Abigail McDonald
December 27, 2021

Anomaly Detection – Fraudulent Transactions

Introduction

Company X has identified some fraudulent transactions and would like to be able to identify any past (available data) and future fraudulent transactions (through functional modeling). The dataset given has individual past transactions with a unique product code and salesperson ID in order to mask personal data. It also includes the quantity of items sold and the value of the transaction. Additionally, some of the transactions have already been identified as fraudulent or legitimate. However, most transactions are unknown in veracity. This split data leads to some potential supervised solutions, but unsupervised models will be looked at as well.

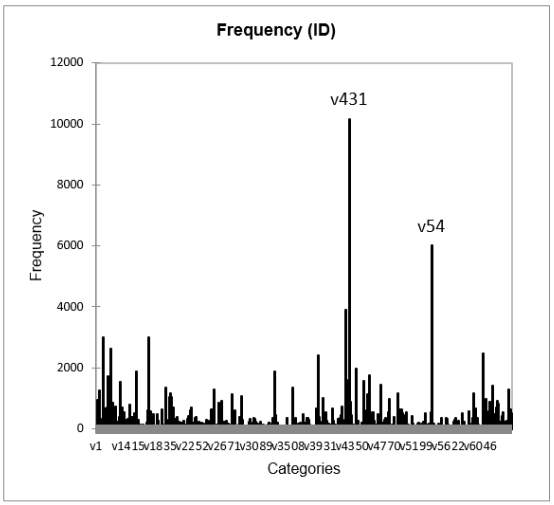
Statistical Analysis

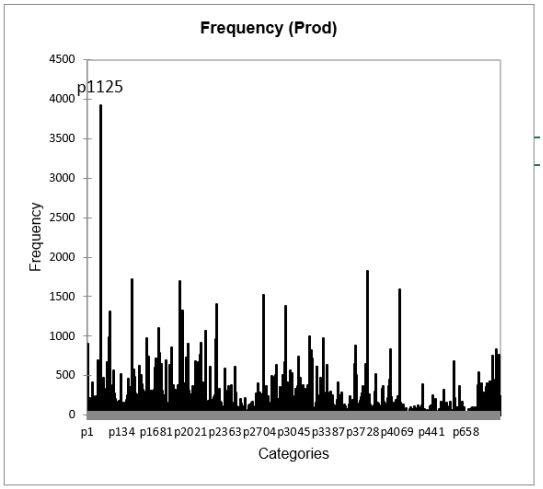
Upon inspection of the data, 'NA's were found in Quant and Val columns indicating known transactions without full data. NAs were then replaced with column means in order to allow for basic statistical analysis and modeling.

Descriptive statistics (Quantitative data):

Statistic	Minimum	Maximum	Range	Median	Mean	Standard deviation (n-1)
Cl_Quant	100.000	473883883.000	473883783.000	180.000	8441.996	902367.521
Cl_Val	1005.000	4642955.000	4641950.000	2690.000	14617.074	69609.807
Cl_Insp	0.000	2.000	2.000	2.000	1.925	0.376

According to basic descriptive statistics, there are large value and quantity fluctuations within columns with high standard deviations. Thus, the quantity and value data were standardized because frequency of occurrence by Prod and ID is important and should be equally weighted for fraud identification. Some interesting future descriptive statistics on identified fraudulent transactions could compare value and quantity against nonfraudulent transactions.





Salesperson ID v431 and v54 had the most frequent number of transactions and product p1125 had the largest single transaction sales of all products.

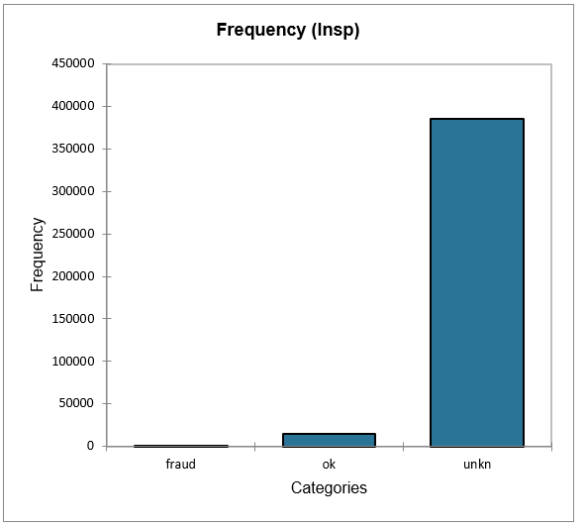
Percentages / Column (ID \ Insp):

