Credit Card Fraud Detection

Introduction and Background

In the information age, sensitive user data such as payment information has become increasingly digitized. This has created vulnerabilities in data safety. In this project, we intend to implement an automated and intelligent detection system for fraudulent transactions.

Problem Definition

The evolution of physical cash to stored information sacrifices security for speed. According to a Federal Trade Commission 2021 report, 88,354 credit card fraud instances resulted in more than $180 million in losses. By using machine learning to detect fraud, we aim to minimize these financial burdens.

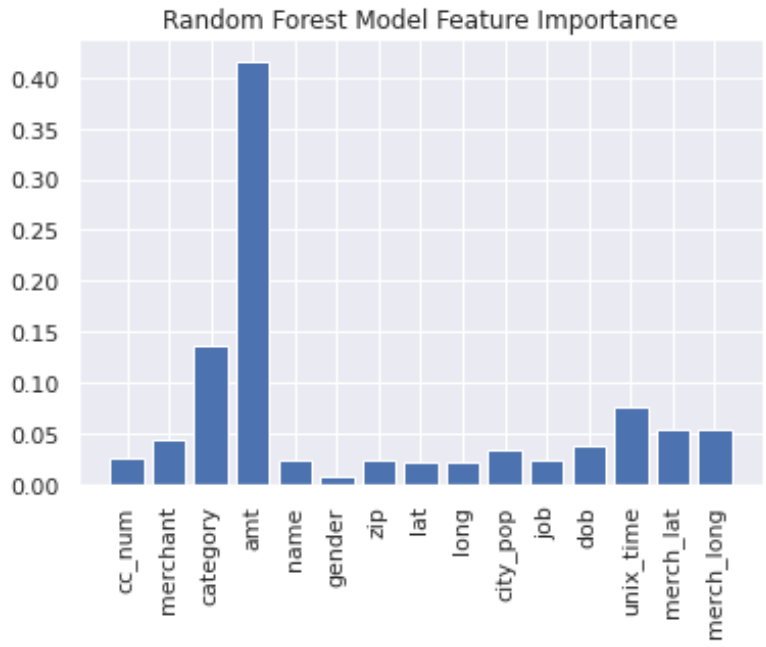
Data Collection

Our dataset is simulated consumer information by the MIT Licensed Synthetic Credit Card Transaction generator. It contains 555718 transactions, 22 features, and a single label, fraud vs. normal transactions.

Feature Selection & Data Preprocessing

In order to decrease the possibility of over-fitting in our models we limited the number of features used in our training dataset. Our first step was to do some preliminary feature reduction such as combining first_name and last_name into one feature name. Due to the surplus of qualitative features in our dataset, we first encoded all qualitative columns. Our encoding approach uses sklearn's preprocessing.LabelEncoder(), which encodes target labels with a value between 0 and n_classes - 1. This applied an integer representation of each unique value per feature.

Numerical data representation allowed us to implement a Random Forest Classifier to generate the most important features, measuring the average impurity decrease computed from all decision trees in the forest. We then determined that there were 8 relevant features that could be used based on the model below and our own predictions of what could affect the likelihood of fraud.



These are the features we decided to keep and their descriptions:

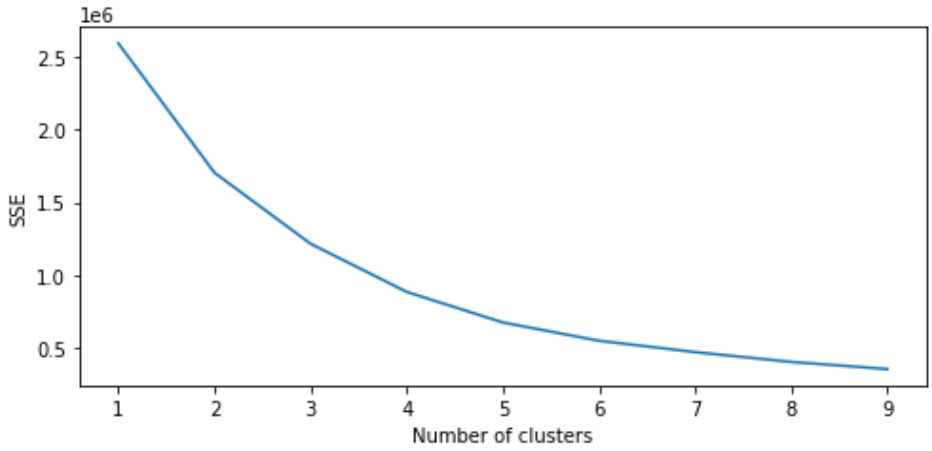
- amount - Represents the monetary value of the transaction.
- category - What kind of items were bought. We used one hot encoding to convert each category to a numerical label.
- unix_time - When the transaction was made.
- lat - The latitude coordinate of the card holder.
- lon - The longitude coordinate of the card holder.
- merch_lat - The latitude coordinate of the purchase.
- merch_lon - the longitude coordinate of the purchase.
- is_fraud - True/false label that indicates whether a transaction is fraudulent.

