CS667: Practical Data Science - Project 3 Report

by

Joshua Gottlieb

Background

The data provided for this project is a financial anomaly dataset. There were 216,960 records. The dataset contained the following columns:

- Timestamp of transaction, precise to the minute
- Transaction ID, a "unique" identifier for each transaction
- Account ID, an identifier for the account associated with each transaction
- Merchant, identifying the merchant associated with the transaction
- Transaction Type, whether the transaction was a Purchase, Withdrawal, or Transfer
- Location, the city of origin for the transaction
- Amount, the amount of the transaction (no units given)

Notably, the financial anomaly dataset was provided without any labeled anomalies or background information. As such, this was a completely unsupervised learning task, and without any input from a subject matter expert, the identification of anomalies was based on best estimation without any ground truth labels or domain-specific insights as guidance.

Data Description, EDA, and Data Preprocessing

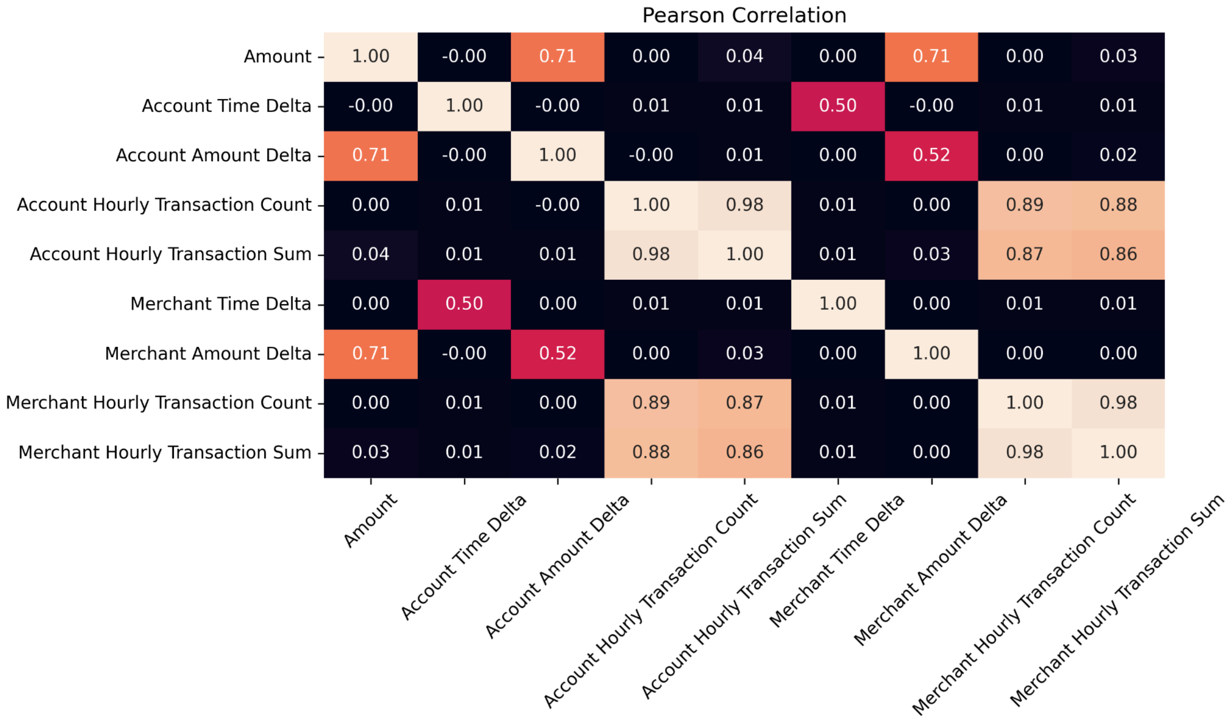
The Transaction ID column was not actually unique, despite what was indicated in the data descriptions. This may mean that Transaction ID represents groups of transactions, or it may just be a sign of poor data quality. Transaction ID was dropped, as it is meant to only be an identifier column and was not used for modeling.

