion Matrix Summary for models with supervised learning

Ultimately, this is a business decision. If our target is not to catch as many fraud cases as we can, whilst keeping the false positives low, the Adjusted Weights Voting Classifier or the Tuned Random Forest are pretty good deals.

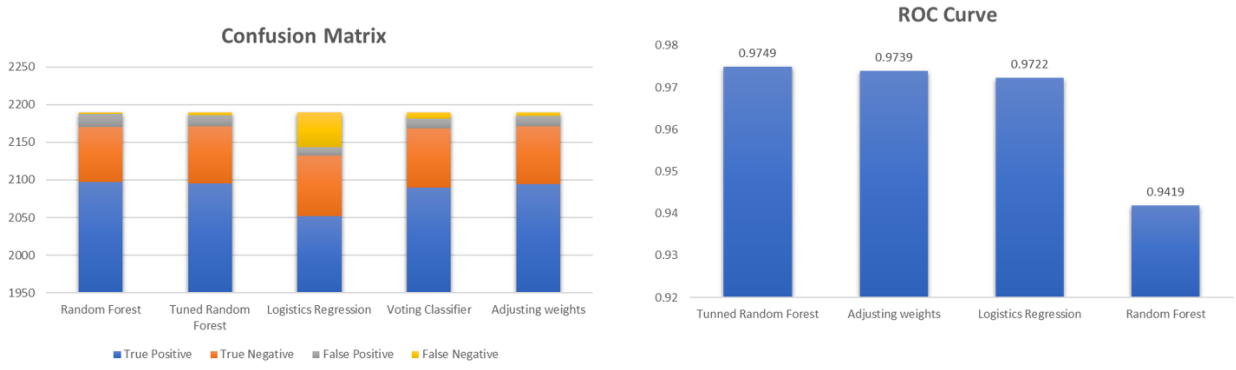


Figure 6 – Confusion Matrix and ROC curve for fraud detection models with supervised learning.

In the matrix listed above (Figure 5), we assumed that catching as many frauds as we can, whilst keeping the false positive low, its exactly what ROC curve tells us. In figure 6, The confusion Matrix of Random Forest and Adjusted Weights Voting Classifier shows high quantity of true positives and low quantities of false positive, in both cases the ROC curves have the higher values (0,9749 and 0,9739 respectively).

3.2 Fraud Detection with unsupervised learning

This section will describe the sequence of steps to analyze fraud with unsupervised data, using K-means clustering and other clustering algorithms to find suspicious occurrences in the data.

There are two datasets, one with 7200 rows and 5 columns (banksim.csv) and the other one with 7189 rows and 18 columns (banksim_adj.csv). The data set description is listed in the Jupyter Notebook.

3.2.1 Models configuration:

Each model contains the following processes and parameters:

- Pre-processing
- Data Partition (70/30)
- Performance evaluation

3.2.2 Data preprocessing, modelling and evaluation

The flow chart listed below (Figure 7) shows the preprocessing steps defined to pre-process the data, build and tune all the algorithms, the model evaluation is used to determine the best performer algorithm.

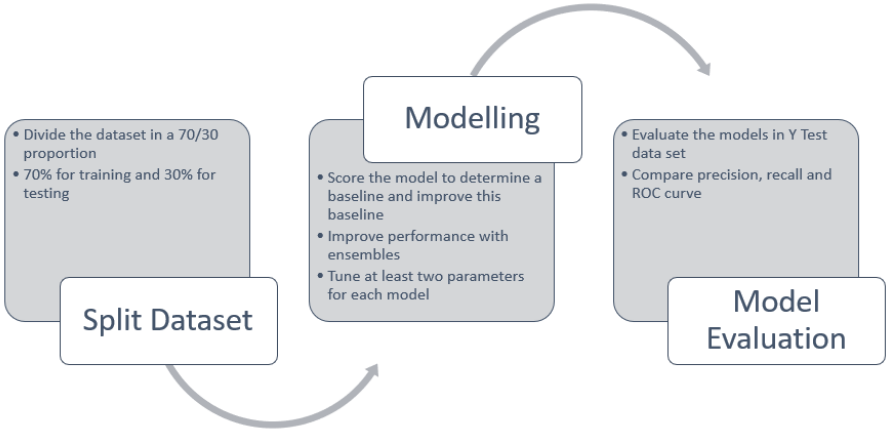


Figure 7 – Workflow for Credit One Dataset

3.2.3 Results

The EDA (see figure 8) shows that fraud by category is not correlated with the amount of spend per category. The top 3 fraud categories are leisure, travel and sport & toys.

