5/9/2021

DSO 562 Project 3

Transaction Fraud in Credit Cards

Team Member: 18

Xue Gao,
Tianze Yang,
Yang Jiao,
Yunlong Wang,
Huiyi Zhang,
Xinzhu Zhang

University of Southern California
MARSHALL SCHOOL OF BUSINESS

Table of Contents

Executive Summary

As the most prevalent form of payment other than cash, credit cards offer stream-lined and versatile experience to consumers and merchants alike. Consumers get to tap into additional liquidity to finance purchases at their own pace without the scrutinies in the usual loan underwriting process as observed in the student-loan or real-estate business. On the flip side, merchants get paid expeditiously while rendering their products more accessible to the customers.

Such convenience comes at a price of safety. In the year of 2018, it is estimated that $9.47 billion were lost in illegal transactions. With the exploding expansion of online shopping, more and more frauds are committed where no physical card is present. A fraudster simply, via one method or another, records the embedded information from the targeted cards and swipe away at a time and location of their choice, behind a computer screen or even a phone. These unscrupulous acts severely dampen the trust between the parties of transactions, create friction and raise cost for all involved.

In this project, we attempt to build a solution against fraudulent transactions for the card issuer, who facilitates the transactions as a service, and often soaks up the lost as a carefully calculated risk. The goal is simple, we design a model that catches as many frauds as possible without flagging too many transactions.

The simulated dataset provides information such as card number, transaction amount, date, and merchant info. We will utilize machine learning algorithms to pick up the underlying patterns and assign a probability that each given transaction is indeed fraudulent. We then rank these in order and identify the top 3% of all transactions and tally the proportion that is really fraud, i.e., Fraud Detection Rate at 3%, which will be used as a metric to measure the models against each other. Due to the relatively small data size, the FDRs are averaged across at least 10 runs to reduce potential influence from randomness,

After painstaking data cleaning and thoughtful feature engineering, we yielded a total of 671 variables, which were later filtered down to 80 based on an average score rank of FDR and KS. Around 30 features remain after the wrapper method, which we believe captures the most critical traits that distinguishes the classes. We fitted dozens of models including decision trees, Logistic regression, KNN, Random Forest, Boosted Tree, and Neural Net, and eventually settled for a random forest model with 20 features, max_features at 7, n_estimators at 90 and max_depth at 10.

This final model provided the most robust performance where the out-of-time validation data yielded the FDR at 3% as 61.45%. Since the oot dataset was never presented in the modeling process, we believe this number would hold against other samples from the same population. See more details in the following report and appendix.

Data Description

Card transactions data is a dataset of 12-month actual card purchases from a US government organization. The purpose of the dataset is to find card transaction fraud. The dataset contains around 96,753 records. There are 10 fields, 1 numerical field, 1 date field, and 8 categorical fields.

Numerical Field:

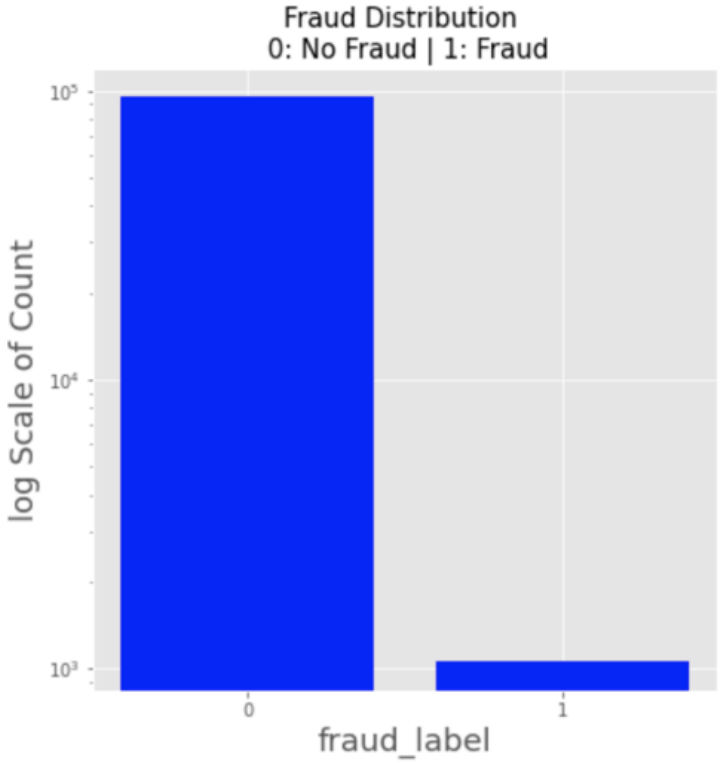
Column Name	# of Records	% populated	Unique Values	Mean	Standard Deviation	Minimum Value	Maximum Value	# Zeros
Amount	96753	100	34909	427.8857	10006.14	0.01	3102046	0

Categorical Fields:

Merchnum, Merch state and *Merch zip* are not 100% populated.
No frivolous values in the most common field value.

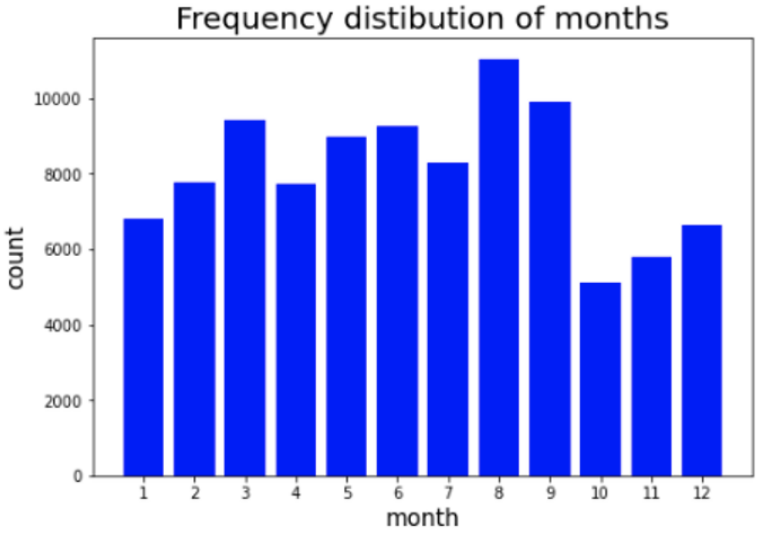
Column Name	# of Records	% populated	Unique Values	Most Common Field Value
Recnum	96753	100	96753	2047
Cardnum	96753	100	1645	5142148452
Date	96753	100	365	2/28/10
Merchnum	93378	**96.51**	13091	930090121224
Merch description	96753	100	13126	GSA-FSS-ADV
Merch state	95558	**98.76**	227	TN
Merch zip	92097	**95.19**	4567	38118
Transtype	96753	100	4	P
Amount	96753	100	34909	3.62
Fraud	96753	100	2	0

Fraud: whether the card transaction is fraud
0: No Fraud | 1: Fraud
Only 1.1% of the records are fraud.