Many time-based features were extracted from a combination of the Timestamp, Account ID, and Merchant columns. They are listed below:

- Month, Day (of month), Hour, Minute, and Weekend Flag
- Account Time Delta: For each account, the time elapsed, in minutes, since the prior transaction. This feature is meant to capture accounts with unusually high or low activity or with outlier bursts of high activity.
- Account Amount Delta: For each account, the difference between the current transaction and prior transaction. This feature is meant to capture accounts with highly varying transaction amounts.
- Account Hourly Transaction Count: For each account, the hourly summed transaction count. Note that this is not a rolling sum and was split at the start of each hour. This is meant to capture patterns in transaction count history and highlight hour-long windows where account activity is unusually high.
- Account Hourly Transaction Sum: Similar to the above feature, the hourly summed transaction amount. This is not a rolling sum and was split at the start of each hour. This feature captures hour-long windows with unusually high transaction sums, allowing anomalous periods composed of many smaller transactions to be detected.

- The same four features were also created, aggregated by Merchant: Merchant Time Delta, Merchant Amount Delta, Merchant Hourly Transaction Count, Merchant Hourly Transaction Sum.

After extracting the above features, the Timestamp column was dropped. The correlation matrix for numeric features is below.

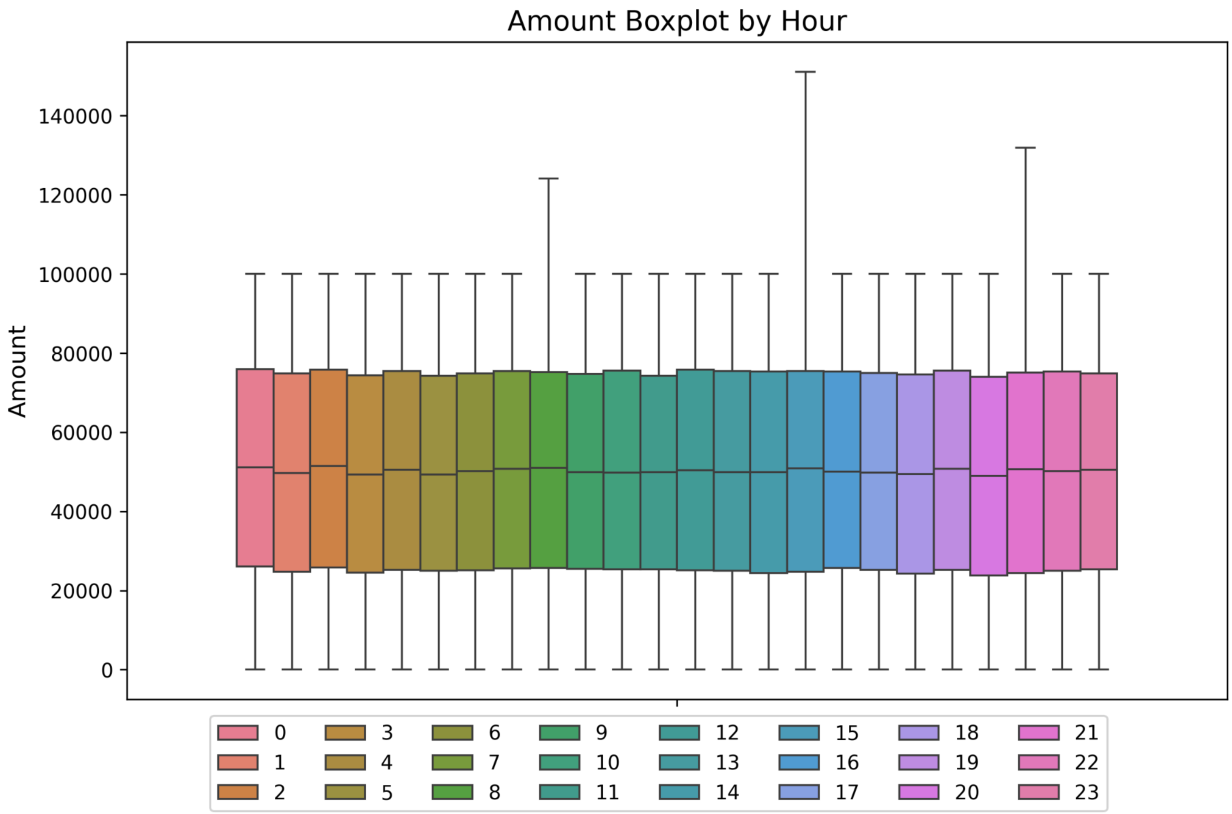


	Amount	Account Time Delta	Account Amount Delta	Account Hourly Transaction Count	Account Hourly Transaction Sum	Merchant Time Delta	Merchant Amount Delta	Merchant Hourly Transaction Count	Merchant Hourly Transaction Sum
Amount	1.00	-0.00	0.71	0.00	0.04	0.00	0.71	0.00	0.03
Account Time Delta	-0.00	1.00	-0.00	0.01	0.01	0.50	-0.00	0.01	0.01
Account Amount Delta	0.71	-0.00	1.00	-0.00	0.01	0.00	0.52	0.00	0.02
Account Hourly Transaction Count	0.00	0.01	-0.00	1.00	0.98	0.01	0.00	0.89	0.88
Account Hourly Transaction Sum	0.04	0.01	0.01	0.98	1.00	0.01	0.03	0.87	0.86
Merchant Time Delta	0.00	0.50	0.00	0.01	0.01	1.00	0.00	0.01	0.01
Merchant Amount Delta	0.71	-0.00	0.52	0.00	0.03	0.00	1.00	0.00	0.00
Merchant Hourly Transaction Count	0.00	0.01	0.00	0.89	0.87	0.01	0.00	1.00	0.98
