Applied Analytics Project - Week 13 Assignment

Bring It All Together

Brady Levenson

ADAN 8888 - Applied Analytics Project

Professor Nurtekin Savas

December 8, 2024

Problem Statement

It is known that fraudulent financial transactions occur on a regular basis, but it is difficult to distinguish between them and legitimate financial transactions simply by viewing the transactions. The inability to correctly classify fraudulent financial transactions can promote uncontrolled and untraceable levels of financial losses for banks, investment firms, corporations, and numerous other entities. We are given a dataset of financial transactions and their unique properties. We will analyze this dataset using various machine learning techniques to determine effective measures to identify historical fraudulent transactions and prevent their occurrence in the future.

The Model Development Life Cycle

The model development life cycle consists of the necessary steps to completely develop a model from start to finish, beginning with an initial phase that involves choosing a problem statement and gathering data that may support a study of that problem statement and proceeding all the way to deploying a model with real world implications of solving that problem statement.

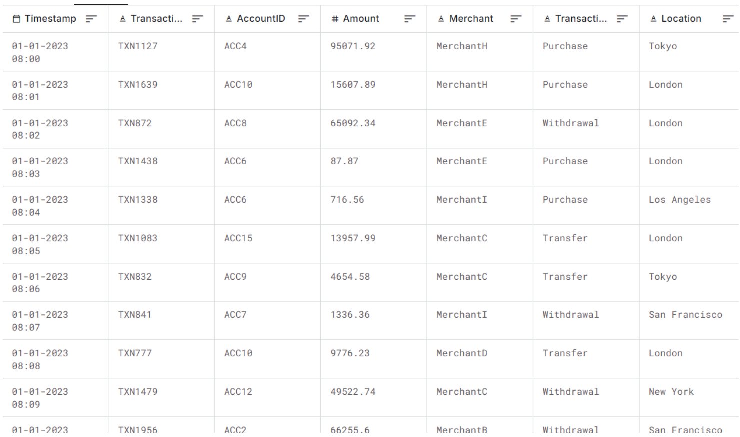
Major Steps - What Was Done?

Choosing a Problem Statement

We began this project by selecting a problem statement that had the potential to solve a real world problem and produce positive business outcomes in the process.

Data Collection

We found a dataset that we believed had the potential to support the resolvement of our problem statement.



Timestamp	Transacti...	AccountID	Amount	Merchant	Transacti...	Location
01-01-2023 08:00	TXN1127	ACC4	95071.92	MerchantH	Purchase	Tokyo
01-01-2023 08:01	TXN1639	ACC10	15607.89	MerchantH	Purchase	London
01-01-2023 08:02	TXN872	ACC8	65092.34	MerchantE	Withdrawal	London
01-01-2023 08:03	TXN1438	ACC6	87.87	MerchantE	Purchase	London
01-01-2023 08:04	TXN1338	ACC6	716.56	MerchantI	Purchase	Los Angeles
01-01-2023 08:05	TXN1083	ACC15	13957.99	MerchantC	Transfer	London
01-01-2023 08:06	TXN832	ACC9	4654.58	MerchantC	Transfer	Tokyo
01-01-2023 08:07	TXN841	ACC7	1336.36	MerchantI	Withdrawal	San Francisco
01-01-2023 08:08	TXN777	ACC10	9776.23	MerchantD	Transfer	London
01-01-2023 08:09	TXN1479	ACC12	49522.74	MerchantC	Withdrawal	New York
01-01-2023	TXN1956	ACC2	66255.6	MerchantB	Withdrawal	San Francisco

Data Cleaning/ Preprocessing

We cleaned and preprocessed our chosen dataset to make it more robust for modeling purposes.

Exploratory Data Analysis

We conducted EDA to gain a better understanding of the core elements of our dataset at hand.

Feature Engineering

We constructed new features based on those present in our original one to explore underlying trends and correlations in our dataset prior to modeling.

Model Building

We constructed three different types of models that we believed could potentially perform our classification task at a high level and considered multiple hyperparameter configurations for each model type.

Validation Set Approach for Determining Best Constructed Model

We evaluated each of our model variations on a validation set to prospect how the models may perform in a live environment while taking measures to prevent data leakage.

Evaluation of Winning Model on Unseen Test Data

We selected the model that exhibited the highest performance on validation set data to move forward with and applied this model on unseen test data (also with data leakage prevention considerations).

Consideration of Data-Centric AI

We explored Data-Centric AI approaches to attempt to improve our model’s performance through the direct manipulation of our dataset.

Analysis of Risk, Bias, and Ethical Considerations

We analyzed the risk, bias, and ethical considerations associated with deploying our model in the real world.