The other 14 features were determined to be not as meaningful to the detection of fraudulent cases and therefore discarded from the dataset.

Methods

Unsupervised Learning:

K-Means



To find the best k-value for our clustering model we implemented the elbow method, shown in the graph above. Based on the graph we found that the ideal number of clusters for our dataset was at a k-value of 4. This was determined based on the decreasing slope of the graph at k = 4, indicating that the error between our predicted and actual values decreased.

The model algorithm was run using the amount value of each transaction along with the distance between the credit card holder's location and the coordinates of the purchase. This distance was calculated using the lat, lon, merch_lat, and merch_lon features.

Local Outlier Factoring (LOF)

Another clustering algorithm we will consider is LOF. This algorithm is more adept at identifying outlier data points by calculating their density deviation (scikit, 2022). For our purposes, we can consider outliers as cases of fraud under the presumption that card transactions that dramatically deviate from clusters are less likely to be normal.

Supervised Learning:

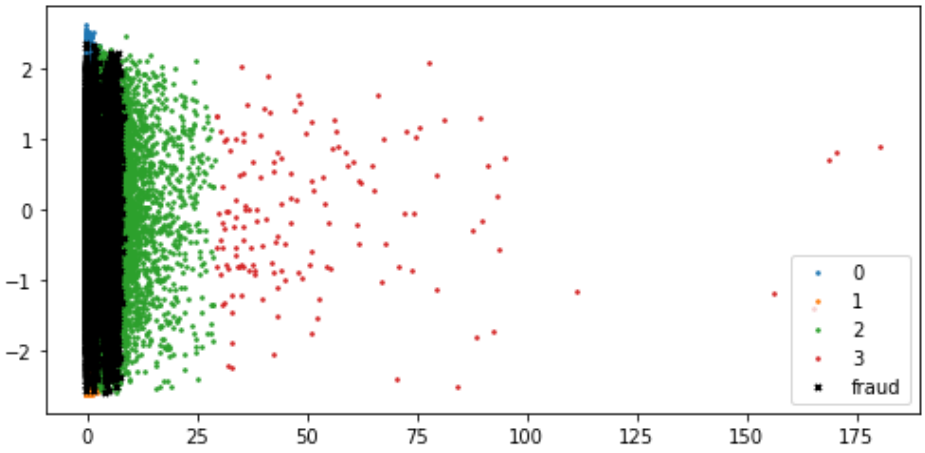
Artificial Neural Networks (ANN)

We will construct an ANN model in order to view any related patterns between fraudulent credit card transactions. Based on this we can create a predicted value range of transaction amounts for cases of fraud and use this as a basis for detection. An ANN model is well suited for this problem given how it learns using previously established patterns, and we will use the backpropagation algorithm to improve the neural network (Mittal et al., 2019).

Results and Discussion

Unsupervised Learning:

K-Means



The above figure is the end model using the k-means with a k-value of 4. Each cluster differs based on distance and amount values. Clusters 0 and 1 consist of low amount transactions and similar distance values, however the direction of cluster 0's distances are opposite of cluster 1. Cluster 2 is made up of medium amount values with no concentration on a specific distance range. Cluster 3 is similar in that it spreads across the entire range of distance values but consists of all high amount value transactions. The density of each cluster also differs greatly. Clusters 0 and 1 are very condensed within the low amount values 0-5. Cluster 2 is slightly larger in its amount range and cluster 3 is more spread out with greater distances between points.

Fraud cases are signified by an 'x' shape on the figure above. The majority of credit card transactions fell into clusters 0-2, the lower to mid value transactions. Likewise all fraudulent cases could also be found in these 3 clusters, with no fraudulent cases appearing in cluster 3, the highest valued card transactions. The fraud points are not concentrated across any of the distance values, implying that the distance between the location of the card holder and the location of the purchase does not have a significant relationship with whether or not the transaction is a case of fraud.

References

Mittal, S., & Tyagi, S. (2019, January). Performance evaluation of machine learning algorithms for credit card fraud detection. In 2019 9th International Conference on Cloud Computing, Data Science & Engineering (Confluence) (pp. 320-324). IEEE.

Outlier detection with local outlier factor (LOF). scikit. (n.d.). Retrieved October 6, 2022, from https://scikit-learn.org/stable/auto_examples/neighbors/plot_lof_outlier_detection.html#:~:text=The%20Local%20Outlier%20Factor%20(LOF,lower%20density%20than%20their%20neighbors.

Proposed Timeline:

Our Gantt chart can be found here

Contribution Table:

Team Member	Contribution
Samantha Burger	Results evaluation and analysis, Midterm Report
Olivia Lawson	Data Cleaning & Preprocessing, Midterm Report
Munim Riddhi	Model coding, Midterm Report
Rob Schleusner	Data Cleaning & Preprocessing, Midterm Report
Samuel Wysocki	Model coding, Midterm Report