Merchant Hourly Transaction Sum	0.03	0.01	0.02	0.88	0.86	0.01	0.00	0.98	1.00

The amount and time delta columns for both account and merchant are moderately correlated. Amount delta columns are also moderately correlated with amount, which makes sense, as differences in amounts are larger when small transactions follow large transactions and vice-versa. The hourly transaction count and sum columns are highly correlated, indicating high multicollinearity. It should be noted that unlike normal modeling, low correlation does not necessarily indicate a useless feature. Most data points are normal and should thus be uncorrelated, and the correlation with anomalous points may be masked by the low presence of anomalous points.

Using the variance inflation factor, the Account and Merchant Hourly Transaction Count features were dropped for being too closely related to the Account and Merchant Hourly Transaction Sum features. It was decided that the sum of hourly transactions was more likely to indicate anomalous patterns, as high volume transactions are only generally concerning if the total amount of money being moved is high. Gaussian Mixture Models are sensitive to collinearity, so it was important to remove collinear columns. The time delta columns were dropped after investigation, as the features did not appear to capture any data that was suspicious.

The categorical variables were tested using a Kruskal-Wallis test with respect to amount, which tests for differences in group distributions by categorical variable. The only significant variable was the hour of the transaction.



There are some meaningful differences between the medians and spread of amounts by hour. All other categorical variables were insignificant (p > 0.05) by the Kruskal-Wallis test. Rather than drop all of the categorical variables, two feature sets were tested.

- Full Feature Set
  - Numeric: Amount, Account Amount Delta, Merchant Amount Delta, Account Hourly Transaction Sum, Merchant Hourly Transaction Sum
  - Categorical: Account, Merchant, Transaction Type, Location, Month, Day, Hour, Minute, Weekend.
- Reduced Feature Set:
  - Numeric: Amount, Account Amount Delta, Merchant Amount Delta, Account Hourly Transaction Sum, Merchant Hourly Transaction Sum
  - Categorical: Hour

