1. Executive Summary



This report provides an analysis and evaluation of Card Transaction Data for detecting fraud using supervised machine learning methods. The original data set contains 96,708 records of card transactions with 10 variables of transaction details such as card number, date of transaction, merchant ID and description, transaction type and amount of transaction.



**Process Overview**

The general process of analysis follows data cleaning and manipulation, building expert variable, selecting important variables, applying fraud algorithms, calculating fraud scores and evaluating results. We divided the dataset into training, testing and out of time and continued to evaluate the fraud score for each of these brackets.

The tools used include R and Excel, and some of the algorithms used for analysis are Bootstrap Forest (aka Random Forest), Boosted Trees, Neural Networks, Naïve Bayes and Logistic Regression. These five models were built and tested using R and their respective performances were recorded. Among all the models built, Random Forest with 150 trees exhibited the best results. The estimated ROI using this model was $231,710.

Using this best model, Fraud Detection Rate for training set was 71.5%, for testing set it was 68.4% and for out of time set it was 68.3%. Through our analysis we could further identify that there were several expert variables that were strong predictors of fraud. Detailed examination of the important features selected from different algorithms indicates that fraudulent records typically have unusual geographical appearance.

1. Data Overview

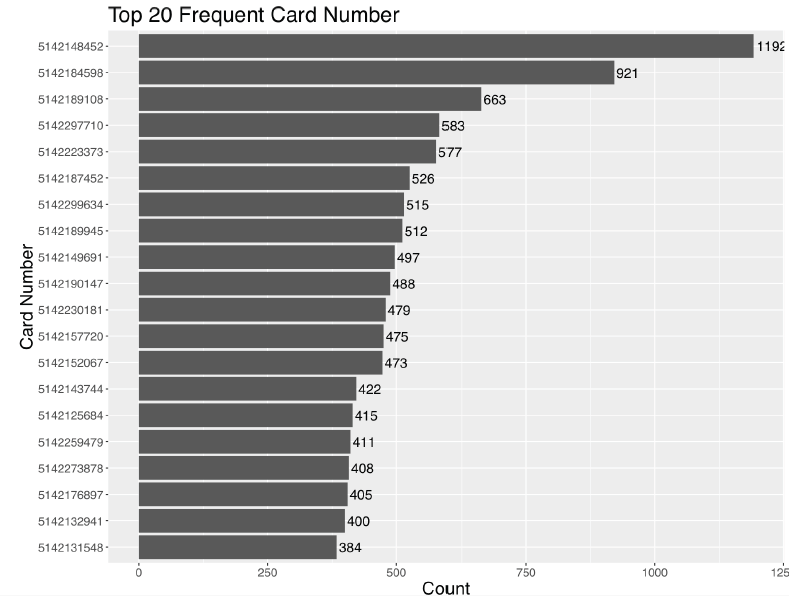
Card payment data is a dataset containing 96,708 records of card transaction from 2010-01-01 to 2010-12-31. It includes information about card number, date, merchant’s number, description, state, zip code, transaction type, and amount. Every record is also labeled as fraud or not. In total, there are 298 labeled fraudulent records.

**10 variables in total** – 1 numeric, 7 categorical, 1 text, 1 date  
**Numeric**: amount  
**Categorical**: recordnum, cardnum, merchnum, merch.state, merch.zip, transtype, and fraud  
**Text**: merch.description  
**Date**: date

Following is the description of the variables we consider to be the most important. The complete Data Quality Report can be found in appendix.

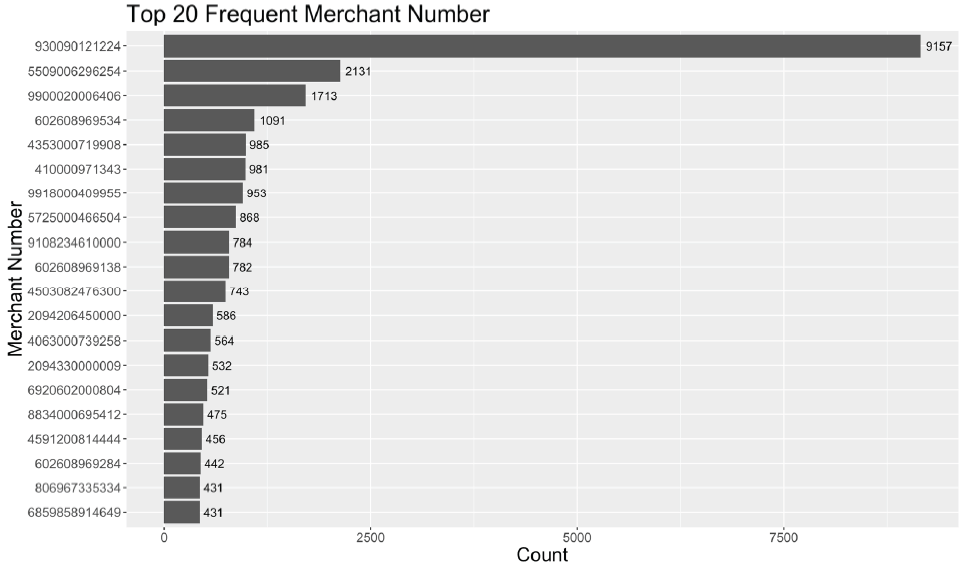
# **2.1 Description of important variables**

**2.2.1 cardnum**  
cardnum is a categorical variable. It is the number of the card used for the payment.  
Distribution   
100% populated and 1644 unique values. The following plot shows top 20 frequent card numbers



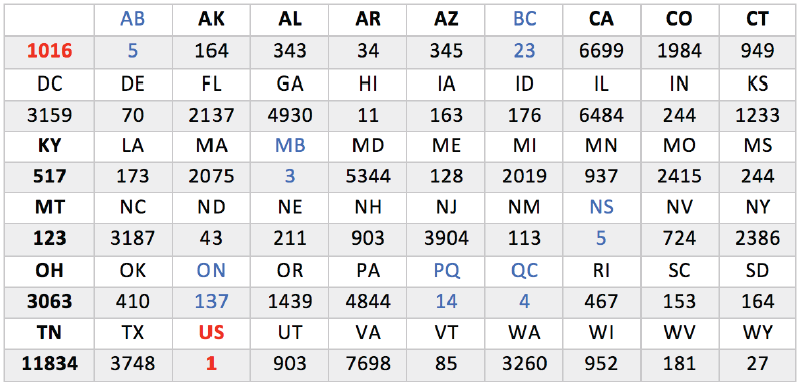
**2.2.2 merchnum**merchnum is a categorical variable. It is the merchant number involved in the payment.

Distribution  
96.5% populated with 13,090 unique values. There are 3,375 missing values, which includes the number of 0’s. The following plot shows top 20 frequent merchant numbers.

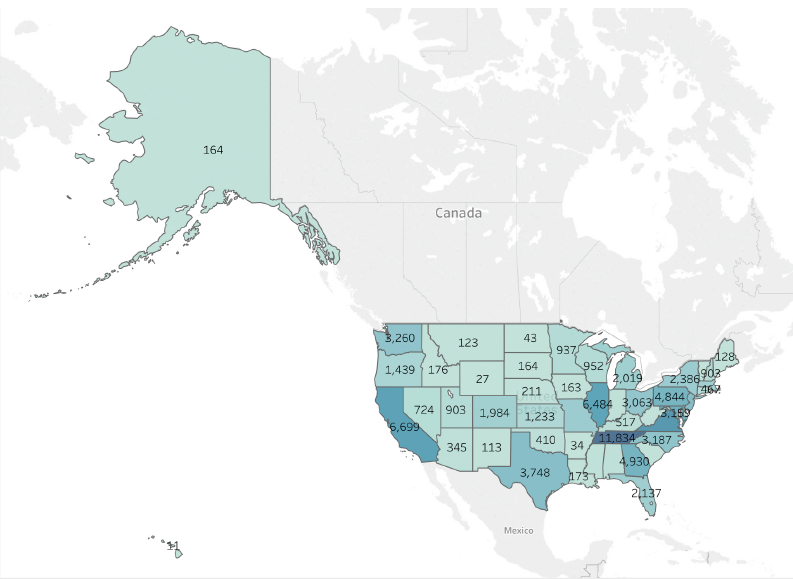


**2.2.3 merch.state**merch.state is a categorical variable, indicating the abbreviations of US states of the each of the merchant.

Distribution  
98.7% populated with 60 unique values. 51 values stand for regions in the United States, 50  
states and District of Columbia. 7 values stand for Canada. Apart from these, there are 1,199 missing values, and 1 invalid value US. Distribution of states is showed in the following table. States in Canada are marked in blue. Invalid and missing values are marked in red. Regions in the United States are in black.



The following picture displays distribution by states in the United States. We can clearly see that TN has the largest number of records (11,990).



**Details for merch.state with colors showing Number of Records**

**2.2.4 merch.zip**

merch.zip is a categorical variable, indicating the zip code of the merchant.

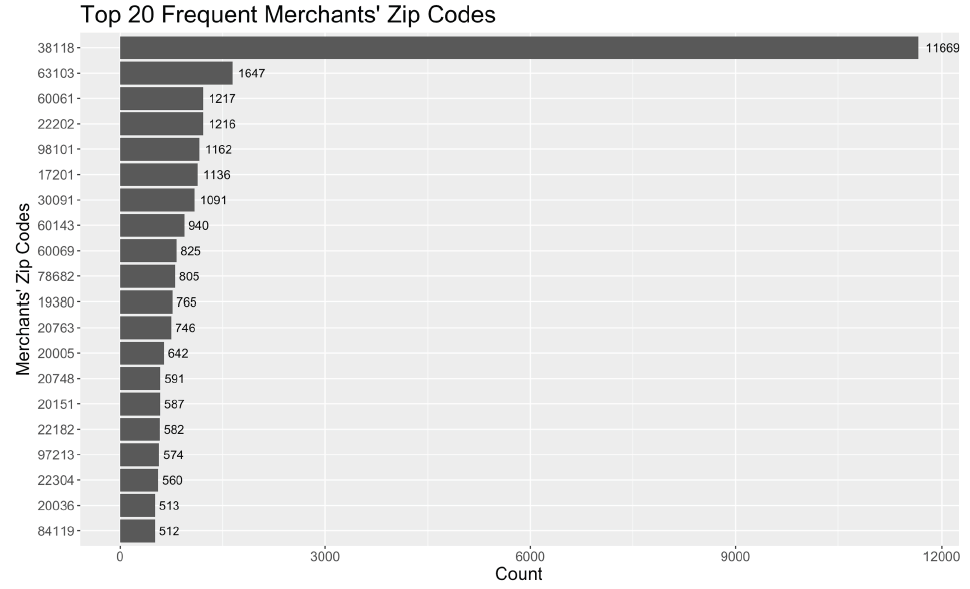
Distribution

95.2% populated with 4567 unique values. 86.99% values have invalid 5-digit zip code. The

following plot shows top 20 zip codes.