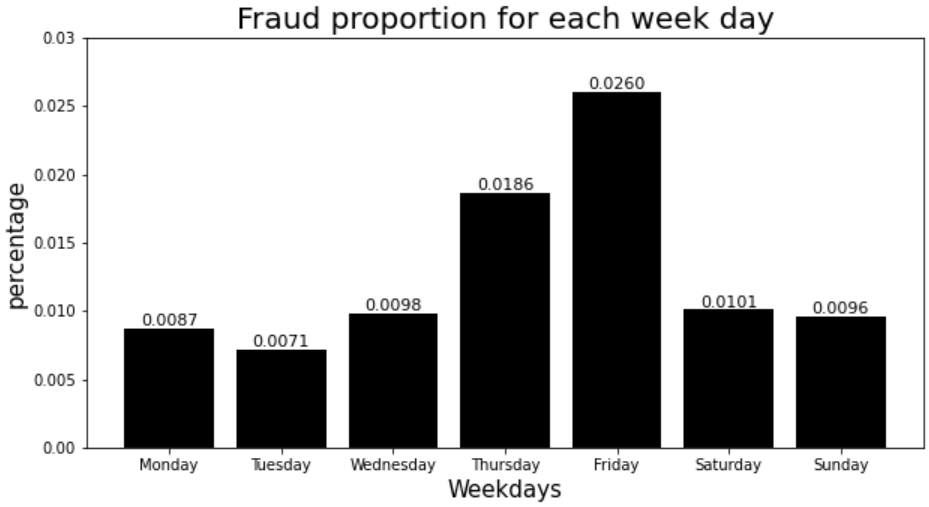


CardNum: Card Number

Date: The date on which the card transaction occurred
Number of transactions reached its peak in August and dropped down in October.



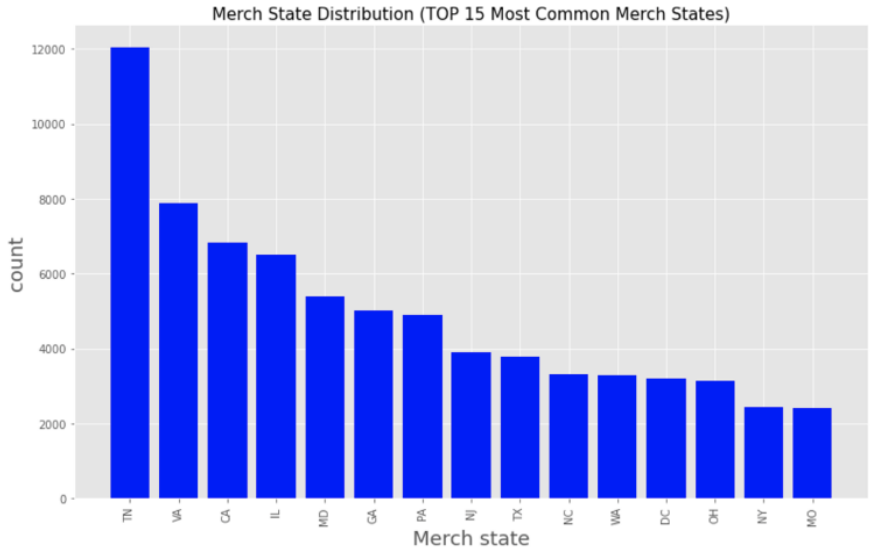
Thursdays and Fridays have the highest risk.



Merchnum: The identifier of the merchandise

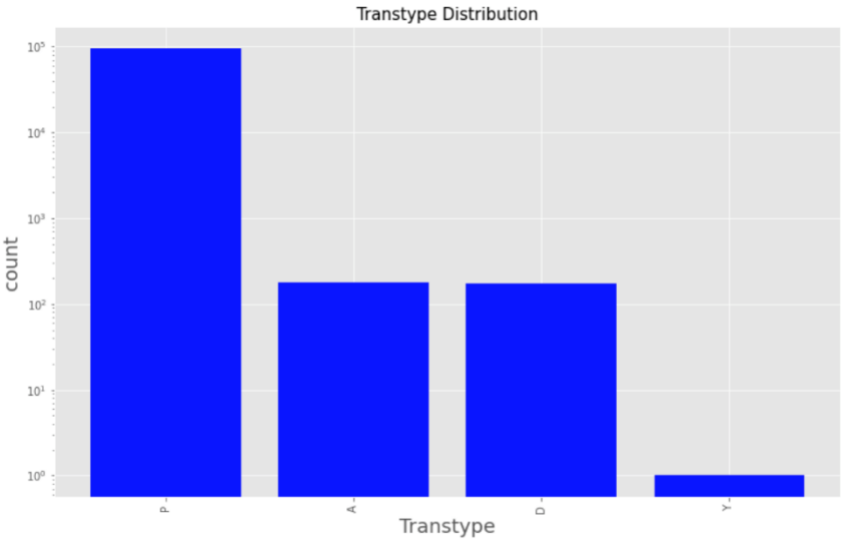
Merch description: Description of the merchant

Merch state: The location of the merchant branch where the purchase is made
The histogram shows the top 15 states where the transactions happened. The greatest number of transactions took place in Tennessee.