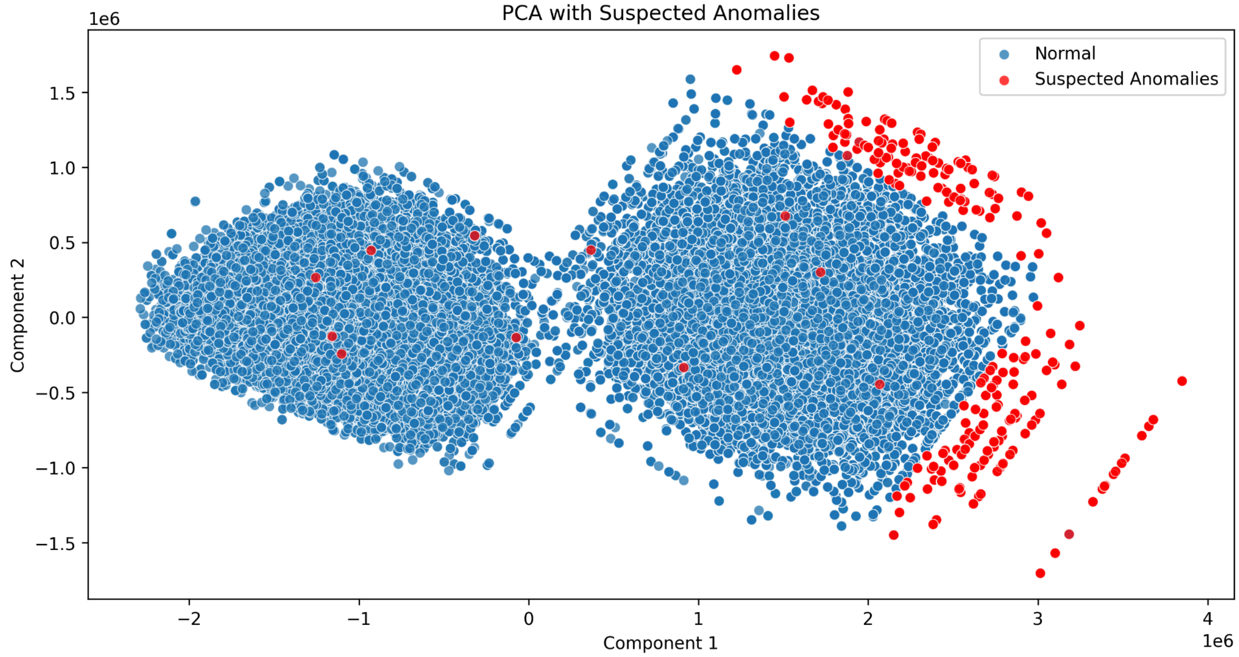
For encoding, the numeric features were min-max scaled. Min-max scaling was chosen since min-max scaling preserves the distribution of values, keeping outliers as outliers. Given that the task is anomaly detection, it is important not to obscure potential anomalous points using scaling algorithms that change the data distributions, like standardization. Time-based features were kept in ordinal scale. Other categorical features were one-hot encoded due to low cardinality and no innate sense of order. The data was split into 80% training and 20% test for modeling and evaluation.

Suspected Anomalies

A set of suspected anomalies was constructed by investigating the distributions of numerical variables and using quantile based outlier detection. The following data points were marked as anomalies:

- Amount values greater than 100,000. The distribution of amounts less than 100,000 was practically uniform (which is unusual itself, possibly indicating synthetic data), while the amounts above 100,000 were far more spread out, ranging from 109,000 to 978,942. These data points were identified by taking the 99th percentile of data points with amounts greater than 99,000. 14 data points were flagged as anomalies due to outlier amount values.
- 99.668% of all values had account hourly transaction sums less than 3.5 million. 720 data points were flagged as anomalies with account hourly transaction sums greater than 3.5 million.
- 99.669% of all values had merchant hourly transaction sums less than 4.8 million. 718 data points were flagged as anomalies with merchant hourly transaction sums greater than 4.8 million.

The suspected anomalies were confirmed using Principal Component Analysis (PCA) projection of the data to two dimensions for visual inspection.



There are two main "clusters" under this PCA projection. The right cluster has a ring of suspicious data points, not all of which were flagged as suspicious anomalies. There is also a small "island" on the bottom right of the graph. The inner points that were flagged as anomalies were the anomalies by excessive amount values. The top right flagged anomalies were the anomalies by account hourly transaction sum. The bottom right flagged anomalies were the anomalies by merchant hourly transaction sum. There were 1,452 suspected anomalies discovered through EDA, although visually, there is a high suspicion that the data points on the left outer edges of the right cluster are also likely anomalies.