Although there is one zip code “38118” appears 10 times more than other values, it is in TN,

which is the most frequent states; hence we do not treat it as a frivolous value.



1. Handling Missing Values

According to data quality analysis, three variables -- merchnum, merch.state and merch.zip are not 100% populated, with missing values and 0’s. We assigned values to these fields based on merch.description. We considered records with different merch.description as different merchants, and assumed that each merchant should have a unique merchant number, be in one state, and only have one zip code. Since all these three variables are categorical, we assigned unique values in these fields according to merch.description, and those values were designed to be quite different from other existing values, therefore easy to recognize. Missing values and 0’s in merchnum were replaced by strings from ‘M1’ to ‘M771’, missing values in merch.state were substituted by numbers from ‘1’ to ‘150’, and NA’s in merch.zip were replaced by strings from ‘M1’ to ‘M644’.

1. Variable Creation

We created 130 expert variables and we kept 40 variables after KS.

Since this analysis involves time, with limited data, we chose four different time windows, 3, 7, 14 and 28 days.

The rationale is to capture more (and different types of) fraudulent records that might be detected in those time windows.

Furthermore, we kept 25 variables after lasso, we use these 25 to train our models.

|  |  |
| --- | --- |
| **Variable Name** | **Description/Formula** |
| avg\_amount\_cardnum\_30 | Average of consumption amount with same card number within 30 days |
| avg\_amount\_cardnum\_15 | Average of consumption amount with same card number within 15 days |
| avg\_amount\_cardnum\_7 | Average of consumption amount with same card number within 7 days |
| avg\_amount\_cardnum\_3 | Average of consumption amount with same card number within 3 days |
| avg\_amount\_cardnum\_1 | Average of consumption amount with same card number within 1 days |
| max\_amount\_cardnum\_30 | Maximum of consumption amount with same card number within 30 days |
| max\_amount\_cardnum\_15 | Maximum of consumption amount with same card number within 15 days |
| max\_amount\_cardnum\_7 | Maximum of consumption amount with same card number within 7 days |
| max\_amount\_cardnum\_3 | Maximum of consumption amount with same card number within 3 days |
| max\_amount\_cardnum\_1 | Maximum of consumption amount with same card number within 1 days |
| sum\_amount\_cardnum\_30 | Sum of consumption amount with same card number within 30 days |
| sum\_amount\_cardnum\_15 | Sum of consumption amount with same card number within 15 days |
| sum\_amount\_cardnum\_7 | Sum of consumption amount with same card number within 7 days |
| sum\_amount\_cardnum\_3 | Sum of consumption amount with same card number within 3 days |
| sum\_amount\_cardnum\_1 | Sum of consumption amount with same card number within 1 days |
| avg\_weekdiff\_amount\_cardnum\_30 | Average of the difference between personal card consumption and average of consumption amount for different day of week with same card number within 30 days |
| avg\_weekdiff\_amount\_cardnum\_15 | Average of the difference between personal card consumption and average of consumption amount for different day of week with same card number within 15 days |
| avg\_weekdiff\_amount\_cardnum\_7 | Average of the difference between personal card consumption and average of consumption amount for different day of week with same card number within 7 days |
| avg\_weekdiff\_amount\_cardnum\_3 | Average of the difference between personal card consumption and average of consumption amount for different day of week with same card number within 3 days |
| avg\_weekdiff\_amount\_cardnum\_1 | Average of the difference between personal card consumption and average of consumption amount for different day of week with same card number within 1 days |
| max\_weekdiff\_amount\_cardnum\_30 | Maximum of the difference between personal card consumption and average of consumption amount for different day of week with same card number within 30 days |
| max\_weekdiff\_amount\_cardnum\_15 | Maximum of the difference between personal card consumption and average of consumption amount for different day of week with same card number within 15 days |
| max\_weekdiff\_amount\_cardnum\_7 | Maximum of the difference between personal card consumption and average of consumption amount for different day of week with same card number within 7 days |
| max\_weekdiff\_amount\_cardnum\_3 | Maximum of the difference between personal card consumption and average of consumption amount for different day of week with same card number within 3 days |
| max\_weekdiff\_amount\_cardnum\_1 | Maximum of the difference between personal card consumption and average of consumption amount for different day of week with same card number within 1 days |
| sum\_weekdiff\_amount\_cardnum\_30 | Sum of the difference between personal card consumption and average of consumption amount for different day of week with same card number within 30 days |
| sum\_weekdiff\_amount\_cardnum\_15 | Sum of the difference between personal card consumption and average of consumption amount for different day of week with same card number within 15 days |
| sum\_weekdiff\_amount\_cardnum\_7 | Sum of the difference between personal card consumption and average of consumption amount for different day of week with same card number within 7 days |
| sum\_weekdiff\_amount\_cardnum\_3 | Sum of the difference between personal card consumption and average of consumption amount for different day of week with same card number within 3 days |
| sum\_weekdiff\_amount\_cardnum\_1 | Sum of the difference between personal card consumption and average of consumption amount for different day of week with same card number within 1 days |
| avg\_monthdiff\_amount\_cardnum\_30 | Average of the difference between personal card consumption and average of consumption amount for different month with same card number within 30 days |
| avg\_monthdiff\_amount\_cardnum\_15 | Average of the difference between personal card consumption and average of consumption amount for different month with same card number within 15 days |
| avg\_monthdiff\_amount\_cardnum\_7 | Average of the difference between personal card consumption and average of consumption amount for different month with same card number within 7 days |
| avg\_monthdiff\_amount\_cardnum\_3 | Average of the difference between personal card consumption and average of consumption amount for different month with same card number within 3 days |
| avg\_monthdiff\_amount\_cardnum\_1 | Average of the difference between personal card consumption and average of consumption amount for different month with same card number within 1 days |
| max\_monthdiff\_amount\_cardnum\_30 | Maximum of the difference between personal card consumption and average of consumption amount for different month with same card number within 30 days |
| max\_monthdiff\_amount\_cardnum\_15 | Maximum of the difference between personal card consumption and average of consumption amount for different month with same card number within 15 days |
| max\_monthdiff\_amount\_cardnum\_7 | Maximum of the difference between personal card consumption and average of consumption amount for different month with same card number within 7 days |
| max\_monthdiff\_amount\_cardnum\_3 | Maximum of the difference between personal card consumption and average of consumption amount for different month with same card number within 3 days |
| max\_monthdiff\_amount\_cardnum\_1 | Maximum of the difference between personal card consumption and average of consumption amount for different month with same card number within 1 days |
| sum\_monthdiff\_amount\_cardnum\_30 | Sum of the difference between personal card consumption and average of consumption amount for different month with same card number within 30 days |
| sum\_monthdiff\_amount\_cardnum\_15 | Sum of the difference between personal card consumption and average of consumption amount for different month with same card number within 15 days |
| sum\_monthdiff\_amount\_cardnum\_7 | Sum of the difference between personal card consumption and average of consumption amount for different month with same card number within 7 days |
| sum\_monthdiff\_amount\_cardnum\_3 | Sum of the difference between personal card consumption and average of consumption amount for different month with same card number within 3 days |
| sum\_monthdiff\_amount\_cardnum\_1 | Sum of the difference between personal card consumption and average of consumption amount for different month with same card number within 1 days |
| cardnum\_30 | Count of records with same card number within 30 days before original record |
| cardnum\_15 | Count of records with same card number within 15 days before original record |
| cardnum\_7 | Count of records with same card number within 7 days before original record |
| cardnum\_3 | Count of records with same card number within 3 days before original record |
| cardnum\_1 | Count of records with same card number within 1 days before original record |
| merchantnum\_cardnum\_30 | Count of different merchant number with same card number within 30 days |
| merchantnum\_cardnum\_15 | Count of different merchant number with same card number within 15 days |
| merchantnum\_cardnum\_7 | Count of different merchant number with same card number within 7 days |
| merchantnum\_cardnum\_3 | Count of different merchant number with same card number within 3 days |
| merchantnum\_cardnum\_1 | Count of different merchant number with same card number within 1 days |
| zip\_cardnum\_30 | Count of different zip with same card number within 30 days |
| zip\_cardnum\_15 | Count of different zip with same card number within 15 days |
| zip\_cardnum\_7 | Count of different zip with same card number within 7 days |
| zip\_cardnum\_3 | Count of different zip with same card number within 3 days |
| zip\_cardnum\_1 | Count of different zip with same card number within 1 days |
| state\_cardnum\_30 | Count of different state with same card number within 30 days |
| state\_cardnum\_15 | Count of different state with same card number within 15 days |
| state\_cardnum\_7 | Count of different state with same card number within 7 days |
| state\_cardnum\_3 | Count of different state with same card number within 3 days |
| state\_cardnum\_1 | Count of different state with same card number within 1 days |
| avg\_amount\_merchantnum\_30 | Average of consumption amount with same merchant number within 30 days |
| avg\_amount\_merchantnum\_15 | Average of consumption amount with same merchant number within 15 days |
| avg\_amount\_merchantnum\_7 | Average of consumption amount with same merchant number within 7 days |
| avg\_amount\_merchantnum\_3 | Average of consumption amount with same merchant number within 3 days |
| avg\_amount\_merchantnum\_1 | Average of consumption amount with same merchant number within 1 days |
| max\_amount\_merchantnum\_30 | Maximum of consumption amount with same merchant number within 30 days |
| max\_amount\_merchantnum\_15 | Maximum of consumption amount with same merchant number within 15 days |
| max\_amount\_merchantnum\_7 | Maximum of consumption amount with same merchant number within 7 days |
| max\_amount\_merchantnum\_3 | Maximum of consumption amount with same merchant number within 3 days |
| max\_amount\_merchantnum\_1 | Maximum of consumption amount with same merchant number within 1 days |
| sum\_amount\_merchantnum\_30 | Sum of consumption amount with same merchant number within 30 days |
| sum\_amount\_merchantnum\_15 | Sum of consumption amount with same merchant number within 15 days |
| sum\_amount\_merchantnum\_7 | Sum of consumption amount with same merchant number within 7 days |
| sum\_amount\_merchantnum\_3 | Sum of consumption amount with same merchant number within 3 days |
| sum\_amount\_merchantnum\_1 | Sum of consumption amount with same merchant number within 1 days |
| avg\_weekdiff\_amount\_merchantnum\_30 | Average of the difference between personal card consumption and average of consumption amount for different day of week with same merchant number within 30 days |
| avg\_weekdiff\_amount\_merchantnum\_15 | Average of the difference between personal card consumption and average of consumption amount for different day of week with same merchant number within 15 days |
| avg\_weekdiff\_amount\_merchantnum\_7 | Average of the difference between personal card consumption and average of consumption amount for different day of week with same merchant number within 7 days |
| avg\_weekdiff\_amount\_merchantnum\_3 | Average of the difference between personal card consumption and average of consumption amount for different day of week with same merchant number within 3 days |
| avg\_weekdiff\_amount\_merchantnum\_1 | Average of the difference between personal card consumption and average of consumption amount for different day of week with same merchant number within 1 days |
| max\_weekdiff\_amount\_merchantnum\_30 | Maximum of the difference between personal card consumption and average of consumption amount for different day of week with same merchant number within 30 days |
| max\_weekdiff\_amount\_merchantnum\_15 | Maximum of the difference between personal card consumption and average of consumption amount for different day of week with same merchant number within 15 days |
| max\_weekdiff\_amount\_merchantnum\_7 | Maximum of the difference between personal card consumption and average of consumption amount for different day of week with same merchant number within 7 days |
| max\_weekdiff\_amount\_merchantnum\_3 | Maximum of the difference between personal card consumption and average of consumption amount for different day of week with same merchant number within 3 days |
| max\_weekdiff\_amount\_merchantnum\_1 | Maximum of the difference between personal card consumption and average of consumption amount for different day of week with same merchant number within 1 days |
| sum\_weekdiff\_amount\_merchantnum\_30 | Sum of the difference between personal card consumption and average of consumption amount for different day of week with same merchant number within 30 days |
| sum\_weekdiff\_amount\_merchantnum\_15 | Sum of the difference between personal card consumption and average of consumption amount for different day of week with same merchant number within 15 days |
| sum\_weekdiff\_amount\_merchantnum\_7 | Sum of the difference between personal card consumption and average of consumption amount for different day of week with same merchant number within 7 days |
| sum\_weekdiff\_amount\_merchantnum\_3 | Sum of the difference between personal card consumption and average of consumption amount for different day of week with same merchant number within 3 days |
| sum\_weekdiff\_amount\_merchantnum\_1 | Sum of the difference between personal card consumption and average of consumption amount for different day of week with same merchant number within 1 days |
| avg\_monthdiff\_amount\_merchantnum\_30 | Average of the difference between personal card consumption and average of consumption amount for different month with same merchant number within 30 days |
| avg\_monthdiff\_amount\_merchantnum\_15 | Average of the difference between personal card consumption and average of consumption amount for different month with same merchant number within 15 days |
| avg\_monthdiff\_amount\_merchantnum\_7 | Average of the difference between personal card consumption and average of consumption amount for different month with same merchant number within 7 days |
| avg\_monthdiff\_amount\_merchantnum\_3 | Average of the difference between personal card consumption and average of consumption amount for different month with same merchant number within 3 days |
| avg\_monthdiff\_amount\_merchantnum\_1 | Average of the difference between personal card consumption and average of consumption amount for different month with same merchant number within 1 days |
| max\_monthdiff\_amount\_merchantnum\_30 | Maximum of the difference between personal card consumption and average of consumption amount for different month with same merchant number within 30 days |
| max\_monthdiff\_amount\_merchantnum\_15 | Maximum of the difference between personal card consumption and average of consumption amount for different month with same merchant number within 15 days |
| max\_monthdiff\_amount\_merchantnum\_7 | Maximum of the difference between personal card consumption and average of consumption amount for different month with same merchant number within 7 days |
| max\_monthdiff\_amount\_merchantnum\_3 | Maximum of the difference between personal card consumption and average of consumption amount for different month with same merchant number within 3 days |
| max\_monthdiff\_amount\_merchantnum\_1 | Maximum of the difference between personal card consumption and average of consumption amount for different month with same merchant number within 1 days |
| sum\_monthdiff\_amount\_merchantnum\_30 | Sum of the difference between personal card consumption and average of consumption amount for different month with same merchant number within 30 days |
| sum\_monthdiff\_amount\_merchantnum\_15 | Sum of the difference between personal card consumption and average of consumption amount for different month with same merchant number within 15 days |
| sum\_monthdiff\_amount\_merchantnum\_7 | Sum of the difference between personal card consumption and average of consumption amount for different month with same merchant number within 7 days |
| sum\_monthdiff\_amount\_merchantnum\_3 | Sum of the difference between personal card consumption and average of consumption amount for different month with same merchant number within 3 days |
| sum\_monthdiff\_amount\_merchantnum\_1 | Sum of the difference between personal card consumption and average of consumption amount for different month with same merchant number within 1 days |
| merchantnum\_30 | Count of records with same merchant number within 30 days before original record |
| merchantnum\_15 | Count of records with same merchant number within 15 days before original record |
| merchantnum\_7 | Count of records with same merchant number within 7 days before original record |
| merchantnum\_3 | Count of records with same merchant number within 3 days before original record |
| merchantnum\_1 | Count of records with same merchant number within 1 days before original record |
| cardnum\_merchantnum\_30 | Count of different card number with same merchant number within 30 days |
| cardnum\_merchantnum\_15 | Count of different card number with same merchant number within 15 days |
| cardnum\_merchantnum\_7 | Count of different card number with same merchant number within 7 days |
| cardnum\_merchantnum\_3 | Count of different card number with same merchant number within 3 days |
| cardnum\_merchantnum\_1 | Count of different card number with same merchant number within 1 days |
| zip\_merchantnum\_30 | Count of different zip with same merchant number within 30 days |
| zip\_merchantnum\_15 | Count of different zip with same merchant number within 15 days |
| zip\_merchantnum\_7 | Count of different zip with same merchant number within 7 days |
| zip\_merchantnum\_3 | Count of different zip with same merchant number within 3 days |
| zip\_merchantnum\_1 | Count of different zip with same merchant number within 1 days |
| state\_merchantnum\_30 | Count of different state with same merchant number within 30 days |
| state\_merchantnum\_15 | Count of different state with same merchant number within 15 days |
| state\_merchantnum\_7 | Count of different state with same merchant number within 7 days |
| state\_merchantnum\_3 | Count of different state with same merchant number within 3 days |
| state\_merchantnum\_1 | Count of different state with same merchant number within 1 days |