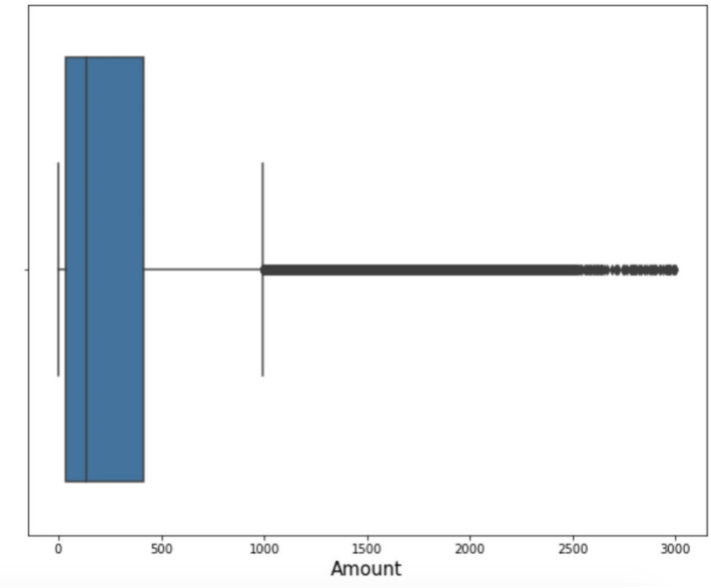


zip: the zip code of the merchant address

Transtype: Type of transaction
There are four types of transaction records. Most of them are type P, which means payment or purchase. That is the type we focus on in this project.



Amount: The dollar amount of the transaction
Most of the transaction amounts are below 1000.



The histogram shows the amount below 1000. A lot of the records have relatively small amounts.


Amount Distribution
0.0025
0.0020
0.0015
0.0010
0.0005
0.0000
0
200
400
600
800
1000
Amount

Data Cleaning

The original dataset has a total of 10 fields. Seven of the ten fields are 100% populated. The missing values are only in Merch num, Merch zip and Merch state fields.

Before we filled the missing values, we firstly verified if the merchant zip and merchant state matched. We used a file that contained zip code range for each state to verify if the zip code and state match. Then we found there are 1407 unmatched records. For example, it shows the zip code of the Washington Service Bureau is 20005 in Washington state, but it is in Washington DC. We addressed this kind of problem by the following steps.

Firstly, we extract two subsets from the whole dataset. One subset is the records that zip and state match, and another one is the records that state and zip do not match. Then we merged these two subsets on description to correct as many zips and states as possible.

Secondly, we grouped the 1407 unmatched transaction data by description, state and zip forming 200 groups. The 200 groups meant that there were 200 different merchants having unmatched zips and states. Then we started searching the 200companies online one by one and corrected them manually. We followed these principles:

- If there is one store in its zip code area, we corrected the state.
- If there is only one store in the state, we corrected its zip. Also, if the zip was missing, we filled it.
- Some companies indicated the state in their merchant descriptions then we knew the zip might be wrong. For example: a merchant called “Florida Plants”, the state was Florida, but zip was in New York. In this case, we thought the state should be right and we corrected the zip.

However, there were some limitations. If the company has many stores in one state, we still did not know its accurate zip. We corrected the wrong zip as unknown. Besides, some of the stores have been shut down and there was no more information online. We just left them as they were.

At this point, we mainly corrected the zip and state. Next, we started filling the missing values.

Since only Merch num, Merch zip and Merch state are not 100% populated. So, we only need to fill the three feature missing values.

For the merchant state feature, there were 1194 missing values initially. Some of them had valid zips but no state values. In this case, we used the zip file again to fill the states. We filled 70 values.

For those records that do not have neither zip nor state we had to group by different meaningful entities several times to fill the values. For example, we grouped by card number and date. We got the mode states for these groups and filled them in. Because we believed it is more likely that one person purchased products in the same state in one day. Then we grouped by card number and merchant description. Since a card holder would purchase more at where he or she lives. We also grouped by merchant number and grouped by card number. In this way, we filled 1092 records. But there were still 32 missing values and we filled them as unknown.

For the merchant zip feature, there were 4616 missing values initially. We have six different group by:

1. We grouped by cardnum, date, merch state and merch description to get the mode zip.
2. We grouped by merch state and merch description to get the mode zip.
3. We grouped by merch state and merch number to get the mode zip.
4. We grouped by card number and date to get the mode zip.
5. We grouped by merch state to get the mode zip.
6. We grouped by card number to get the mode zip.

We have filled 4613 values through grouping by, then we filled unknown for the left 3 records.

For merchant number we did a similar process. There were originally 3374 missing values. We also did 6 times group by:

1. We grouped by merch state, merch zip and merch description to get the mode of the merchant number.
2. We grouped by merch state and merch description to get the mode of the merchant number.
3. We grouped by merch zip and merch description to get the mode of the merchant number.
4. We grouped by merch state to get the mode of the merchant number.
5. We grouped by merch description to get the mode of the merchant number.
6. We grouped by merch zip to get the mode of the merchant number.

Using the group by six times, we filled all the missing values in the merchant number field.

In the end, we removed a single large problematic transaction. We also only kept P type transaction records.

Candidate Variables

While crafty, fraudulent behaviors are not without a trace. It is often observed that criminal acts are accompanied by hysteria or desperation. Attempted frauds, therefore, often show a burst of activity in a short amount of time. Firstly, since this project is credit transaction fraud and there are several symbols of credit card transaction fraud. For example, there are activities erupt at different businesses, which means many transactions took place in an unusual merchant. The second symbol is transactions exceeding the normal purchase amount for the same or different merchants and this means the amount is bigger than the regular credit card purchase of the same card number. The third symbol is some transaction took place at merchants not used before. The last symbol is increased usage in card-not-present. Signatures as such help us put a handle on an otherwise unapproachable problem.

According to those fraud symbols, we decided to create 4 kinds of variables: Amount variables, Day Since variable, Frequency variable, and Relative Frequency variable. The amount variable is the reflection of abnormal transactions. The day since variable, frequency variable, and relative frequency variable are the reflection of abnormal activities erupt at different businesses, transactions took place at merchants not used before.