Methodology

Two model types were tested: Gaussian Mixture Model (GMM), and Isolation Forest (IForest). For each model, 40 different hyperparameter combinations were tested for each of the reduced and full datasets, resulting in 80 models per model type, and 160 models overall. Because this is an unsupervised task, there are no default metrics, and no cross-validation was performed.

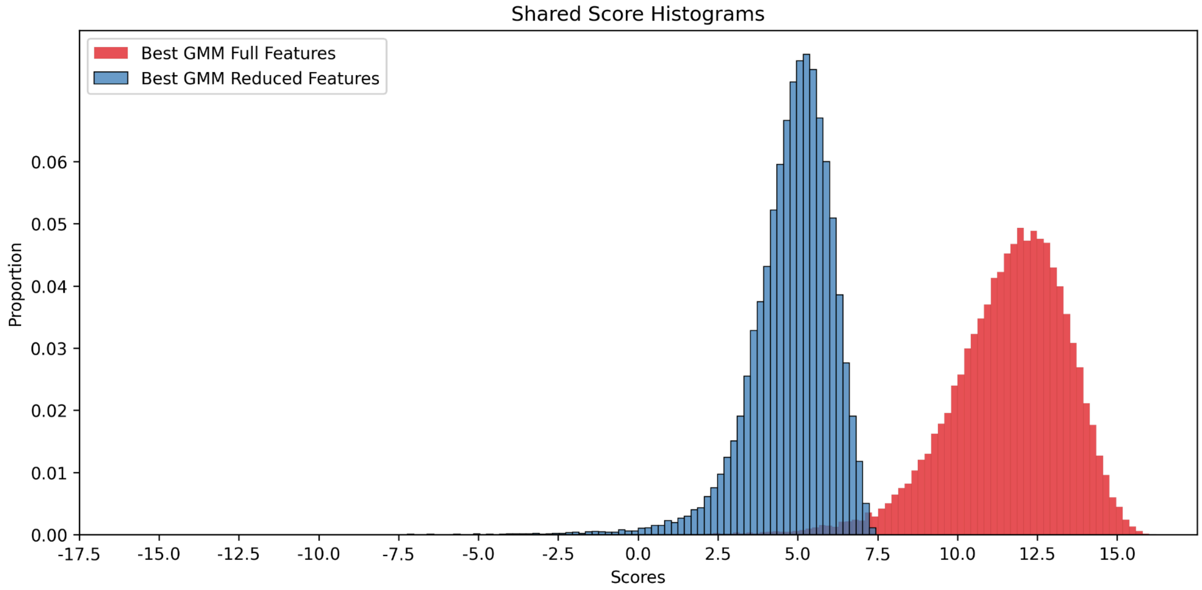
GMM models try to assign each data point to one of the clusters (controlled by the n_components parameter). Because we are using GMM for anomaly detection, we do not want predictions for components; rather, we want predictions for anomalies vs. non-anomalies. Instead, we calculated the log-likelihoods for each sample, and samples with low log-likelihoods were labeled as anomalies, since they are less likely to be generated by the GMM. The threshold used was a quantile based on the "contamination rate", which is simply the ratio of expected anomalies in the dataset. Since there are no labeled anomalies, we used the ratio of suspected anomalies in the training dataset as the base contamination rate. The GMM models were tested with n_components ranging from 2 to 20 and with each of the four available covariance types.