We then built four types of variables using R:

1. **Type I** variables are intended to capture unusual amounts of transaction, both at the card

level and the merchant level.

Example: card\_amount\_to\_avg\_3 tells if a particular transaction amount is unusual

compared to historical averages of the past 3 days.

1. **Type II** variables are intended to capture unusual transaction frequency during a set period of time, both at the card level and the merchant level.

Example: card\_frequency\_3 describes the transaction frequency for each specific card number in the past 3 days

1. **Type III variables** are location related variables, which are intended to capture merchants with different zip codes and states in a set period of time.

Example: zip\_with\_merchnum\_3 tells how many zip codes are related with a particular merchant in the past 3 days.

4**) Type IV variables** are intended to catch card appearance pattern, either for a merchant or

for a card holder.

Example: merchnum\_per\_card\_3 tells how many different merchants a card was used for transaction within the past 3 days.

# **4.1 Type I Variables:**

**card\_amount\_to\_avg:** ratio of transaction amount of a specific card number to its historical average amount by time window 3, 7, 14 and 28 days. To set a baseline, the first value of each specific card number in a specific time window was replaced by a neutral value – 1 before building this set of variables.

**merchant\_amount\_to\_avg:** ratio of transaction amount of a specific merchant number to its historical average amount by time window 3, 7, 14 and 28 days. To set a baseline, the first value of each specific merchant number in a specific time window was replaced by a neutral value - 1 before building this set of variables.

**card\_amount\_to\_max:** ratio of transaction amount of a specific card number to its historical maximum amount by time window 3, 7, 14 and 28 days. To set a baseline, the first value of each specific card number in a specific time window was replaced by a neutral value – 0 before building this set of variables.

**merchant\_amount\_to\_max:** ratio of transaction amount of a specific merchant number to its historical maximum amount by time window 3, 7, 14 and 28 days. To set a baseline, the first value of each specific merchant number in a specific time window was replaced by a neutral value - 0 before building this set of variables.

**card\_amount\_to\_median:** ratio of transaction amount of a specific card number to historical median amount by time window 3, 7, 14 and 28 days. To set a baseline, the first value of each specific card number in a specific time window was replaced by a neutral value - 1 before building this set of variables.

**merchant\_amount\_to\_median:** ratio of transaction amount of a specific merchant number to its historical median amount by time window 3, 7, 14 and 28 days. To set a baseline, the first value of each specific merchant number in a specific time window was replaced by a neutral value - 1 before building this set of variables.

**card\_amount\_to\_total:** ratio of transaction amount of a specific card number to its historical total amount by time window 3, 7, 14 and 28 days. To set a baseline, the first value of each specific card number in a specific time window was replaced by a neutral value - 0 before building this set of variables.

**merchant\_amount\_to\_total:** ratio of transaction amount of a specific merchant number and historical total amount by time window 3, 7, 14 and 28 days. To set a baseline, the first value of each specific merchant number in a specific time window was replaced by a neutral value - 0 before building this set of variables.

**4.2 Type II Variables:**

**card\_frequency:**  transaction frequency for each specific card number by time window 3, 7, 14 and 28 days.

**merchant\_frequency:** transaction frequency for each specific merchant number by time window 3, 7, 14 and 28 days.

# **4.3 Type III Variables: Location**

**zip\_with\_merchnum:** number of different zip codes related to a particular merchant in the time windows of 3, 7, 14 and 28 days.