Then we linked the original fields and created new entities:

Cardnum_merch: Cardnum+Merchnum

Cardnum_zip: Cardnum+Merch zip

Cardnum_state：Cardnum+Merch State

Merchnum_zip: Merchnum+Merch zip

Merchnum_state: Merchnum+Merch State

Cardnum_merchdes: Cardnum+Merch description

Merchdes_zip: Merch description+Merch zip

Merchdes_state: Merch description+Merch state

In total, we created 11 entities

For the amount variable, we calculate the mean, max, total, median and use the actual number of amounts for a record to divide those calculations with the total number of entities over 0, 1, 3, 7, 14, 30 days and we get 528 amount variables.

Another group day since variables mean how many days has passed since the last time the entities appeared. Total number is 11 vars.

Then, we built frequency variables, and they represented the proportion of the number of times we have seen that entity in the past days comes from the recent past. We calculate the number of those entities appearing for the past 0,1,3,7,14, 30 days. Total number is 66 vars.

In the end, we built relative frequency variables, and they represented the proportion of the number of times we have seen that entity in the past days comes from the recent past. We use the number of the entities appearing in the last 0,1 days to divide by the number of the same entities that appear for 7, 14, and 30 days.

Finally, we got 671 variables.

Variable groups	Variable name format
Amount	entities_avg_xx, entities_max_xx, entities_med_xx, entities_total_xx, entities_actual/avg_xx, entities_actual/max_xx, entities_actual/med_xx, entities_actual/total_xx
Day since	entities_day_since
Frequency	entities_count_xx
Relative Frequency	entities_count_yy_by_xx

xx: 0, 1, 3, 7, 14 30days ； yy: recent days

Feature Selection

After creating variables, we standardized all the features by z-scaling and separated the cleaned data into modeling and out-of-time dataset. Modeling dataset includes training and testing, which is from January 15th to October 31st. Out-of-time dataset contains data from November 1st to December 31st. We only used modeling data to do feature selection.

We created 671 variables and some of them may be either highly correlated with others or not significant enough to predict the output variable. We performed the filter method to drop the number of variables to around 80 and further reduced the feature dimension by wrapper method.

Filter Method

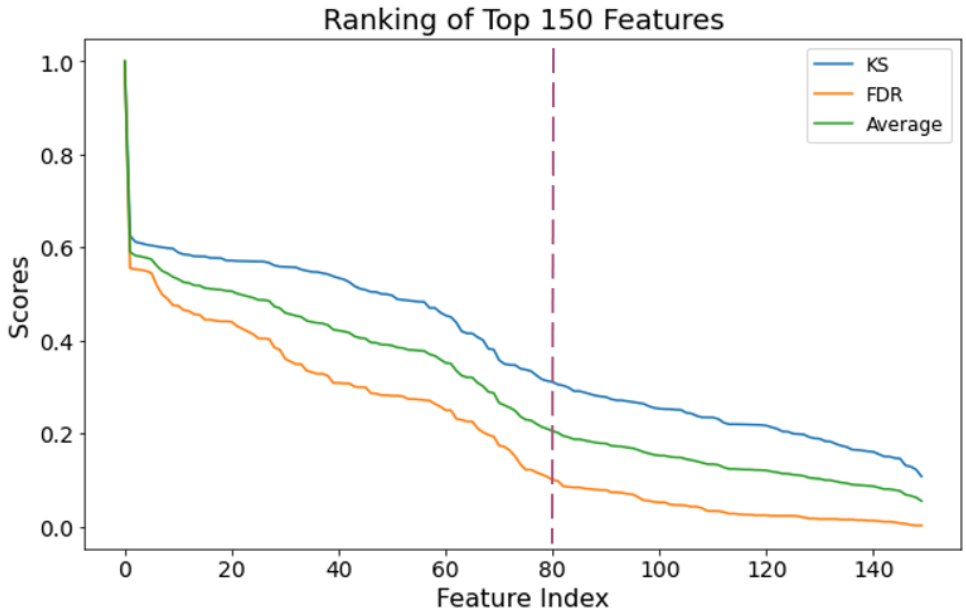
Filter method is independent of any modeling methods and we can know the importance of each variable to predict the output value. We performed the filter method by using Kolmogorov-Smirnov (KS) score and fraud detection rate at 3%.

Kolmogorov-Smirnov (KS) Score:
KS score is a common measure for classification problems. A higher KS score means a better separation between two distributions. We used KS score to measure the differences between fraud records and non-fraud records for each variable created. Specifically, for each variable, we gathered a list of fields corresponding to fraud records and the other list of random numbers between 0 and 1. Then we applied stats.ks.2samp function to compute KS for all variables.

Fraud Detection Rate (FDR):
In general, fraud detection rate is the percentage of all the fraud found at a score cutoff. For this project, we used 3% as our cutoff threshold and calculated the FDR for each candidate variable. Specifically, for each variable, we sorted the data in descending order and calculated the percentage of fraud records in the top 3% of the population.

We calculated the KS and FDR scores for each variable and sort them in decreasing order. We also calculated the average ranking of the two scores in order to filter variables.

We plotted the relationship between three scores and the top 150 features. The following figure shows the score difference becomes very small after 80 features. Therefore, we chose to keep the top 80 features and used those features in the wrapper method.



The table below shows the top 10 variables of our filter result, the highest KS score is around 0.68 and the highest FDR score is around 0.64.