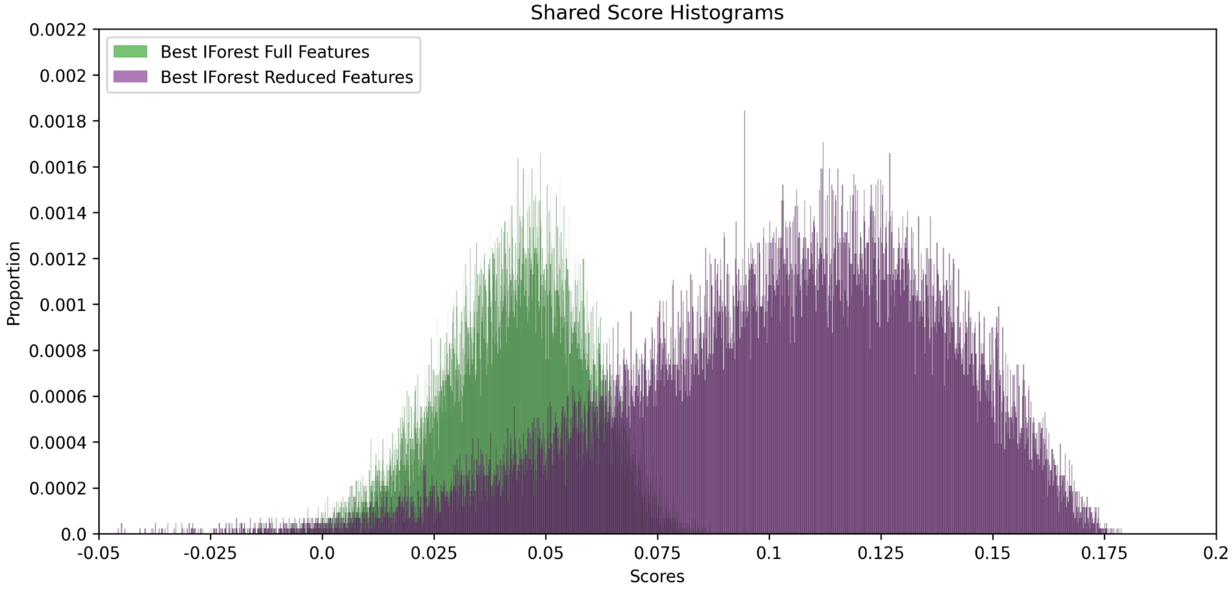
IForest models directly predict anomalous points by the average length of trees required to isolate each point. The number of anomalous points is directly controlled by a contamination rate parameter. Under normal circumstances, domain knowledge or labeled anomalies would be used to determine the contamination rate. For this project, the ratio of suspected anomalies in the training dataset was used as the base contamination rate. The IForests were trained using 1/16, 1/8, 1/4, 1/2, and 1 times the base contamination rate, as a conservative estimate of the number of anomalies in the dataset.

Three clustering metrics were calculated for each model: the Silhouette coefficient, the Calinski-Harabasz index, and the Davies-Bouldin index. Each of these clustering metrics attempts to capture "goodness of fit" for the clusters by measuring the relative density of intra-cluster points and the relative spread between clusters. It should be noted that these metrics are designed for clustering data and not for anomaly detection. Anomalous datapoints are generally rare and poorly clustered by definition, which can lead to unclear results for each of these metrics. The Akaike Information Criterion (AIC) and Bayesian Information Criterion (BIC) were also used for GMM models, which measures the goodness of fit of the GMM model

to the data points. This can also be misleading since anomalous points are rare, meaning that the GMM model can fit very well to normal data while failing to capture anomalies.
These metrics were used for choosing candidate models to investigate more closely, but the main evaluation method was to plot the predicted anomalies using PCA projection and to visually evaluate the results. This method is ultimately subjective and is biased by the projection induced by PCA, which may not agree with other projection methods such as t-SNE or UMAP.

Results





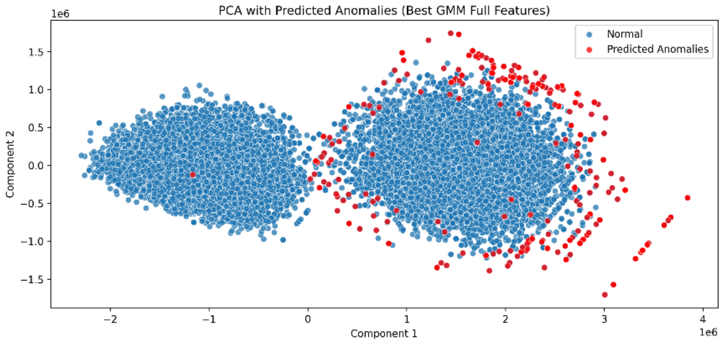
The best GMM models used the "tied" covariance type, meaning each component used a shared general covariance matrix. By using a shared general covariance matrix, the log-likelihood scores had a single left-skewed distribution, which made the identification of outliers easier. Notably, the GMM model on the full feature set had a greater amount of variation, which made it better at picking up anomalies compared to the reduced feature set. The full feature set GMM used only 4 components, which is roughly aligned with the visual interpretation under PCA projection of the number of clusters.