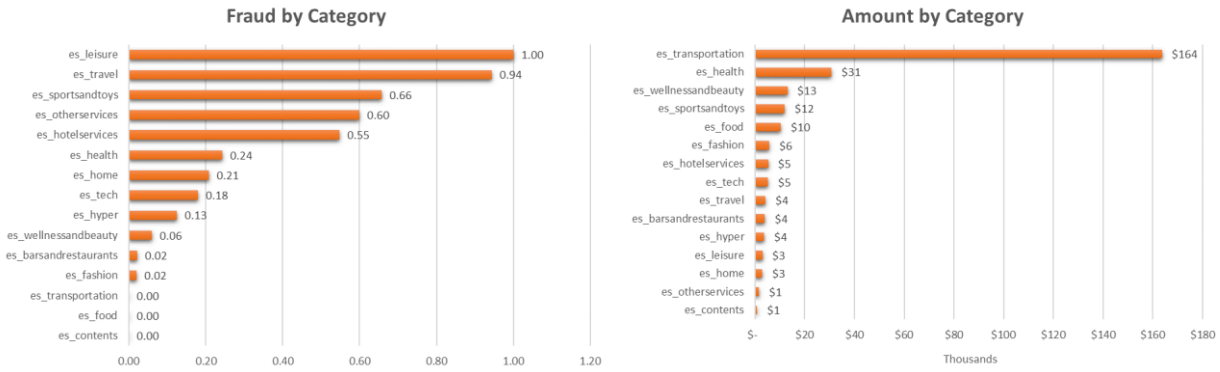


Figure 8 – Fraud by category and Amount by category

The method selected is K-means. The objective of k-means is to minimize the sum of all distances between the data samples and their associated cluster centroids. The Elbow method was used to determine the optimal number of clusters.

Figure 9 – Optimal number of clusters for K-means algorithm / Fraud confusion matrix.

The resulting optimal number is around 3 clusters and the confusion matrix show that 2029 transaction were flagged as Non-Fraud, whilst 38 transactions were Frauds. The type 2 error (false positive) is higher in this model (20) compared with the supervised learning model.

4. Results Discussion

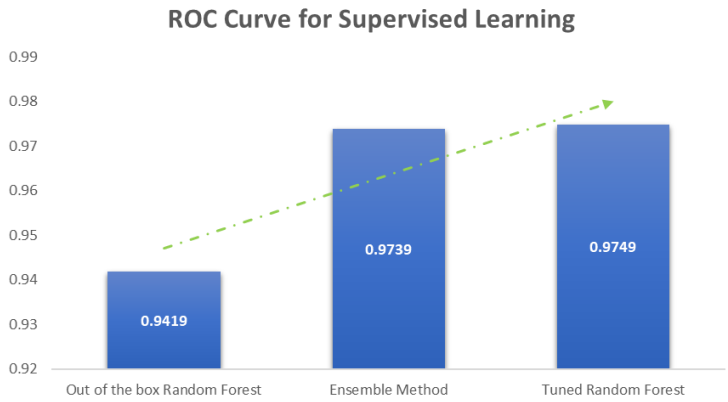


Figure 10 – Enhanced models through parameters tuning and ensemble methods

The models developed to analyze the fraud with supervised learning shows an increase in the ROC value, through the modification of parameters and combination of models with the ensemble method. Figure 10, shows a positive trendline of improvement with these techniques.

Figure 11, compares the ROC curve and True Positives / True Negatives for both methods. The supervised learning methods has a higher ROC curve value (0.9749 against 0.8109) a 20% difference. Additionally, the K-Means models catch less frauds (2029) than the tuned random forest (2095).

This is not a fair comparison, since both methods uses different approaches. The supervised learning uses a full set of labeled data while training an algorithm. A fully labeled means that each example in the training dataset is tagged with the answer the algorithm should come up with on its own.