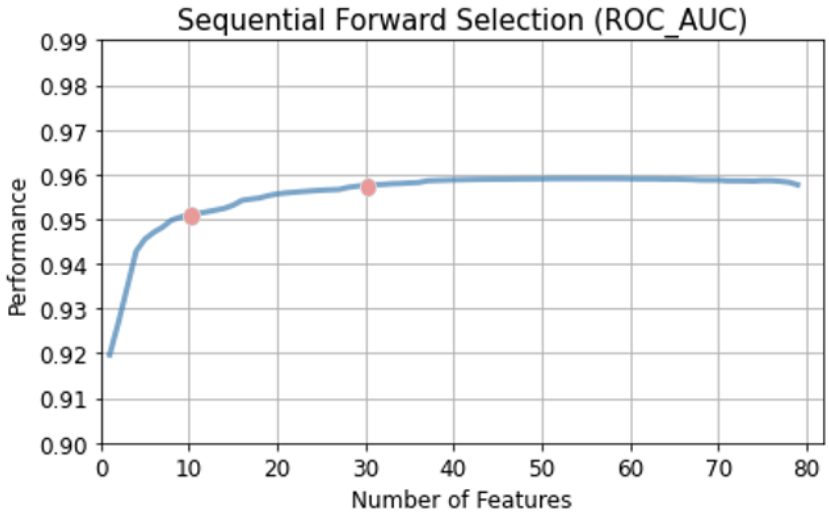
	KS	FDR	Avg	KS Rank	FDR Rank	Average Rank
Fraud	1.000000	1.000000	1.000000	1.0	1.0	1.00
cardnum_zip_total_7	0.685118	0.634793	0.659955	2.0	6.0	4.00
cardnum_zip_total_3	0.677772	0.637097	0.657435	4.0	4.0	4.00
cardnum_merch_total_7	0.681401	0.633641	0.657521	3.0	7.0	5.00
cardnum_merchdes_total_7	0.671453	0.638249	0.654851	9.0	3.0	6.00
cardnum_merch_total_14	0.675820	0.631336	0.653578	5.0	8.5	6.75
cardnum_merch_total_3	0.675293	0.631336	0.653315	6.0	8.5	7.25
cardnum_merchdes_total_3	0.661562	0.642857	0.652210	13.0	2.0	7.50
cardnum_state_total_3	0.673721	0.627880	0.650801	7.0	10.0	8.50
cardnum_merchdes_total_14	0.665663	0.635945	0.650804	12.0	5.0	8.50
cardnum_zip_total_14	0.672152	0.625576	0.648864	8.0	11.0	9.50

Wrapper Method

Filter method ignores correlations so we used forward stepwise selection as our wrapper to remove correlations and further reduced the number of variables. We used logistic regression with an L2 penalty as our model and ROC_AUC score is the measure of goodness. This allows us to see how performance changes on different subsets of variables.

The plot below shows the performance of the model from the forward selection with different numbers of variables. From the plot, we can see the performance improves a lot from 1 to 10

features, improves a little from 10 to 30 features and becomes stable after 30 features. According to this plot, we decided to build models based on 20 and 30 features.

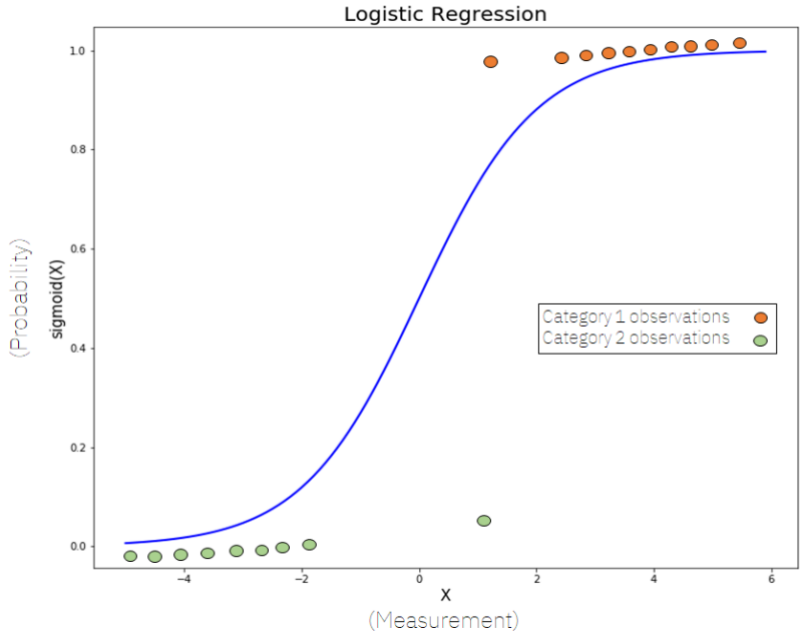


The table below shows the final list of our 20 features.

20 Features	
cardnum_zip_total_7	cardnum_state_max_7
cardnum_merchdes_total_7	cardnum_merch_max_14
cardnum_state_total_3	cardnum_total_7
cardnum_merchdes_total_14	merchnum_zip_total_1
cardnum_zip_total_14	merchdes_state_total_0
cardnum_state_total_7	merchdes_zip_max_0
cardnum_merch_total_1	merchdes_zip_total_0
cardnum_merch_total_30	merchdes_zip_max_1
cardnum_merchdes_max_14	cardnum_total_0
cardnum_merchdes_total_0	cardnum_merchdes_max_0

Model Algorithms

Logistic Regression



Logistic regression is the appropriate regression analysis to conduct when the dependent variable is a binary variable. It is used to describe the relationship between one dependent binary variable and a set of independent variables. It does output the predicted probability of an outcome that has two values (i.e. 0 and 1).

Instead of fitting a straight line or hyperplane, the logistic regression model uses the logistic function to squeeze the output of a linear equation between 0 and 1. The logistic function is defined as :

$$\text{logistic}(\eta) = \frac{1}{1 + exp(-\eta)}$$

In the logistic regression the constant moves the curve left and right and the slope defines the steepness of the curve. By simple transformation, the logistic regression equation can be written as: p(X)=e0+1X1+...+nXn1+e0+1X1+...+nXn.

With the fraud label being the dependent binary variable and the candidate variables we selected before being the independent variables, we trained the 5 models in total with a number of variables

ranging from 10 to 40 and listed all average FDR at 3% across 10 times for each hyperparameter choice. We used logistic regression as our baseline model to get the basic performance we can get without overfitting problems. The best model is the highlighted one with 30 variables and average FDR at 3% of 30.44%.

Model	Hyperparameter		Average FDR(%)		
Logistic regression	num_vars	penalty	Train	Test	OOT
1	10	L2	0.67411	0.673559	0.267039
2	15	L2	0.686041	0.677296	0.272067
3	20	L2	0.684966	0.67865	0.300559
4	30	L2	0.685718	0.683575	0.304469
5	40	L2	0.686516	0.68506	0.277654

Decision Tree

The decision tree is a rudimentary algorithm and a critical component to more robust ensemble methods such as random forest and xgboosting. It divides individual observations into distinct groups based on the value of features. The algorithm quickly adapts to find the best way to minimize impurities in each grouping either by the Gini or entropy value. It does this so effectively that it quickly overfits the data by dividing too many times or too deep a depth. In such scenarios, it helps to set a max_depth feature in the sci-kit learn package, which limits the highest number of times a tree could divide, and effectively deals with over-fitting so a model works well for other samples of data from the same population.