Model Deployment

We determined the specific conditions that would need to be in place in order for us to move forward with our model deployment.

Why Did We Take the Aforementioned Steps?

Choosing a Problem Statement

We chose a problem statement pertaining to financial fraud detection because we saw the issue of financial fraud as one that causes significant losses to businesses and stakeholders year over year. Additionally, we believed that financial fraud detection systems currently in place could benefit from new modeling approaches that have the potential to reduce the immense financial losses continually incurred from fraud.

Data Collection

We needed to find a dataset that could serve as a starting template for the modeling task we were planning to execute. The financial anomaly detection dataset from Kaggle contained a large number of financial transactions (216,960) and enough distinct features to conduct a variety of machine learning-based analyses that may support the finding of subtle discrepancies in certain financial transactions that may be indicative of fraud.

Data Cleaning/ Preprocessing

We needed to prepare the data for our modeling task. We did this by removing noncontributory features and null values and reconfiguring existing features to be more suited for machine learning model input.

Exploratory Data Analysis

We performed EDA to gain a strong basis for what modeling techniques may be most effective in addressing our problem statement.

Feature Engineering

We performed feature engineering on our dataset to produce new features that better reflect the nature of the data than the original raw data could on its own. We implemented dimensionality reduction techniques on our newly produced dataset so it could be analyzed more efficiently by our chosen models.

Model Building

We built multiple types and variations of machine learning models that have the capability to perform anomaly detection tasks. We created a hierarchy of models based on complexity from simple to moderate to complex to gain a comprehensive understanding of how model complexity level affects model performance as well as to ensure that we would be moving forward with a modeling approach that outperformed numerous other model variations.

Validation Set Approach for Determining Best Constructed Model

We implemented a train-validation-test set approach to this anomaly detection task so that we could determine the highest quality model for the classification task through its implementation on validation set data. This allowed us to move forward with a strong model without having to test numerous different models in a live environment where there is more risk and concerns associated with erroneous models entering industry settings.

Evaluation on Unseen Test Data

We evaluated our selected “winning model” on a test dataset to simulate how the model would perform in a live environment where it would be taking real financial transactions as input in real time.

Consideration of Data-Centric AI

After selecting a model that we felt strongly about, we performed data-centric AI to attempt to improve model performance through dataset manipulation. This process allows us to determine what combinations and portrayals of features of real financial transaction data should be input to our model to attain high-quality results.

Analysis of Risk, Bias, and Ethical Considerations

We analyzed the risk, bias, and ethical considerations of our model to ensure that our model was fair and equitable and to mitigate the potential for future ramifications of deploying the model including legal and reputation-based concerns.

Model Deployment

We determined specific circumstances represented by green, yellow, and red zones that correspond with the manner with which we would want to deploy our model at any given time. This process was undertaken to ensure that we had a monitoring plan in place that could tell us when we can confidently deploy the model without issue, when we should closely monitor the model and potentially decide on ways to improve the model, and when we should immediately pull the model out of production.

How Did We Execute Them?

Choosing a Problem Statement

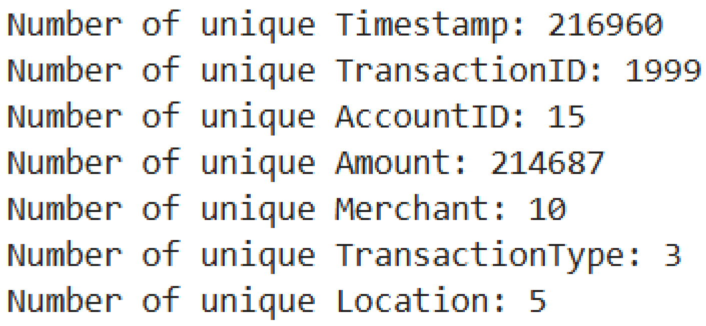
To choose a problem statement geared towards solving a real world problem, I began by broadly considering large industries such as finance, healthcare, technology, and real estate, among others. I wanted to conduct a task related to my future career aspirations and decided the finance industry would be the best platform for me to do this with respect to. From there, I considered numerous issues present in this industry, including compliance challenges, market volatility, and inflation to name a few, and I decided that fraud detection was a prevalent one that could be leveraged to produce large financial gains through loss prevention.

Data Collection

We searched the internet for datasets consisting of large volumes of financial transaction data points and enough unique features to conduct a comprehensive analysis based on. Kaggle's Financial Anomaly Dataset checked each of these boxes, which made it a strong candidate to facilitate our modeling task.