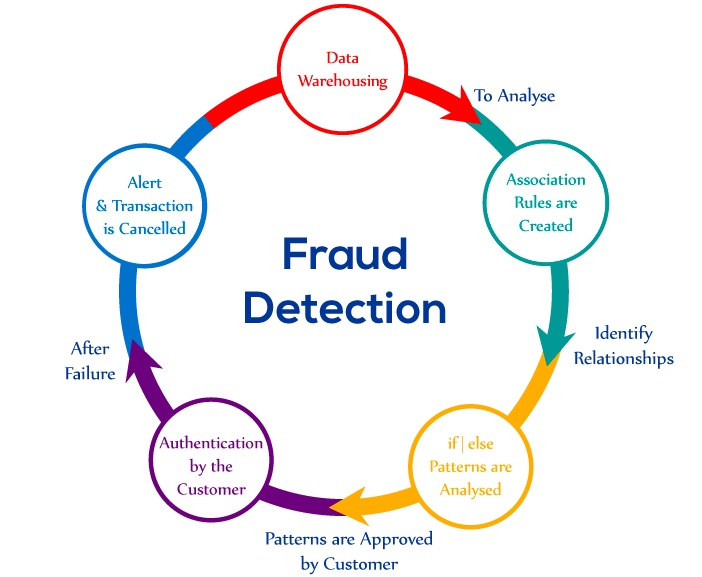
**state\_with\_merchnum:** number of different states related to a particular merchant in the time windows of 3, 7, 14 and 28 days.

# **4.4 Type IV Variables: Purchasing Pattern**

**cardnum\_per\_merch:** number of different cards associated with a particular merchant in time windows of 3, 7, 14 and 28 days.

**merchnum\_per\_card:** number of different merchants associated with a particular card in time windows of 3, 7, 14 and 28 days.

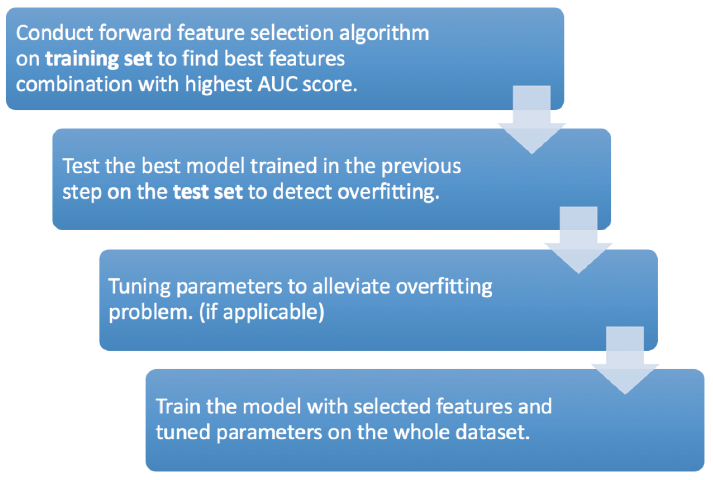
1. Supervised Fraud Algorithm



# **5.1 Training/Test Set Split**

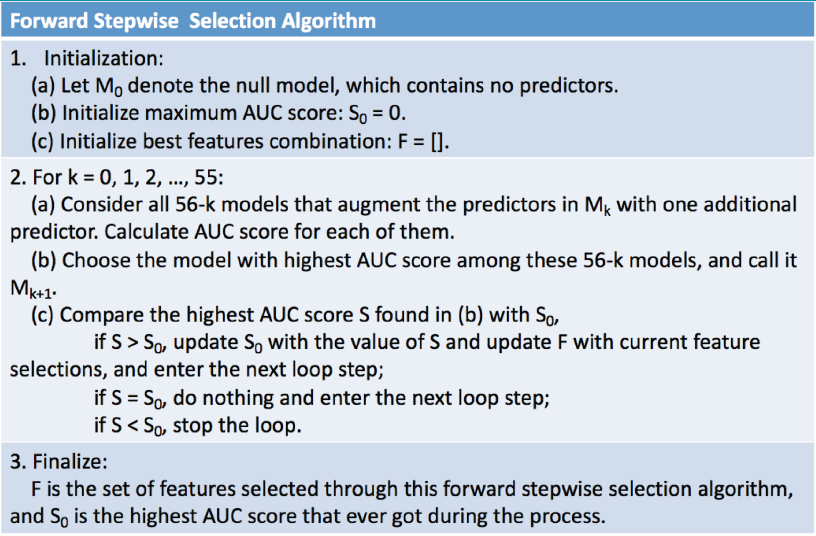
Before training all the models, we first extracted records after the first 28 days, Z-scaled all the features, randomly shuffled the observations, and then split the whole dataset into training, testing and out of time set with the 6:4:2. i.e. First 6 months for training set, next 4 months for testing set and finally last 2 months for the out of time set.

# **5.2 Forward Feature Selection**



**5.2.1 Forward Feature Selection Algorithm**

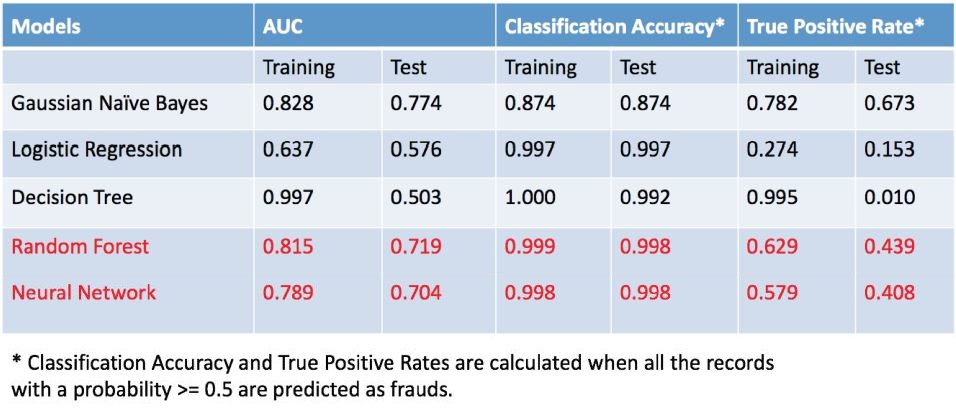
We used a wrapper feature selection method called forward stepwise selection algorithm to select features during the modeling process. The algorithm we applied is described below.



This algorithm helped us to find the features combination that gives highest AUC score on the training set in a forward-selection manner. We used AUC scores as the comparison criterion, because AUC scores take all classification cut-offs into consideration (i.e. considered all the cases that we predict probability >= p as fraud, where p could be any value between 0 and1), reflecting the overall performance of the classifier in all fraud score quantiles.

**5.2.2 Supervised Models**

We have tried five supervised models combining the forward feature selection algorithm. The performance metrics of each model are shown below.



**5.2.2.1 Gaussian Naive Bayes**

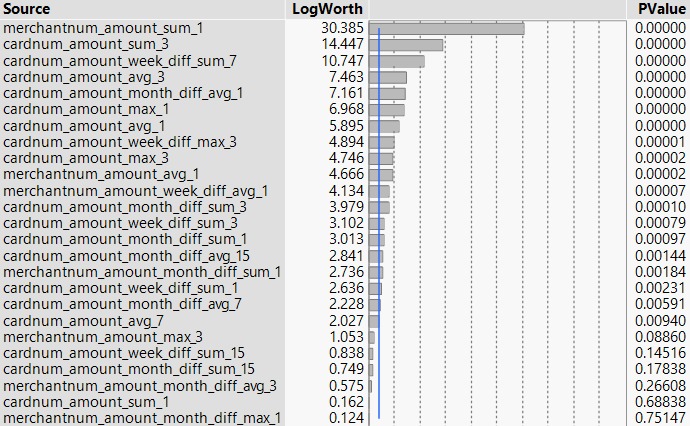
Gaussian Naive Bayes Classifier is a simple probabilistic classifier based on applying Bayes' theorem with strong (naive) independence assumptions between the features. It was the most basic model we used, and was very fast to train. Accordingly, since its assumption of independence was not well satisfied, its performance was quite poor. Nine features were selected: zip\_with\_merchnum\_7, card\_amount\_to\_median\_28, merchant\_amount\_to\_median\_28, state\_with\_merchnum\_3,merchant\_frequency\_3,card\_frequency\_7, merchant\_amount\_to\_avg\_7, card\_frequency\_3, and merchant\_amount\_to\_avg\_3. This model does not suffer from overfitting as much and has high AUC scores and True Positive Rates. However, its classification accuracy is only about 87%, which is unacceptable since there are only 0.3% frauds in the dataset. This low accuracy is caused by predicting a lot of non-frauds as frauds.

**5.2.2.2 Decision Tree**

Decision Tree partitions the feature spaces into multiple high-dimensional boxes, and give predictions according to the majority vote in each box. A single decision tree is extremely prone to overfitting. As is shown in the summary table, although the tree selected only 2 features- merchant\_amount\_to\_avg\_28, card\_amount\_to\_avg\_14, it achieved almost perfect performance on the training set by repetitively partitioning, and performed terrible on the test set.

**5.2.2.3 Logistic Regression**

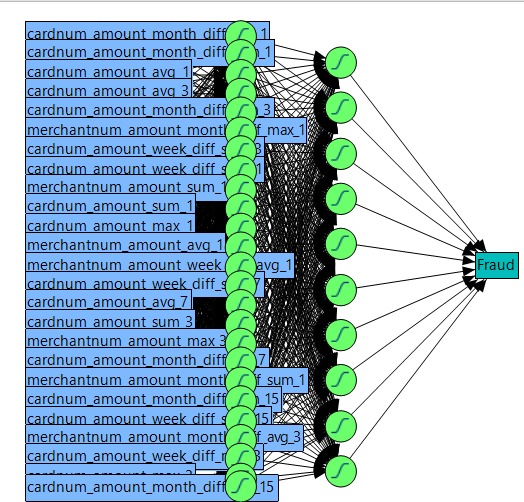
Logistic Regression model uses sigmoid function to estimate the probability, and predict probability of each record. In our case, Logistic regression selected 45 features, including zip\_with\_merchnum\_14,card\_amount\_to\_median\_7,card\_amount\_to\_max\_3,card\_amount\_to\_avg\_28, state\_with\_merchnum\_7, merchnum\_per\_card\_3,etc. Although it does not suffer a lot from overfitting, its True Positive Rate is relatively low, indicating many frauds are predicted with a low probability.