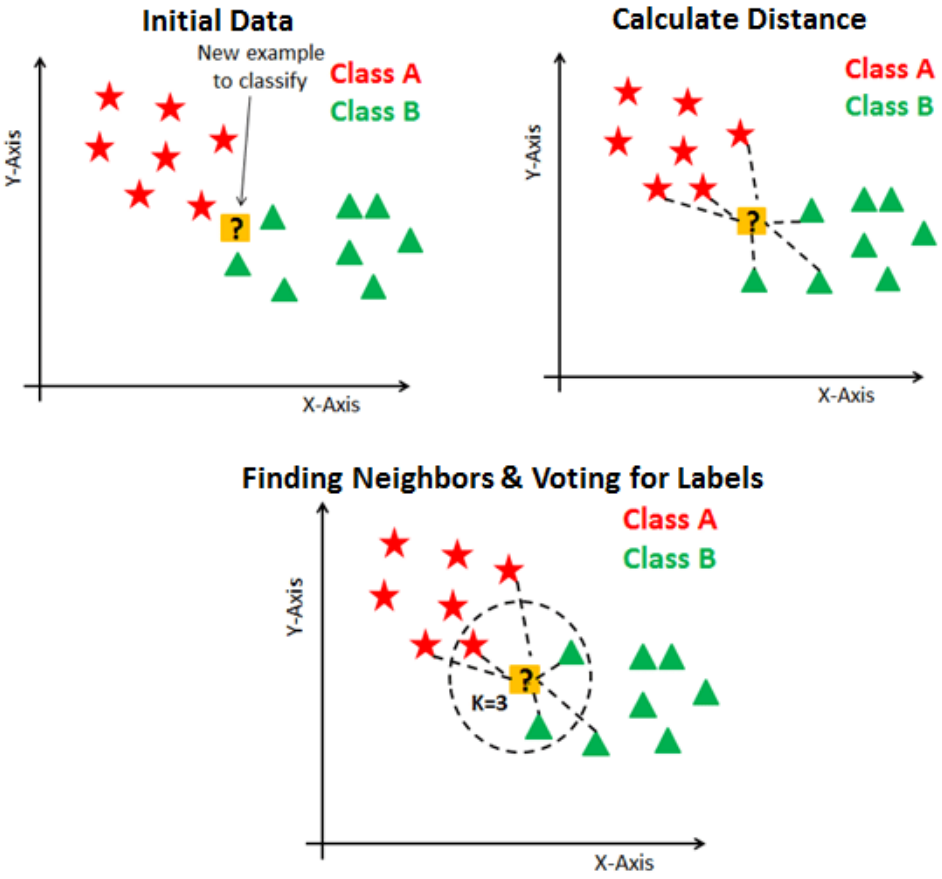
# of variables	max_depth	trn	tst	oot
20	7	0.735299851	0.707724148	0.433519553
20	8	0.757940024	0.691665472	0.417877095
20	9	0.775707476	0.714460877	0.408379888
21	7	0.739434657	0.677505745	0.434078212
21	8	0.766640013	0.702676153	0.45027933
21	9	0.782074554	0.719106484	0.435195531
24	7	0.750929972	0.687808269	0.466480447
24	8	0.756416425	0.685091601	0.400558659
24	9	0.774962625	0.686908169	0.381564246
25	7	0.742357462	0.688984698	0.425139665
25	8	0.753050307	0.671226784	0.373743017
25	9	0.773702291	0.687216337	0.393296089
26	7	0.743259898	0.68019461	0.425139665
26	8	0.756585764	0.681238704	0.402793296
26	9	0.775405329	0.696855551	0.360893855

In this project, we took the average of 10 runs to calculate the train test and oot value of FDR to reduce the impact of sample size. Given the variables provided in the above steps, the highest performance was achieved when we used 24 variables and a max_depth of 7. The oot FDR at 3% reached 46.65%. It also has been observed that the number of variables in the targeted range between 20 and 30 has much less of an impact than the choice of max_depth. At max_depth 7-9, the model consistently yielded the best performance of OOT above 40%, whereby the addition of more variables from the same wrapper does not always give improved results. The rationalization could be that in terms of decision tree model building, choosing an inadequate or too high of a

depth has a much stronger effect on the end results than the incremental inclusion and removal of variables.

K-Nearest Neighbors Classification

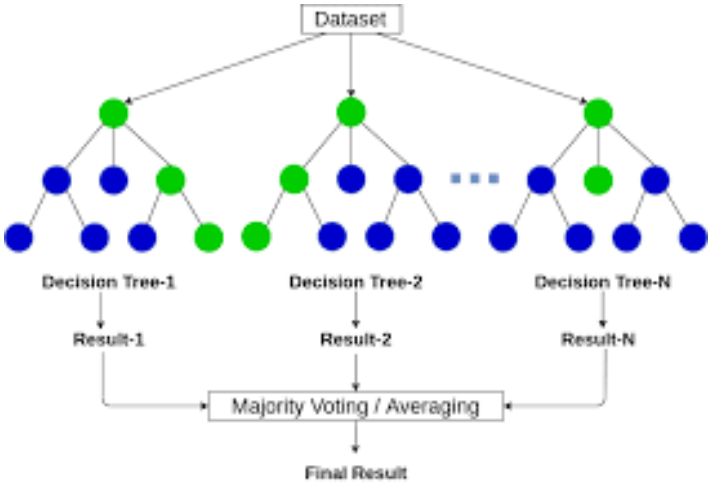
The KNN algorithm is a supervised machine learning technique which assumes that similar things exist in proximity. K-NN algorithm stores all the available data and classifies a new data point based on the similarity. When a new data point appears, the model searches the k closest point to the new point and then classifies points by majority vote of its k neighbors. Each object votes for their class and the class with the most votes is taken as the prediction.