Data Cleaning/ Preprocessing

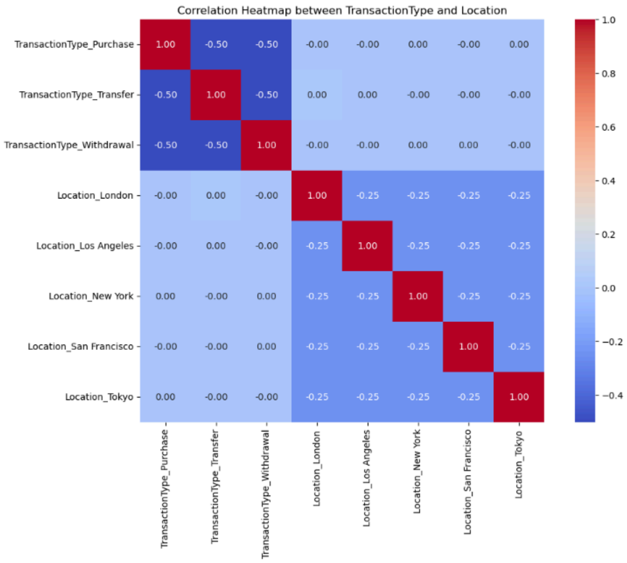
We cleaned and preprocessed our data by removing variables that were not contributory to our modeling task, adding variables that were, and only retaining data that we found to be relevant. We targeted features that took on too many unique values to analyze efficiently to drop, and we added features that organized our data better including buckets for transaction dollar amounts and specific time increments.



```
Number of unique Timestamp: 216960
Number of unique TransactionID: 1999
Number of unique AccountID: 15
Number of unique Amount: 214687
Number of unique Merchant: 10
Number of unique TransactionType: 3
Number of unique Location: 5
```

Exploratory Data Analysis

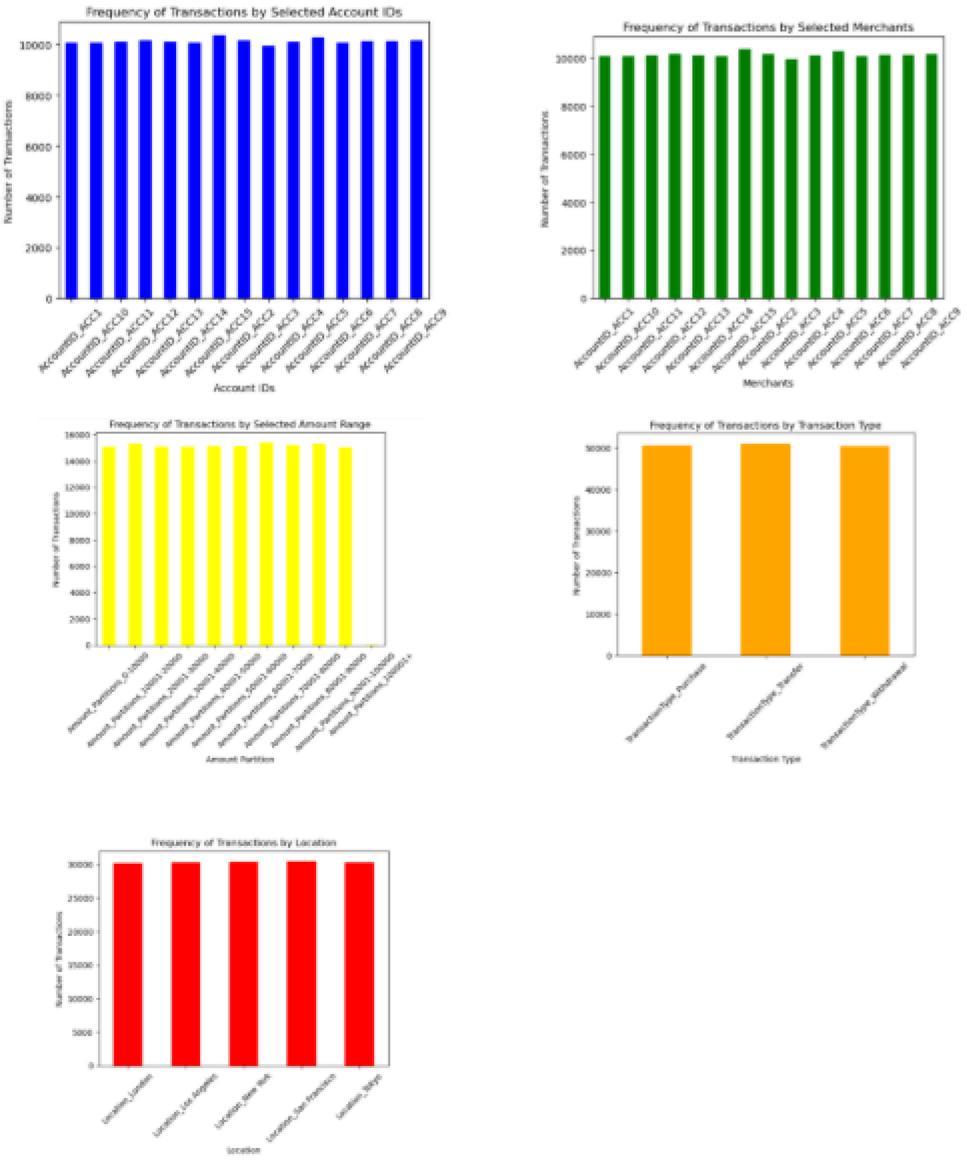
To conduct EDA, we performed correlation analyses to gain an idea of the relative correlation of certain features in the dataset and analyzed descriptive statistics such as frequency and mean to gain a better idea of the inner workings of this dataset.