**Logistic Regression partitioning to evaluate the contribution of predictors on the response**

**5.2.2.4 Neural Network**

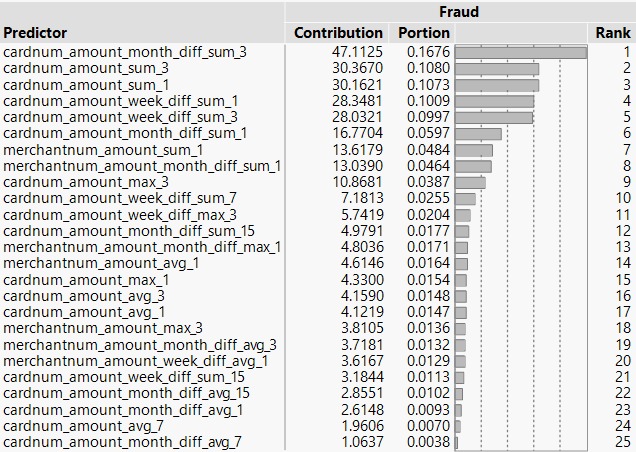
To build a neural network classifier, we used “Multi-Layer Perceptron Classifier” in Scikit-Learn package in Python. After tuning, the parameters we used are “relu” function as the activation function, L2 penalty as 0.0001, batch size is 200, learning rate is 0.001, and 100 nodes in each hidden layer. And 3 layers are built with 13 features selected: zip\_with\_merchnum\_14, card\_amount\_to\_median\_3,merchant\_amount\_to\_avg\_28,card\_amount\_to\_median\_14,merchant\_frequency\_3,card\_frequency\_3,merchnum\_per\_card\_7,cardnum\_per\_merch\_14,merchnum\_per\_card\_3,cardnum\_per\_merch\_28,merchant\_frequency\_28,merchant\_amount\_to\_median\_3,card\_amount\_to\_max\_14. The performance of Multi-Layer Perceptron Classifier is similar to random forest, and it improved on the overfitting issue.



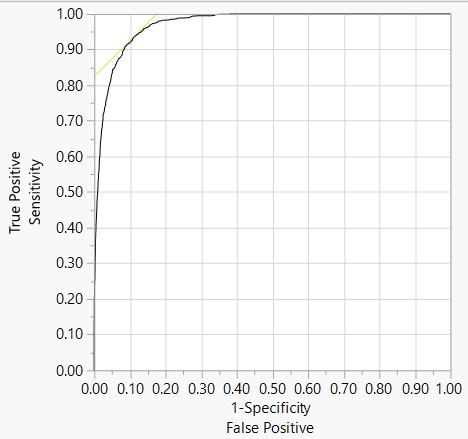
**Neural Network**

**5.2.2.5 Random Forest**

Random Forest is a modified version of decision tree, which combines multiple small trees, and considers only small amount of features at each split. Therefore, it performs far better than decision tree on the overfitting problem. It selected 9 features - zip\_with\_merchnum\_14, merchant\_amount\_to\_max\_7,cardnum\_per\_merch\_28,merchant\_frequency\_28,merchnum\_per\_card\_28,card\_frequency\_3,card\_amount\_to\_median\_28,merchant\_amount\_to\_max\_28,merchant\_frequency\_14. And the parameters after tuning are: 25 trees, max depth of each tree is 20, and minimal sample size on each terminal node is 5. It also generated good AUC score and classification accuracy. Although the True Positive Rate was not good enough on test data, better result could be achieved by lowering the prediction threshold, such as predicting all the observations with probability >= 0.3 as frauds.

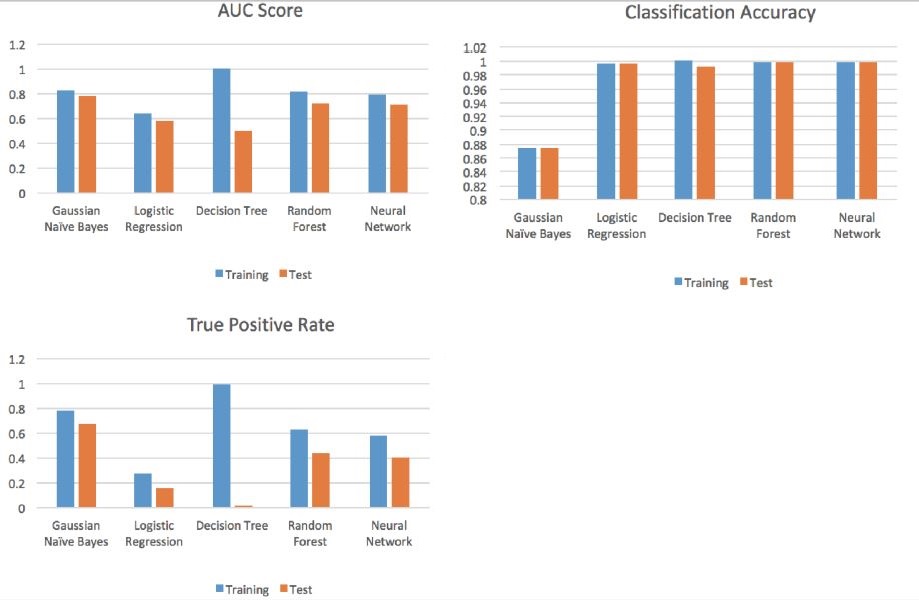


**Bootstrap forest partitioning to evaluate the contribution of predictors on the response**

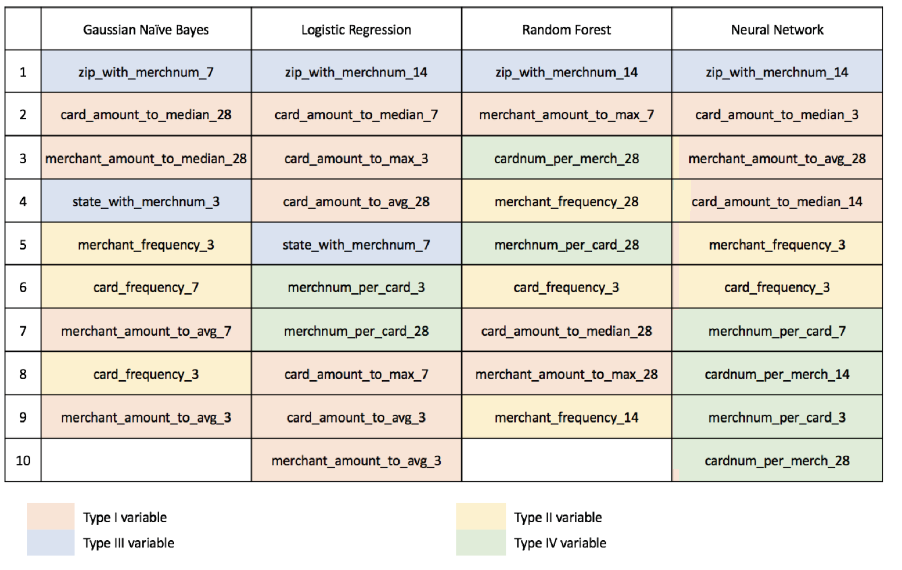
****

**ROC Plot for Random Forest : 97.128% Area under curve**

**5.2.3 Summary**

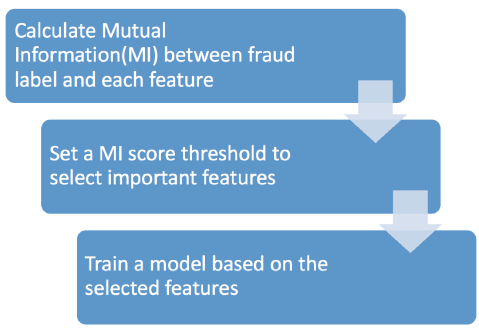


According to the above analysis and comparison, our conclusions are: Decision Tree model suffers a lot from overfitting; Gaussian Naive Bayes model has poor classification accuracy, predicting lots of non-frauds as fraud; Logistic Regression model do not overfit much, but both cannot capture enough frauds; Random Forest and Neural Network models are the best performing models among all the six models.



When we look at the top 10 selected variables (the 10 variables that are first selected) for each model, a noticeable pattern is that all the models selected type III variable - the number of different zip code related to a particular merchant number - at the very beginning, indicating that geographical pattern is a very useful feature to detect frauds. Besides, all the models selected several type I variables, since unusual transaction patterns can also play a key role in detecting frauds. Some of the Type II and Type IV variables were also selected as top 10 variables, but generally speaking, they are not as important as the other two types of variables.

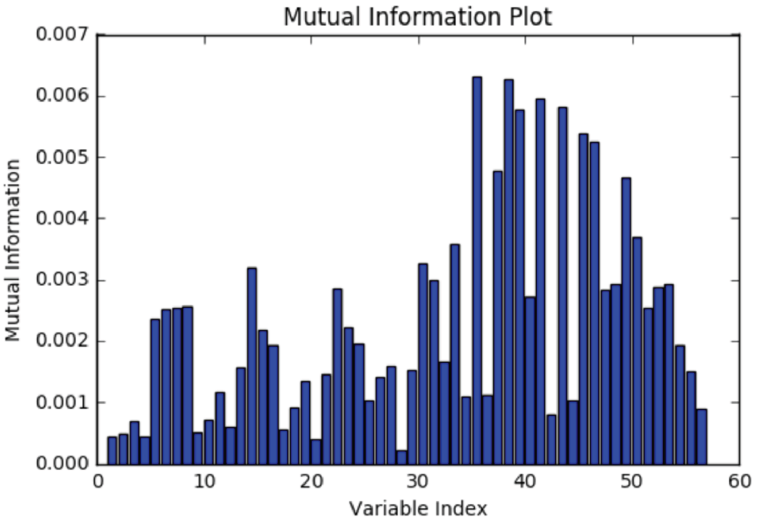
# **5.3 Feature Selection Based on Mutual Information**



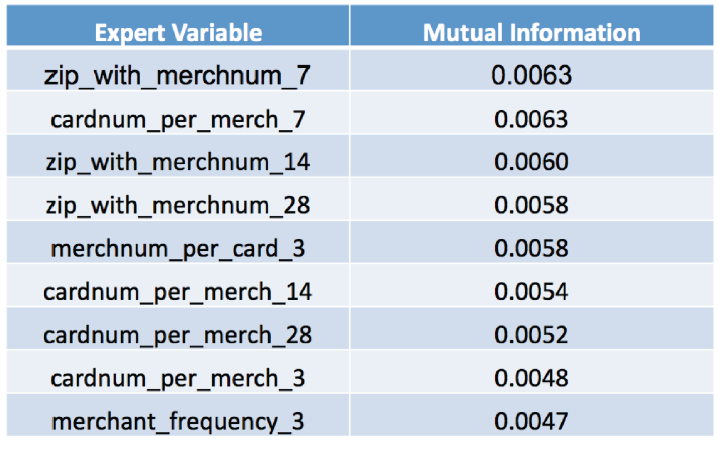
We have also tried to select features based on Mutual Information score, and trained models only on the selected important features. Mutual Information criterion was applied to select significant features in fraud analysis.