ID \ Insp	fraud	ok	unkn	Total
Total	100.00%	100.00%	100.00%	100.00%
v54	1.65%	1.60%	1.50%	1.50%
v3034	1.57%	0.00%	0.01%	0.02%
v4998	1.57%	0.13%	0.03%	0.04%
v472	1.57%	0.24%	0.36%	0.36%
v216	1.18%	0.04%	0.01%	0.02%
v359	1.10%	0.03%	0.01%	0.01%
v3048	1.10%	0.03%	0.02%	0.02%
v2468	0.87%	0.01%	0.01%	0.01%
v5902	0.79%	0.01%	0.01%	0.01%
v5421	0.79%	0.10%	0.02%	0.03%
v1014	0.79%	0.14%	0.24%	0.24%
v215	0.71%	0.04%	0.02%	0.02%
v1144	0.71%	0.12%	0.07%	0.07%
v194	0.63%	0.02%	0.03%	0.03%
v213	0.63%	0.05%	0.02%	0.02%
v4997	0.55%	0.01%	0.00%	0.00%
v4948	0.55%	0.08%	0.01%	0.01%
v955	0.55%	0.64%	0.31%	0.32%
v739	0.55%	0.66%	0.34%	0.35%

Percentages / Column (Prod \ Insp):

Prod \ Insp	fraud	ok	unkn	Total
Total	100.00%	100.00%	100.00%	100.00%
p2456	2.28%	0.40%	0.14%	0.15%
p1000	0.94%	0.61%	0.21%	0.22%
p2459	0.71%	0.10%	0.04%	0.04%
p226	0.71%	0.12%	0.04%	0.05%
p314	0.71%	0.12%	0.04%	0.05%
p545	0.71%	0.14%	0.05%	0.05%
p3774	0.71%	1.25%	0.42%	0.45%
p2516	0.63%	0.07%	0.02%	0.03%
p1125	0.63%	2.70%	0.91%	0.98%
p318	0.55%	0.03%	0.01%	0.01%
p547	0.55%	0.05%	0.02%	0.02%
p3199	0.55%	0.44%	0.15%	0.16%
p2273	0.55%	0.96%	0.33%	0.35%
p319	0.47%	0.06%	0.02%	0.02%
p566	0.47%	0.06%	0.02%	0.03%
p966	0.47%	0.17%	0.06%	0.06%

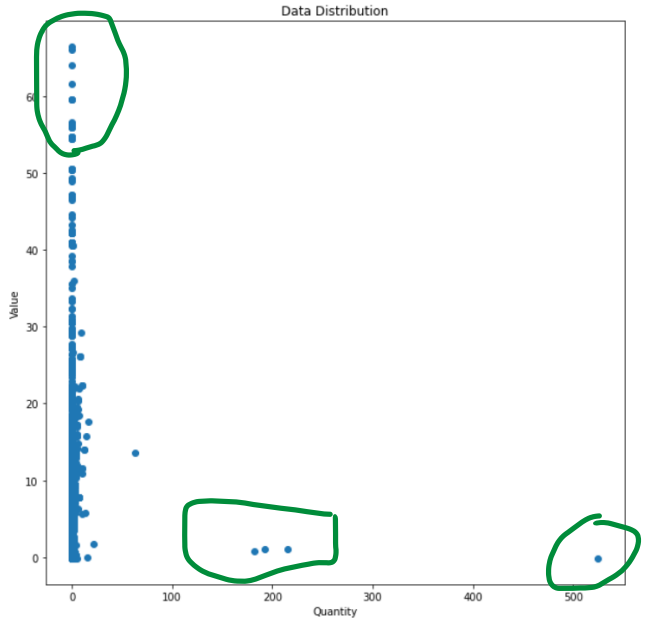
A contingency file (in excel workbook) includes a contingency table listing product and vendor id with fraud counts. In both figures, the largest cases of fraud (by percentage) are noted, the highlight indicates that the fraud percentage is larger than the ok percentage flagging potential problematic ids.



Descriptive Statistics (Python) for Known Data Only

```
Total number of Transactions are 15732
Number of Normal Transactions are 14462
Number of fraudulent Transactions are 1270
Percentage of fraud Transactions is 8.78
```