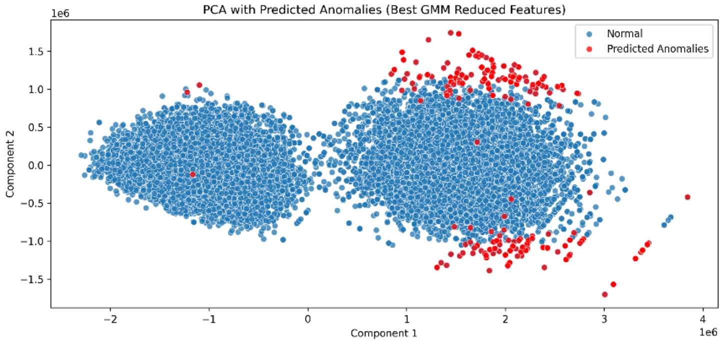
The best IForest models used the default hyperparameters (1x base contamination, max depth 256, 1.0 feature ratio). The scores for the IForest models were much more stochastic, although they roughly resembled normal distributions. The IForest models struggled to identify anomalies. The IForest models performed better on the reduced feature set, as the removal of low-variance categorical features made the IForest choose "bad" features less frequently and enabled the IForest to isolate anomalies in earlier splits.

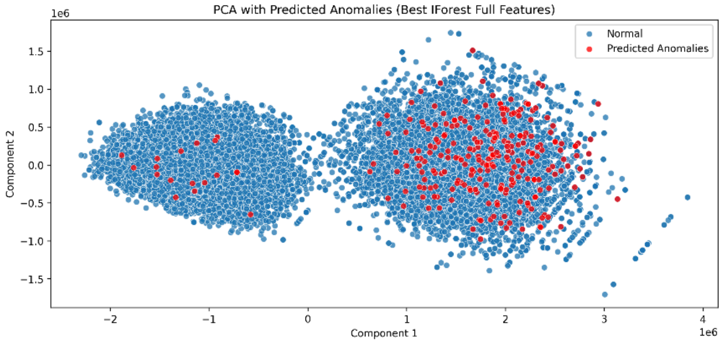
It is not clear why the GMM models performed better on the full feature set. It is possible that because GMM is trying to fit the data to a number of components rather than directly predicting anomalies that the GMM is able to leverage the lower information categorical variables better for component assignments, which are later converted into anomaly predictions. Since IForest tries to predict anomalies directly, by randomly making splits, categorical features with low variance between groups produces splits with low isolation power, causing the IForest to perform worse using the full feature set.

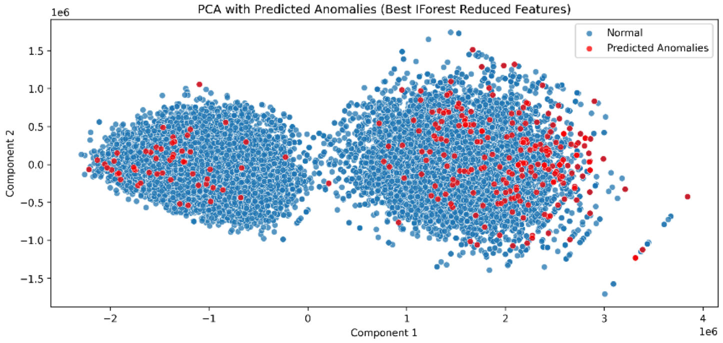
The predictions for each of the best models is shown on the next page. The GMM using the full feature set captured most of the "ring" outliers on the rightmost cluster, as well as the entire right island. This matches closely with visual intuition for outliers based on the PCA projection, and the GMM was even able to capture the suspected outliers that were not able to be cleanly identified via EDA. It is possible that, with a higher threshold of predicted outliers, this GMM model would capture practically all of the visual outliers. The GMM using the reduced feature set performed worse, mainly capturing anomalies on the right side of the right cluster. This makes sense given that the reduced feature set closely matches with the process used to find the suspected anomalies during EDA, so by losing information that was not able to be investigated during EDA, the GMM model was not able to capture new anomalies. Notably, both GMM models captured all five of the "inner" outliers that represented the high amount anomalies discovered during EDA.