While unsupervised learning uses the anomaly detection approach: Banks detect fraudulent transactions by looking for unusual patterns in customer's purchasing behavior.

For instance, if the same credit card is used in California and Denmark within the same day, that's cause for suspicion.

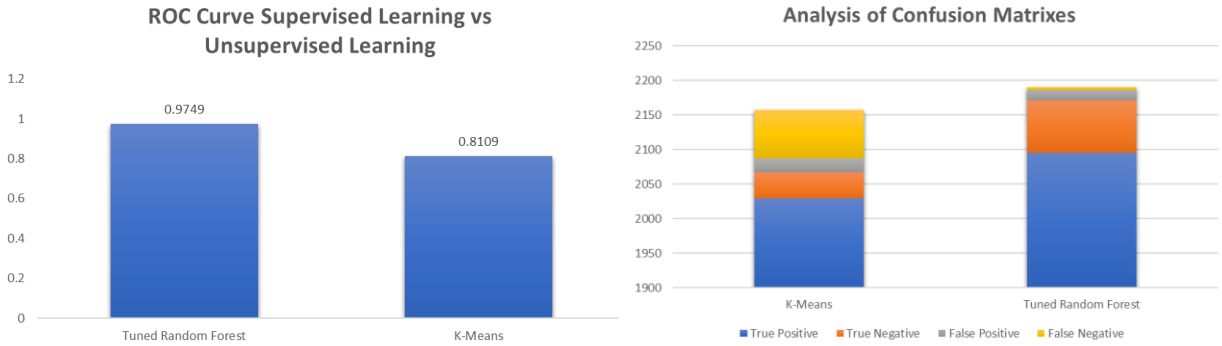


Figure 11 – Supervised learning vs unsupervised learning

The techniques used in this capstone projects tried to illustrate real problems and real situations in which either labeled data or unlabeled data are the options, sometimes we can get one or the other, rarely we can get both. As result the knowledge of these techniques became handy.

5. Recommendations

The first recommendation is aligned with the use of other methods that do not require the cluster calculation such as DBSCAN, which works by forming groups or clusters of points that are close to each other and leaving the remaining ones as noise points or outliers.

Secondly, because organized crime schemes are so sophisticated and quick to adapt, defense strategies based on any single, one-size-fits-all analytic technique will produce sub-par results. Each use case should be supported by expertly crafted anomaly detection techniques that are optimal for the problem at hand. As a result, both supervised and unsupervised models play important roles in fraud detection and must be woven into comprehensive, next- generation fraud strategies.

A supervised learning, uses models that are trained on a rich set of properly "tagged" transactions. Each transaction is tagged as either fraud or non-fraud. The models are trained by ingesting massive amounts of tagged transaction details in order to learn patterns that best reflect legitimate behaviors. When developing a supervised model, the amount of clean, relevant training data is directly correlated with model accuracy.

Unsupervised learning, uses models designed to spot anomalous behavior in cases where tagged transaction data is relatively thin or non-existent. In these cases, a form of self-learning must be employed to surface patterns in the data that are invisible to other forms of analytics.

Unsupervised models are designed to discover outliers that represent previously unseen forms of fraud. These AI-based techniques detect behavior anomalies by identifying transactions that do not conform to the majority. For accuracy, these discrepancies are evaluated at the individual level as well as through sophisticated peer group comparison. By choosing an optimal blend of supervised and unsupervised AI techniques you can detect previously unseen forms of suspicious behavior while quickly recognizing the more subtle patterns of fraud that have been previously observed across billions of accounts.

6. References

1- Fraud Detection: How Machine Learning Systems Help Reveal Scams in Fintech - https://www.altexsoft.com/whitepapers/fraud-detection-how-machine-learning-systems-help-reveal-scams-in-fintech-healthcare-and-ecommerce/
2- Demystfying Machine Learning for Banking - https://feedzai.com/wp-content/uploads/2017/03/DML-Final.pdf
3- Machine learning for fraud detection - https://www.ravelin.com/insights/machine-learning-for-fraud-detection

7. Python scripts

Provided in Jupyter Notebook and Github link.