This Mutual Information (MI) method measured the mutual dependence between each variable we built and the dependent variable ‘fraud’. More specifically, it quantified the ‘amount of information’ obtained from each variable through the variable ‘fraud’ and gave a score. It is equal to zero if and only if two random variables are independent, and higher values mean higher dependency.

Before calculating the mutual information, we first z-scaled all the features to get rid of the influence of different units. Then we used the “mutual\_info\_classif” function in the Scikit-Learn package in Python to calculate the mutual information score. The mutual information calculated are shown in the graph below.



As we can see from the graph, about one-fourth of the mutual information scores are higher than 0.003. To be precise, 13 out of 56 mutual information are higher than 0.003, 9 out of 56 mutual information are higher than 0.004, and 7 out of 58 mutual information are higher than 0.005. To avoid overfitting by involving too many variables, we decided to set the threshold at 0.004. Therefore, the 9 features we selected were:



The performance of models trained with these features are shown as below. We can see the performance is not better than the features selected through forward selection method. This could be due to the fact that, when we select features based on mutual information, we only considered the dependency between each feature and fraud, but did not take into consideration the effect of combining features. For example, we selected three type III variables - zip\_with\_merchnum\_7,zip\_with\_merchnum\_14,zip\_with\_merchnum\_28, but the information they provided could be highly overlapped. Besides, some features could work much better when they are used together.

**5.4 Conclusion of the Algorithms**

****

To conclude, we have tried two feature selection methods: wrapped forward stepwise feature selection and filter feature selection method based on mutual information. Forward stepwise feature selection gives us better classification results than feature selection based on mutual information. To strike a balance between prediction accuracy and overfitting, we think the best models are Random Forest Model and Neural Network Model with features selected by the forward feature selection.

1. Results

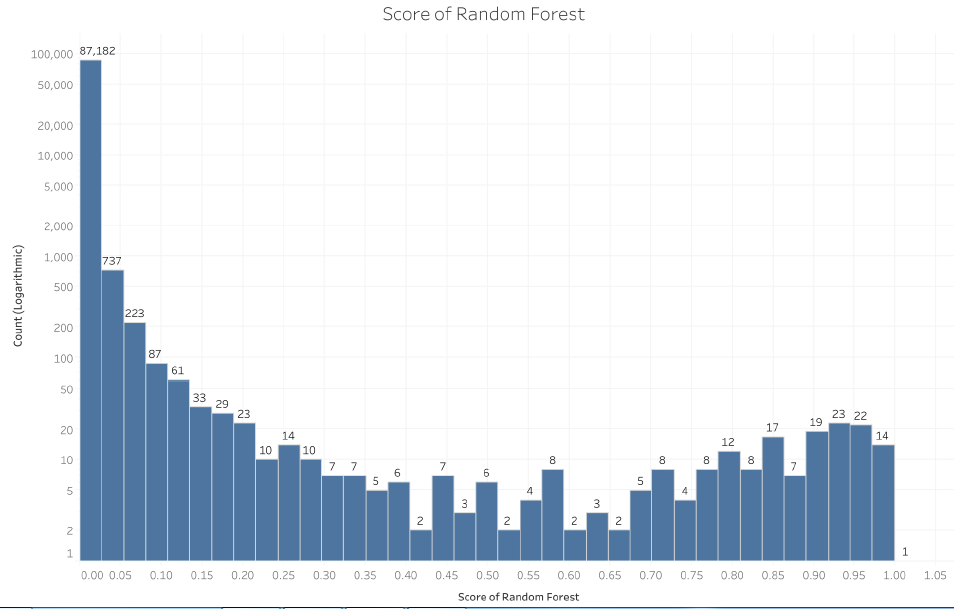
As analyzed above, we found that the models built using forward stepwise feature selection have better performance than those built using features selected based on mutual information. In the following part, we will go through the fraud detection results from Gaussian Naive Bayes Model, Logistic Regression Model, Random Forest Model, and Neural Network Model with forward stepwise feature selection method. Here we ignore Decision Tree Model, since it overfitted too much. After that, we will also show the result of ensemble modeling methods, which calculate fraud score by averaging the scores of Neural Network and Random Forest or pick the maximum score between Neural Network and Random Forest scores.

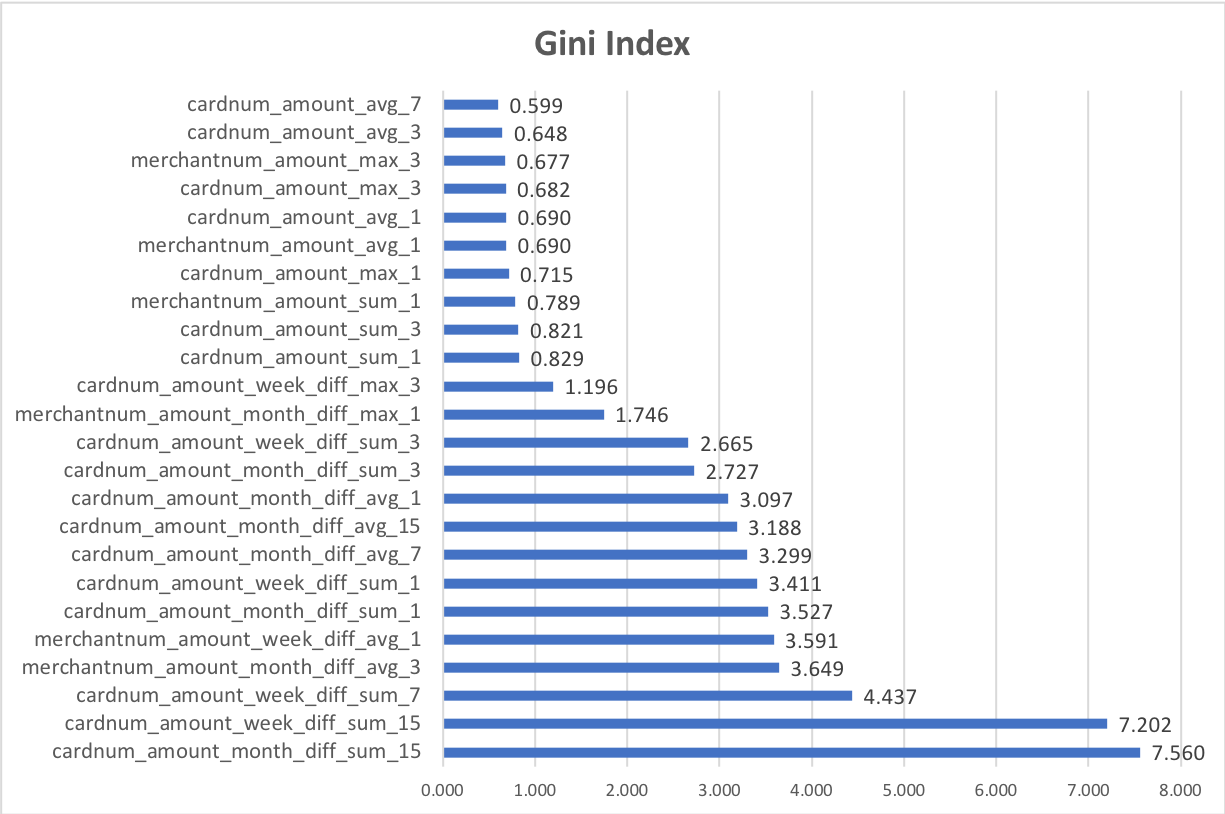


**FDR at 2% for Different Models**

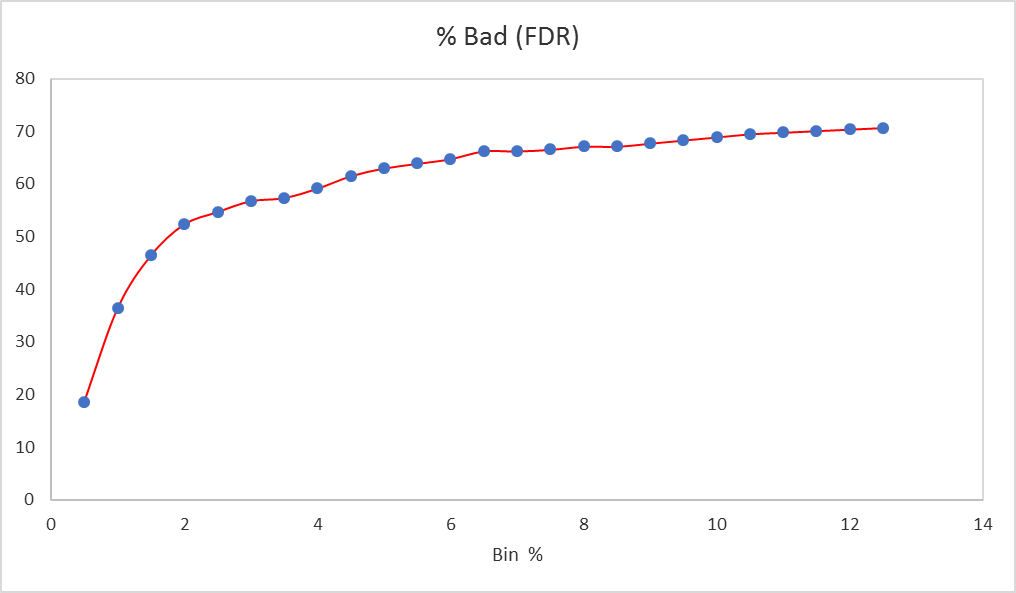
**6.1 Results of Best Model**

**Random Forest:** For this case, Random Forest outperformed the other algorithms that we tried. The below diagram shows the histogram of fraud scores from Random Forest. The fraud scores are scattered from 0 to 1 and there is a good amount of fraud scores on right i.e. records which are flagged as fraud. We used logarithmic scale on y-axis for clarity.



****



****

The plot below demonstrates the business values of Random Forest Model.