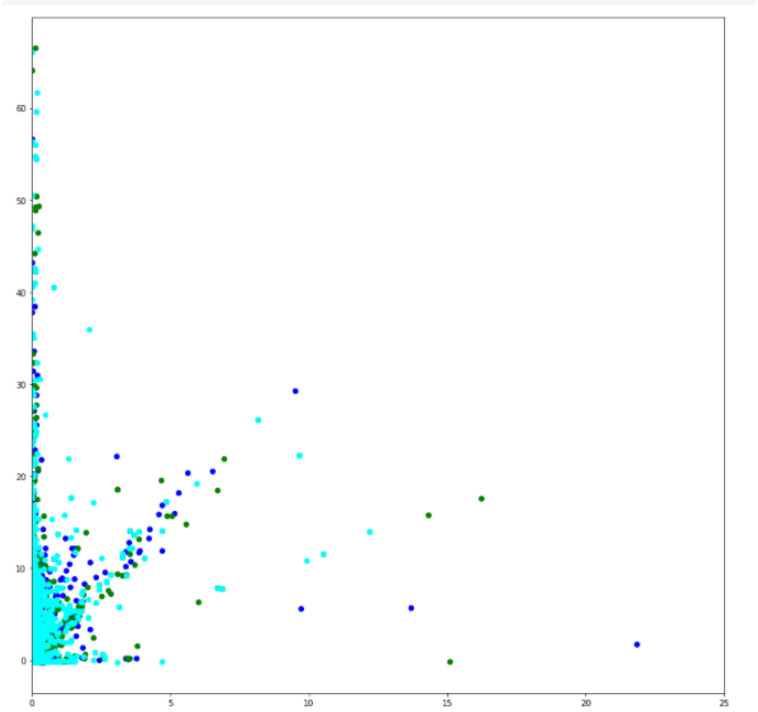
Known fraudulent transactions make up only 8.78% of the known data (exluding 'unkn' label). This will be a challenge when running supervised models.



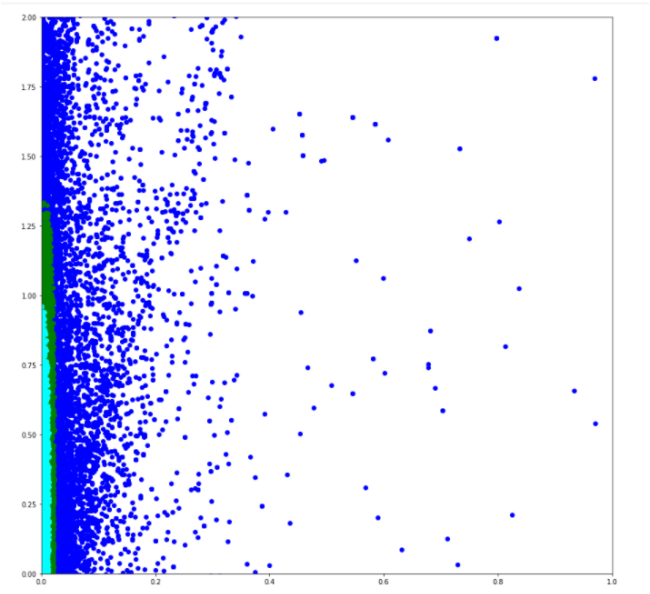
The standardized data distribution indicates some potential outliers with large quantity amounts but very low value paid. These could be transactions to look at, especially with price point data for item cost/listed price. The same could be said of the extreme values with low quantity.

Unsupervised Approaches

Although supervised models are usually considered in fraud cases, I ran a few unsupervised approaches to look for anomalous data.



The KNN model was run with 3 clusters as indicated by the supervised modeling. The clusters appear three dimensional. The KNN cluster values for the unsupervised model are in a column in the Excel spreadsheet. I do not have the cloud computing resources to run a Gower distance (correlation matrix) on the KNN cluster values versus the Inspection values, but this would be a good future endeavor with appropriate processing capabilities.



The results from running 3 classes of Gaussian Mixed Models indicates that there are some flagged clusters around the low quantity, high value region (cyan and green) that may be indicative of fraudulent activity.

Supervised Approaches

The data was split into a training data set consisting of known data ('fraud', 'ok' labels) and a testing set of unknown data ('unkn' label). As the fraud composes such a small percentage of the known data, oversampling by upsampling was necessary to raise precision in modeling. A variety of models were run including: Decision Tree, K Nearest Neighbors, Logistic Regression, Random Forest, and SVM. The accuracy and F1 scores are as follows:

Model	Accuracy %	F1 score
Decision Tree	92.24510551741673	0.37371663244353187
KNN (3 cluster)	93.10958555809814	0.43186582809224316
Logistic Regression	91.96542079837274	0.018633540372670808
Random Forest	92.06712433257056	0.04294478527607362
SVM	91.96542079837274	0.0125

Accuracy scores are high, most likely due to the imbalance of known data. However, F1 score (precision/recall) are very low for most models and low for Decision Tree and KNN. Due to the KNN model performing the best, this model was selected to run the testing data on. The predicted labels are available in the spreadsheet. KNN was run with multiple cluster amounts but was found to provide the highest F1 score when run with 3 clusters.

Results

Even with oversampling, the supervised models had low F1 scores. With more known data, a supervised model could be a good choice for fraud detection, but the known fraud is currently too low to provide high precision/recall scores. All things consider with current resources, the KNN model with 3 clusters performed the best and would be my recommended model going forward to predict future fraud.

Files

Jupyter Notebooks:

- Sales_unsuper *(associated CSV is salessimplified.csv)*
- Sales_supervised *(associated CSV is sales.csv)*

Sales Analysis (.xlsm file)