With the fraud label being the dependent variable and the 20 selected features being the independent variables, we trained KNN models with parameters k=5, 10, 15, 20, 25. We then used the model to predict the probability of fraud on the train, test and the oot dataset separately and get the best average FDR rate of 0.871, 0.816 and 0.514 for each of the dataset.

Model	Hyperparameter		Average FDR(%) at 3%		
KNN Classification	# of variables selected	# of neighbors	Train	Test	OOT
1	20	5	1.0000	0.7966	0.4330
2	20	10	0.9384	0.7872	0.4911
3	20	15	0.9006	0.8078	0.4955
4	20	20	0.8835	0.7980	0.5134
5	20	25	0.8712	0.8161	0.5140

Random Forest



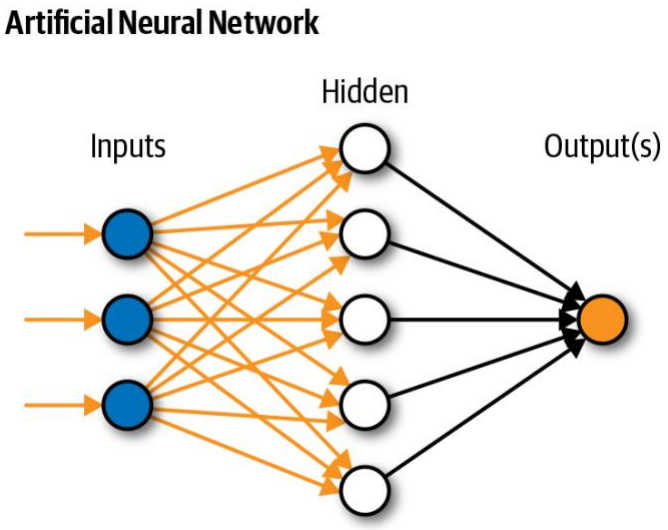
Random forest is an integrated learning method based on classification and regression. It constructs a large number of decision trees during training and outputs the pattern of classes (classification) or average/average prediction of a single tree (regression). We used the

randomForest package in the Python sklearn package. When we trained the random forest model, we tried generating 50 to 100 decision trees with max_depth of 5, 6, and 10. The best result for the model is n_estimators equals 90, max_depth equals 10 and the result of train FDR is 0.88, FDR for test is 0.826. FDR for OOT is 0.607

Random Forest	Variables	max_features	n_estimators	max_depth	Train	Test	OOT
1	20	7	50	5	0.765901	0.750501	0.54581
2	20	7	60	5	0.761212	0.740436	0.540223
3	20	7	70	5	0.77323	0.735393	0.551955
4	20	7	80	5	0.774602	0.746239	0.55419
5	20	7	90	5	0.776083	0.742632	0.543017
6	20	7	100	5	0.766295	0.762917	0.548045
7	20	7	50	6	0.784133	0.754692	0.561453
8	20	7	60	6	0.793667	0.754174	0.569274
9	20	7	70	6	0.791662	0.759538	0.578212
10	20	7	80	6	0.796532	0.752971	0.568715
11	20	7	90	6	0.793572	0.7514	0.575419
12	20	7	100	6	0.787293	0.786178	0.579888
13	20	7	50	10	0.883028	0.805392	0.600559
14	20	7	60	10	0.886988	0.808326	0.605587
15	20	7	70	10	0.88376	0.819331	0.597765
16	20	7	80	10	0.878466	0.80716	0.597765
17	20	7	90	10	0.883583	0.825568	0.607263
18	20	7	100	10	0.886215	0.816744	0.606145
19	20	7	100	15	0.981077	0.839859	0.614525

Neural Network

Artificial neural networks are statistical models directly inspired by, and partially modeled on biological neural networks. For example, a brain neuron receives an input and based on that input, fires off an output that is used by another neuron. The neural network simulates this behavior in learning about collecting the data and predicting outcomes.

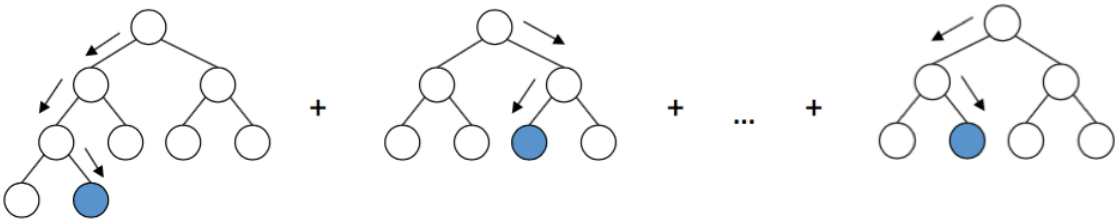


In the above graph, we can see that a typical neural network model consists of an input layer, hidden layers, and an output layer. The input layer is formed by all the independent variables. Each hidden layer is a set of nodes (neurons). Each neuron in the hidden layer receives weighted signals from all the nodes in the previous and transforms the linear combination of signals. The transform/activation function can be a logistic function(sigmoid) or something else. Finally, the output layer is the dependent variable.

We trained 16 neural network models with different hyperparameters as in the following table. For each hyperparameter choice, we ran 10 times with different train/test splits and listed the average FDRs at 3% on training, testing, and oot data. According to each model's performance, we didn't observe overfitting so we could trust the results of these models. The best model is the highlighted model no.6 with average FDR of 0.603 on OOT data.