**Assumptions :**

* $1000 loss for every fraud that’s not caught.
* $10 loss for every false positive (good that’s flagged as a bad)
* Cutoff point should be 16.5% because beyond that the ROI becomes flat.



**6.2 Conclusion**



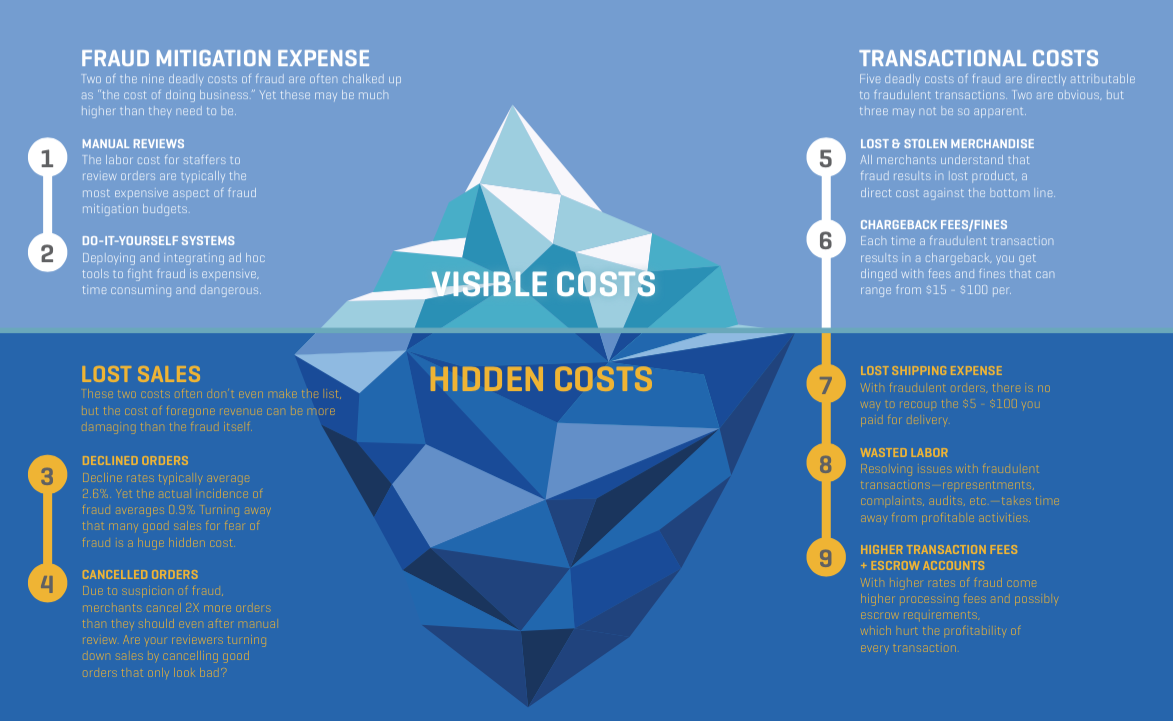
From comparing all the models and their performances, we could decide that Random Forest Model with the 9 variables selected during the forward feature selection process performed best.

And to generate most business value, i.e. ROI, we should flag all the records with top scores as frauds. The estimated ROI using the given dataset is $231,710.

|  |  |  |  |
| --- | --- | --- | --- |
| **Population Bin(%)** | **Fraud Savings ($)** | **Lost Sales ($)** | **ROI** |
| 0.5 | 63000 | 0 | 63000 |
| 1 | 123000 | 30 | 122970 |
| 1.5 | 157000 | 320 | 156680 |
| 2 | 177000 | 750 | 176250 |
| 2.5 | 185000 | 1300 | 183700 |
| 3 | 192000 | 1860 | 190140 |
| 3.5 | 194000 | 2470 | 191530 |
| 4 | 200000 | 3040 | 196960 |
| 4.5 | 208000 | 3590 | 204410 |
| 5 | 213000 | 4170 | 208830 |
| 5.5 | 216000 | 4770 | 211230 |
| 6 | 219000 | 5370 | 213630 |
| 6.5 | 224000 | 5950 | 218050 |
| 7 | 224000 | 6580 | 217420 |
| 7.5 | 225000 | 7200 | 217800 |
| 8 | 227000 | 7810 | 219190 |
| 8.5 | 227000 | 8440 | 218560 |
| 9 | 229000 | 9050 | 219950 |
| 9.5 | 231000 | 9660 | 221340 |
| 10 | 233000 | 10270 | 222730 |
| 10.5 | 235000 | 10880 | 224120 |
| 11 | 236000 | 11500 | 224500 |
| 11.5 | 237000 | 12120 | 224880 |
| 12 | 238000 | 12740 | 225260 |
| 12.5 | 239000 | 13360 | 225640 |
| 13 | 241000 | 13970 | 227030 |
| 13.5 | 243000 | 14580 | 228420 |
| 14 | 245000 | 15190 | 229810 |
| 14.5 | 245000 | 15820 | 229180 |
| 15 | 248000 | 16420 | 231580 |
| 15.5 | 248000 | 17050 | 230950 |
| 16 | 249000 | 17670 | 231330 |
| **16.5** | **250000** | **18290** | **231710** |
| 17 | 250000 | 18920 | 231080 |
| 17.5 | 251000 | 19540 | 231460 |
| 18 | 251000 | 20170 | 230830 |
| 18.5 | 251000 | 20800 | 230200 |
| 19 | 251000 | 21430 | 229570 |
| 19.5 | 251000 | 22060 | 228940 |
| 20 | 251000 | 22690 | 228310 |
| 20.5 | 251000 | 23320 | 227680 |
| 21 | 251000 | 23950 | 227050 |
| 21.5 | 251000 | 24580 | 226420 |
| 22 | 251000 | 25210 | 225790 |
| 22.5 | 252000 | 25830 | 226170 |
| 23 | 252000 | 26460 | 225540 |
| 23.5 | 253000 | 27080 | 225920 |
| 24 | 253000 | 27710 | 225290 |
| 24.5 | 254000 | 28330 | 225670 |
| 25 | 254000 | 28960 | 225040 |
|  |  | **max ROI** | **231710** |

**6.3 Business Insights and Real-World Scenario**

****

****

Appendix

Data Quality Report

**Dataset:** Card Transaction Data

**Description:** This data is a simulated representation of the 96708 card transaction details applications during 2010. A typical observation contains information about the card and merchant details, along with geographical location parameters and dollar amount transacted.

**Number of Records:** 96708

**Number of Variables:** 10

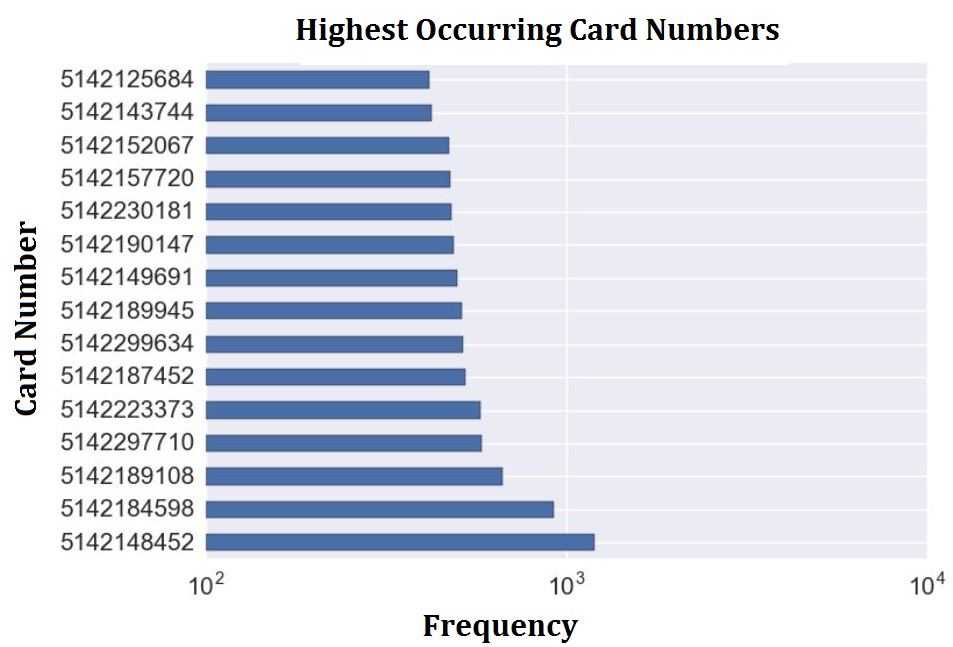
**Summary statistics for numerical variables**



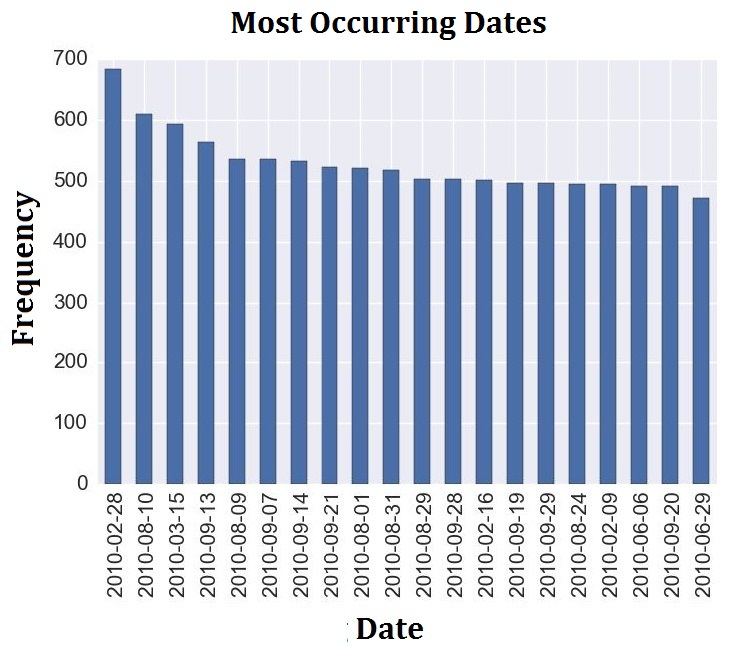
**Description and visualization of data**

**1. *Recordnum*** is a categorical variable. It works as the ordinal reference number for each property record. There are 96708 records overall. Each row is a unique number/identifier and hence, a visualization is not required.