Feature Engineering

We engineered features designed to address concerns relating directly to financial anomaly detection. The features we engineered at this stage were Hour_Group (groups transaction by six hour time increments of the day), Hour_Group_AccountID (groups transactions made by specific accounts in specific time intervals of the day), AccountID_Mean (calculates the mean transaction amount by a given account), AccountID_std (calculates the standard deviation of transactions made by a given account), and Deviation_From_Mean_Account (calculates the number of standard deviations a given transaction amount is from its account's mean transaction amount). We then performed one-hot encoding to provide numerical representations of our dataset's categorical variables (AccountID, Merchant, TransactionType, Location, Amount_Partitions, Day) and embedding to reduce the dimensionality of our large dataset while retaining its vital statistical properties.

Aggregated Counts of One-Hot Encoded Variables

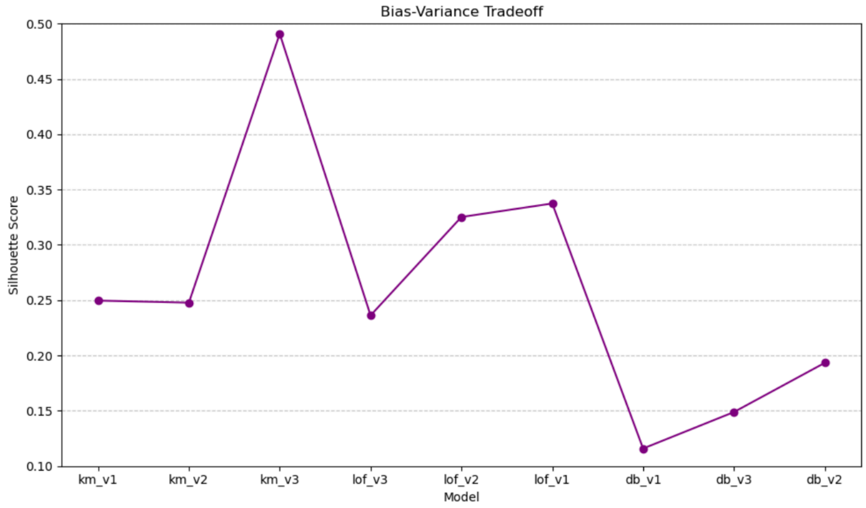


Model Building

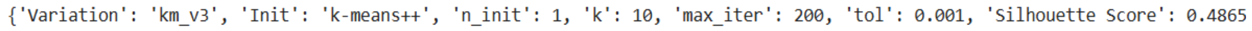
We researched anomaly detection algorithms of varying complexity levels and decided to build models based around K-Means Clustering (lowest complexity), Local Outlier Factor (moderate complexity), and Density-Based Spatial Clustering of Applications with Noise (highest complexity). We manipulated the core hyperparameters of these models to produce variations of each model type to determine which ones would perform the best at identifying anomalous financial transactions.

Validation Set Approach for Determining Best Constructed Model

We performed a 70/15/15 split for train/validation/test sets to allow for plenty of data to train our models on while leaving enough data to attain viable outputs when applying our models to validation and test sets. Operations on our respective train, validation, and test datasets were performed separately in a pipeline fashion to prevent data leakage that could lead to significant biases in our final results. Our primary model evaluation metric was silhouette score, with higher scores indicating data points being more similar to data in their own cluster than neighboring clusters, signifying more well-defined separation of clusters and "easier" anomaly detection. We decided to move forward with one of our K-Means Clustering models as it achieved the highest silhouette scores on average compared to our other models.