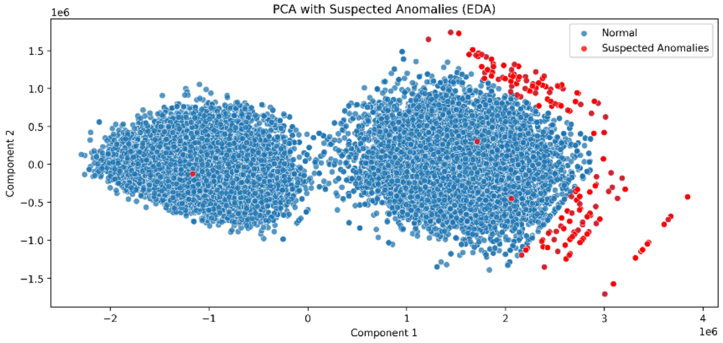
The IForest models greatly struggled to identify anomalies. Neither the full nor reduced feature set enabled the IForest models to capture the outer ring outliers on the right cluster. While the reduced feature set IForest performed better than the full feature set IForest, it still only captured some of the right-most "island". The reduced feature set IForest did capture all five of the "inner" outliers, representing high amount anomalies discovered during EDA, but the full feature set IForest only captured one of these anomalies. In general, by visual inspection, the anomalies detected by the IForest are of much lower quality.


PCA with Predicted Anomalies (Best GMM Full Features)


PCA with Predicted Anomalies (Best GMM Reduced Features)


PCA with Predicted Anomalies (Best IForest Full Features)


PCA with Predicted Anomalies (Best IForest Reduced Features)


PCA with Suspected Anomalies (EDA)

Limitations and Business Implications

Without labeled anomalies or domain expertise, it is impossible to truly evaluate the quality of each of these anomaly detection algorithms. The PCA projection provides some insights, but it is worth acknowledging that the human brain is adept at finding patterns even if those patterns do not actually exist. Under different projection schemes, these patterns may be different or entirely absent, and so visual inspection is not an ideal measurement metric. However, the suspected anomalies found during EDA do represent suspicious patterns: high transaction amounts or high volumes of transactions over short time periods. These patterns do appear to be reflected in the PCA projection which would imply that these engineered features are effective and that this is a reasonable method for evaluating the anomaly detection capabilities of each model.

Unfortunately, because of the nature of anomaly detection, typical unsupervised metrics such as the Silhouette score, the Calinkski-Harabasz index, and the Davies-Bouldin index can lead to improper conclusions. These clustering metrics evaluate the strength of clusters based on intra-cluster similarity and inter-cluster distance, but by definition, the intra-cluster similarity of anomalies is low, and for this dataset, the inter-cluster distances may not be especially high. In particular, all of the high amount transactions which are quite likely anomalous are buried with all of the normal data points, which can make it not only difficult for models to extract these anomalies but also difficult to rate the clusters, since the clusters are not well-separated.

For deployment into a business environment, it is important to collect some labeled anomaly data or consult a subject matter expert. The current methodology for highlighting suspected anomalies is rudimentary and may miss important fraudulent behavior. The anomaly rate for thresholding should be fine-tuned based on labeled anomalies and subject matter expert input. In general, it may be better to overestimate potential anomalous transactions. Flagging non-anomalous transactions as anomalous can be resolved by talking to the appropriate merchant/account holder/vendor, but failing to flag anomalous transactions leads to fraud and potential losses for the financial company. Over time, labels can be gathered for anomaly detection, and the model performance can be monitored using supervised classification metrics, such as false positive and false negative rates, to ensure that the model is not degrading as data distributions shift.