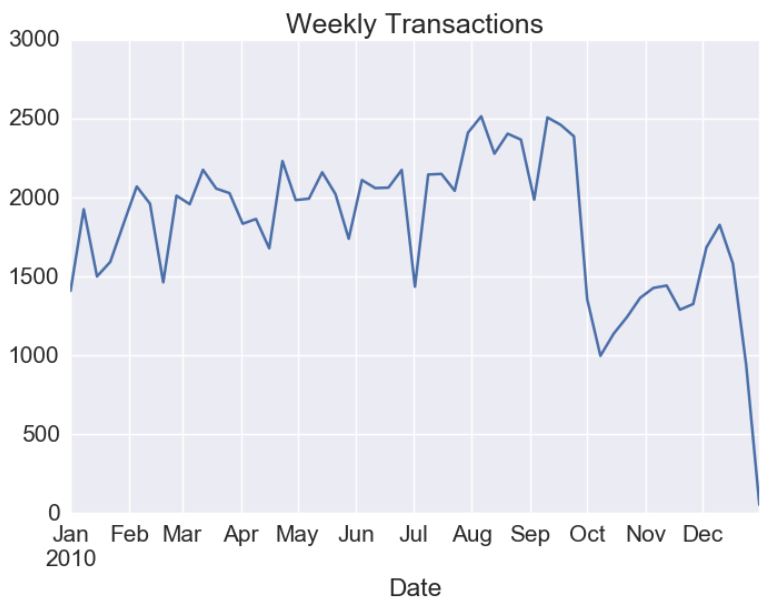
**2. *Cardnum*** refers to the Card Number used to make transaction.

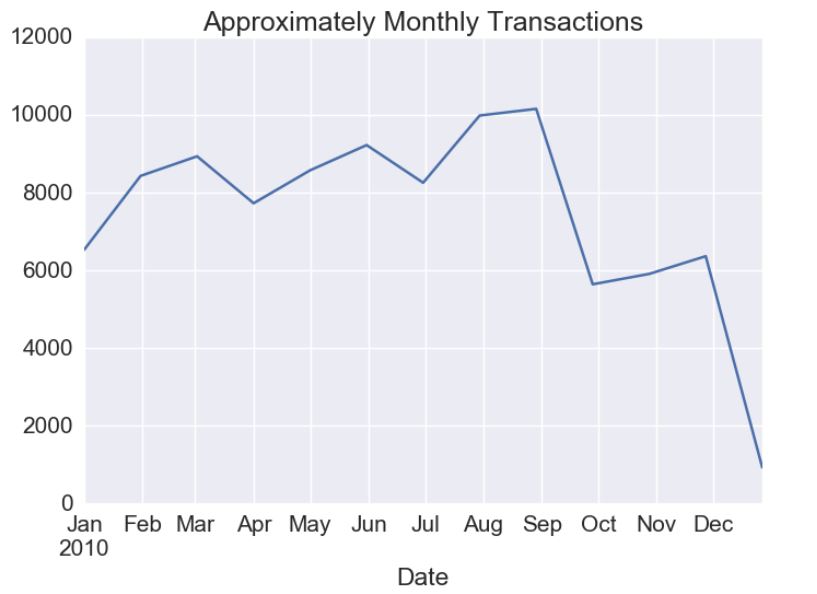


**3. *Date*** refers to the date on which the transaction was made.

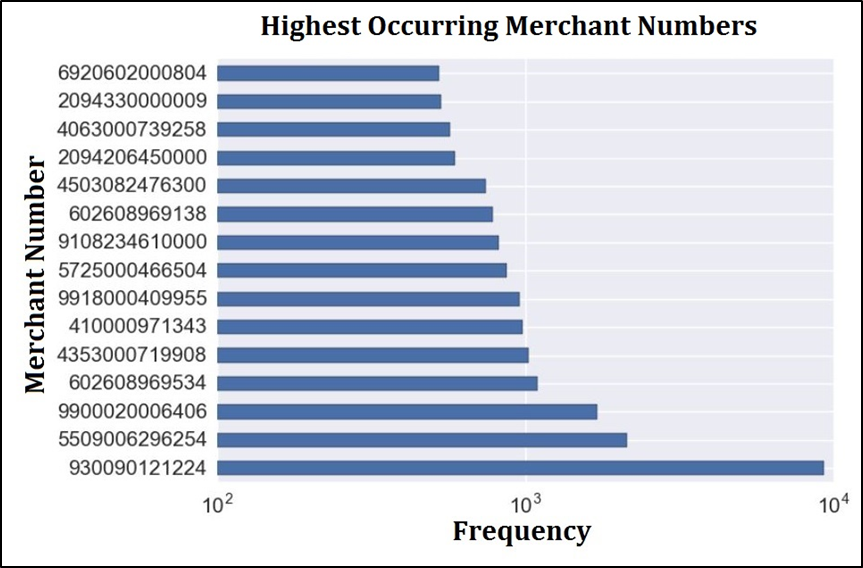
****



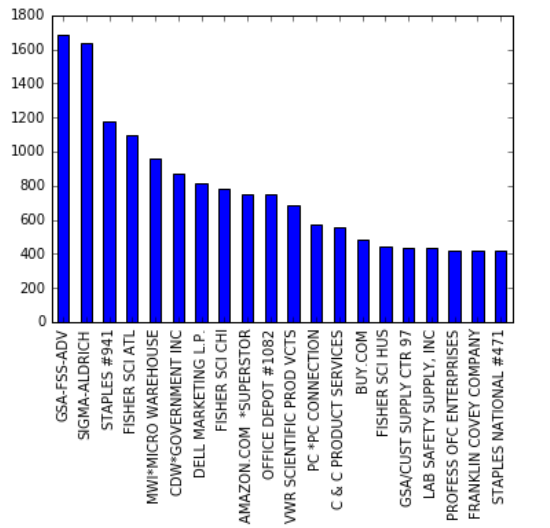




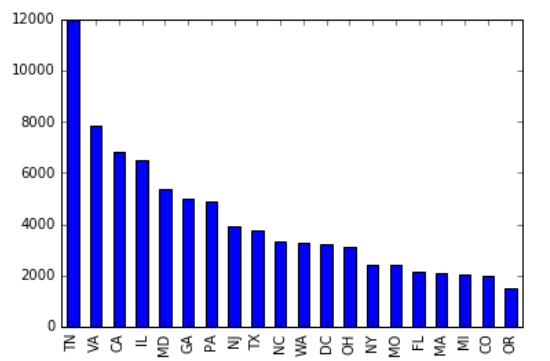
**4. *MerchantNum*** refers to unique Merchant ID associated with each transaction.



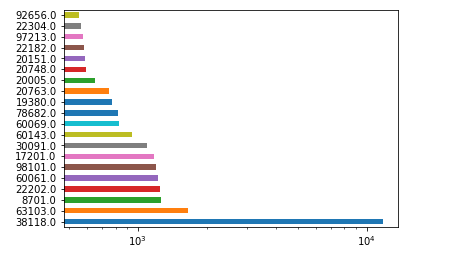
**5. *Merch Description*** refers to the name of the merchant which carried out the transaction. 20 most frequently occurring merchants have been shown below.



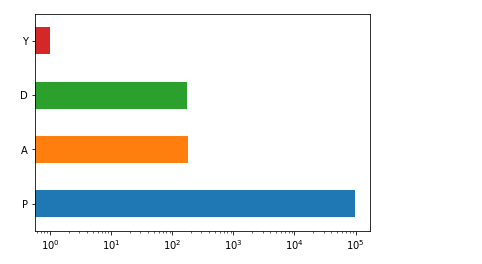
**6. *Merchant State*** refers to the state in which the merchant is located. States are abbreviated using 2-letter codes.



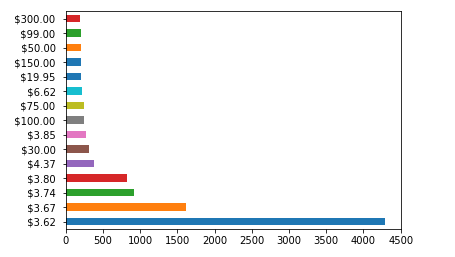
**7. *Merchant Zip*** contains information about each merchant’s ZIP code. Below is a graph of the top 20 most frequent zip codes.



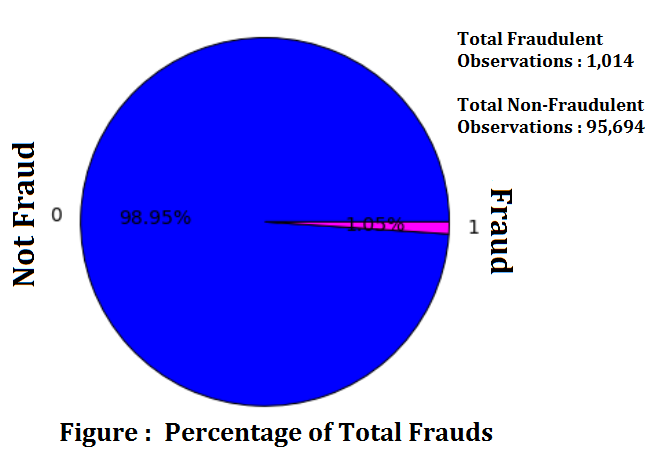
**8. *Transtype*** is a categorical variable referring to the type of transaction. There are four different types of transactions in the dataset with no missing values.



**9. *Amount*** contains information about the dollar amount corresponding to each transaction. Below is a graph of the top 15 most frequent transaction amounts.



**10. *Fraud*** contains information whether observation includes fraudulent data or not. Of the total 96,708 observations, we observed that 95,694 observations were noted as accurate/legitimate, while 1,014 observations were marked as fraudulent.



**Summary**

The Data Quality Report created for the Card Transaction dataset is required for the integrity of the data management by covering gaps of data issues. We not only cleaned and transformed the data, but also rectified inconsistencies and redundancies. Furthermore, we tabulated and visualized all variables, which provided us new insights into the dataset. These reports are valuable when administered on data that has made multiple iterations and additions before that data becomes authorized or stored for enterprise intelligence. As we gather more data to add to our current database, we can construct similar accuracy checks for all data sourced to our class project.

**Future Scope**

This report will aid data governance by monitoring data to find exceptions and anomalies undiscovered by previous data management operations. Further data quality checks may be defined at attribute level to have full control on its remediation steps. This report will serve as a cornerstone for all future phases of the group project.