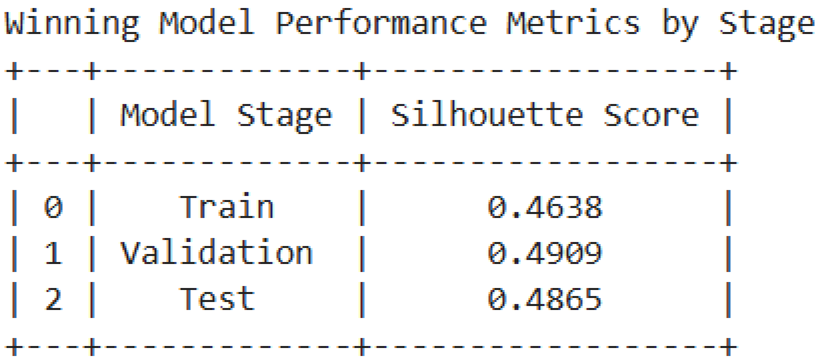


Evaluation on Unseen Test Data



```
{'Variation': 'km_v3', 'Init': 'k-means++', 'n_init': 1, 'k': 10, 'max_iter': 200, 'tol': 0.001, 'Silhouette Score': 0.4865
```

Our winning model performed moderately well on unseen test data and similarly on each of the train and validation datasets. This is exemplary of a finely hyperparameter-tuned K-Means Clustering Model's potential to perform adequately on unseen data that has been formatted efficiently, preemptively. Based on the Silhouette Score Performance Metric used to evaluate the model's performance on test data, it appears the model has some predictive capabilities for this classification task, though it is not perfect by any means and does have limitations.



```
Winning Model Performance Metrics by Stage
+---+-------------+------------------+
|   | Model Stage | Silhouette Score |
+---+-------------+------------------+
| 0 |    Train    |      0.4638      |
| 1 | Validation  |      0.4909      |
| 2 |    Test     |      0.4865      |
+---+-------------+------------------+
```

Consideration of Data-Centric AI

We considered the data-centric AI approaches Synthetic Minority Over-Sampling Technique (synthetically produce more prospectively anomalous samples with the intent of gaining a better understanding of this class), Principal Component Analysis (identify linear combinations present in the dataset while retaining 95% of the dataset's variance) , and Gaussian Noise Injection (artificially add noise to the dataset at hand with the intent of forcing a given model to generalize to make conclusions rather than memorizing patterns in the data). The manner in which we implemented these techniques did not directly lead to improvements in model performance, though it is an approach worth notable reconsideration upon model deployment especially in instances where the finance industry may be experiencing data or concept drift.

Analysis of Risk, Bias, and Ethical Considerations

While this model has proven to have moderate potential to successfully identify potentially fraudulent financial transactions in real time and flag them to undergo further review, it poses some considerable ethical risks that stakeholders need to take into account when deciding if this modeling approach is practical for their specific use. One primary concern is the model acting in a discriminatory manner towards people of certain protected categories including but not limited to elderly people, men or women, or people of certain ethnicities due to our dataset's inclusion of variables that may have correlation with protected classes. This factor alone could decentivize stakeholders from deploying the model or make them want to work with the developers of the model to remove inherent biases. Using a model that could contain biases of this variety may also result in serious legal ramifications including lawsuits or fines. If a stakeholder is found to have used this model in a discriminatory manner, they may be subject to these legal consequences. Beyond unfair practices and legal aspects, stakeholders may also avoid using this model out of a desire to preserve their reputations and images. By utilizing potentially unfair systems or services, stakeholders may gain bad rapport with their customers or other close industry associates.

Model Deployment

We determined that this model may be deployable in live financial industry settings with a comprehensive monitoring plan in place. Our model would constantly calculate the silhouette score of groups of real-time financial transactions, with this calculation representing a determining factor of whether the model should stay in production without obvious reservation (green), be monitored closely (yellow), or be removed from production altogether (red). If possible, it may also be beneficial to obtain associated data based on protected classes, as touched on in the previous section, to ensure that no features being used in our final dataset are too closely related to any of those protected classes to be used in our model.

Monitoring Plan Threshold Values:

Green: silhouette_score > 0.5
Yellow: 0.3 ≤ silhouette_score ≤ 0.5
Red: silhouette_score < 0.3

Conclusions and Closing Remarks

Our finely tuned K-Means Clustering Model has exhibited moderate potential to perform well at identifying fraudulent financial transactions in real time in financial industry settings. By adhering to a strict monitoring plan and being attentive to ethical concerns and regulations, our model can be leveraged by banks, investment firms, corporations, and numerous other entities to identify and prevent financial fraud from occurring and consequently save substantial sums of financial funds.