Capstone 1 Project : Statistical Data Analysis
Fraud Detection in Mobile Payment Data
Springboard Data Science Career Track

Eric Cruz

There may be variables or combinations and functions of variables that seem significant in explaining the question of whether a transaction is fraud. In particular, the "Type" and "Amount" come forward in the data visualizations for reasons discussed below. See link to the statistical tests on these variable as described below:

https://github.com/cruzer42/Capstone-Project-1-for-Springboard/blob/master/Capstone%20Project1%20Statistical%20Data%20Analysis.ipynb

There is a significant subgroup in the "Type" variable because only 2 of the 5 types are represented for the target cases (e.g. isFraud=1). See the crosstab frequency table, or contingency table of Type/isFraud:

isFraud	0	1
type		
CASH_IN	1399284	0
CASH_OUT	2233384	4116
DEBIT	41432	0
PAYMENT	2151495	0
TRANSFER	528812	4097

We can perform the Pearson's Chi-squared test of the null hypothesis that the frequency of categorical variable "Type" is independent across the "isFraud" categories. As expected, the p-value is virtually zero, so the null hypothesis is rejected. This indicates dependence of the categorical variable on the category.

stats.chi2_contingency()

```
(22082.53571319108, 0.0, 4, array([[1.39747778e+06, 1.80622440e+03],
    [2.23461179e+06, 2.88821075e+03],
    [4.13785187e+04, 5.34812728e+01],
    [2.14871781e+06, 2.77719374e+03],
    [5.32221110e+05, 6.87889834e+02]]))
```

The more important subgroup is the target variable itself, the "isFraud" boolean indicator. The dataset is highly imbalanced with only 0.13%, or about 8 thousand of approximately 6 million transactions. As shown in the Data Storytelling analysis, the visualization of “Amount” shows the fraud transactions in a smaller, tighther range than the nonfraud transactions. We can test the null hypothesis that the mean “Amount” is the same for both categories, using a t-test. See the mean and std calculated by category, along with the corresponding p-value calculated from the t-test.

```
The number of nonfraud amounts is 6354407

The number of fraud amounts is 8213

The std amount nonfraud is 596236.9813471739

The std amount with fraud is 2404252.9472401612

The mean amount nonfraud is 178197.04172739814

The mean amount with fraud is 1467967.299140387

The calculated stdev_diff is 602079.9804398556

The mean fraud minus nonfraud amount is 1289770.257412989

The calculated t-stat is 194.01200466038233
```

```
ttest_ind(fraud_amount, nonfraud_amount)
Ttest_indResult(statistic=194.01200466037974, pvalue=0.0)
```

The resulting p-value of virtually zero indicates that the null hypothesis that the Amount is the same for both categories is rejected, or the Amount is statistically different.

Also shown in the Data Storytelling analysis, there is a strong correlation between “Amount” and “Old Balance Orig” which can be seen in the fraud transactions but not the non-fraud transactions, and higher but not quite as significant between “Amount” and “New Balance Orig”. This is plausible given the provided description “In this specific dataset the fraudulent behavior of the agents aims to profit by taking control or customers accounts and try to empty the funds by transferring to another account and then cashing out of the system.”

The Paysim dataset is a synthetic simulation of mobile payment data, because actual data is generally not available due to proprietary concerns. This means the creation of underlying data can be considered a bootstrap approach. The test to determine fraud is inferential, but not exactly along the lines of traditional hypothesis testing due to the imbalance observed in the target boolean variable. Rather than a linear regression analysis, a more appropriate approach for binary classification is logistic regression. It also seems plausible to consider a Bayesian approach, such as the GaussianNB model. However, there are several other models that may be appropriate given the binary classification problem, which will be considered as we learn about models in upcoming sections of the course. Some potential candidates are Random Forest Classifier, Support Vector Machine, and KNN just to name a few.