Model	Hyperparameter						Average FDR at 3%		
Neural Network	# of variables selected	Layers	Nodes	Epochs	Learning Rate	Batch Size	Train	Test	OOT
1	20	1	10	10	0.01	480	0.760203	0.751065	0.577095
2	20	1	10	10	0.01	1000	0.764284	0.756082	0.567039
3	20	1	10	10	0.01	100	0.776275	0.760164	0.57933
4	20	1	20	10	0.01	480	0.785399	0.763673	0.585475
5	20	1	20	20	0.01	480	0.803592	0.761826	0.563128
6	20	1	30	10	0.01	480	0.786203	0.771915	0.602793
7	20	1	30	20	0.01	480	0.803158	0.780053	0.563128
8	20	1	30	30	0.01	480	0.821322	0.777653	0.549721
9	20	1	10	30	0.01	480	0.794559	0.766155	0.551955
10	20	1	40	100	0.01	1000	0.858016	0.804812	0.54581
11	20	1	40	10	0.01	480	0.789634	0.763531	0.601117
12	20	1	40	20	0.01	480	0.803234	0.787195	0.573184
13	20	1	40	100	0.01	480	0.86153	0.799362	0.524581
14	30	1	30	10	0.01	480	0.779426	0.765228	0.588827
15	30	1	40	10	0.01	480	0.78463	0.772869	0.600559
16	30	1	40	20	0.01	480	0.805561	0.782066	0.567598

Gradient Boosting Tree



Gradient Boosting Tree is an extremely popular machine learning algorithm that has been proven to be successful across many domains. Whereas random forest builds an ensemble of deep independent trees, Gradient Boosting Tree builds an ensemble of shallow trees one by one with each tree learning and improving on the previous one. Even though shallow trees are weak predictive models by themselves, they can be "boosted" together to produce a powerful predictive model when appropriately tuned. In this project, we used the Gradient Boosting Classifier from Scikit-learn package to predict the fraud transaction.

In total, we trained 12 boosted tree models with different hyperparameters as in the following table and listed all average FDR at 3% across 10 times for each hyperparameter choice. According to each model performance, we didn't observe overfitting problems so we could trust the results across 10 runs. The best model is the highlighted Boosted Tree 1 with average FDR of 54.5% on out of time data.

Model	Hyperparameter				Average FDR(%)		
Boosted tree	num of vars	learning_rate	n_estimators	max_depth	Train	Test	OOT
1	20	0.06	60	7	0.948409	0.842267	0.545251
2	20	0.06	60	12	1	0.828858	0.49162
3	20	0.06	120	7	0.986052	0.841965	0.552514
4	20	0.06	120	12	1	0.853939	0.479888
5	20	0.06	170	7	0.997023	0.868238	0.553073
6	20	0.06	170	12	1	0.842706	0.494413
7	20	0.25	60	7	0.995906	0.806665	0.463128
8	20	0.25	60	12	0.998844	0.823272	0.488827
9	20	0.25	120	7	0.98305	0.791115	0.449162
10	20	0.25	120	12	0.999512	0.855785	0.486592
11	20	0.25	170	7	0.977686	0.782049	0.430168
12	20	0.25	170	12	0.999184	0.865377	0.506704

Results

By training several models using different machine learning algorithms and adjusting hyperparameters for each particular machine learning model, we compared the model performance based on average FDR at 3% on out of time data and found that the best model is the Random Forest as highlighted in the following table with the Average FDR at 3% of 60.73%.

Model	Average FDR at 3%(%)		
	Train	Test	OOT
Logistic Regression	0.685718	0.683575	0.304469
Decision Tree	0.75093	0.687808	0.46648
KNN	0.8712	0.8161	0.514
Random Forest	0.883583	0.825568	0.607263
Neural Net	0.786203	0.771915	0.602793
Gradient Boosting Tree	0.948409	0.842267	0.545251

Then, we retrained the best Random Forest on the train data and reevaluated the model on both test and oot data. We took a closer look at the structure of the predicted results on all three datasets by computing critical statistics by individual percentiles and then generated the tables below to examine how our final model performed on the three datasets when detecting fraudulent transactions.

According to the following table reports, we could see how many frauds we can catch at each population bin and the cumulative frauds at each population bin by our model.
In the first percentile of transactions, all three sets yielded the highest proportion of true frauds, 79.75% in training, 65.14% in testing, and 37.43% in oot. Subsequent percentiles quickly diminish in marginal gains but cumulatively reaches 87.72% for training, 83.40% for testing, and 61.45% for oot by the 3rd percentile of data.

What also catches attention is the fact that the underlying fraud rates in the three data sets are not quite so equal, this could be attributed to the small sample size of the response variable. As a result, it is noticeable how the number of bads caught beyond the first 3% in the OOT set falls to single digits and scatters between 0 and 5. A smoother transition from large to small numbers could be expected if the data was more propagated. On the other hand, a conclusion could be strived for that the algorithm effectively clusters most fraud records to the top of the tables. The percentage of Bads caught in the OOT set at 61.45% means that well over half of all frauds are flagged by this model at the price of mere 3% of all transactions, even on data never fed into it. The implication of this insight is significant and will be further discussed in the following section.

Model Performance on Training

Train	#Records	#Goods	#Bads	Fraud Rate								
	56442	55815	627	1.11%								
	Bin Statistics					Cumulative Statistics						
Population Bin %	# Records	# Goods	# Bads	% Goods	% Bads	Total # Records	Cumulative Goods	Cumulative Bads	% Goods	% Bads (FDR	KS	FPR
1	564	64	500	11.34752	88.65248	564	64	500	0.114665	79.744817	79.630152	0.128
2	565	530	35	93.80531	6.19469	1129	594	535	1.06423	85.326954	84.262724	1.1102804
3	564	549	15	97.34043	2.659574	1693	1143	550	2.047837	87.719298	85.671462	2.0781818
4	565	555	10	98.23009	1.769912	2258	1698	560	3.042193	89.314195	86.272002	3.0321429
5	564	548	16	97.16312	2.836879	2822	2246	576	4.024008	91.866029	87.842021	3.8993056
6	565	564	1	99.82301	0.176991	3387	2810	577	5.034489	92.025518	86.991029	4.8700173
7	564	559	5	99.11348	0.886525	3951	3369	582	6.036012	92.822967	86.786955	5.7886598
8	564	560	4	99.29078	0.70922	4515	3929	586	7.039326	93.460925	86.421599	6.7047782
9	565	563	2	99.64602	0.353982	5080	4492	588	8.048016	93.779904	85.731889	7.6394558
10	564	563	1	99.8227	0.177305	5644	5055	589	9.056705	93.939394	84.882689	8.582343
11	565	555	10	98.23009	1.769912	6209	5610	599	10.05106	95.53429	85.483229	9.3656093
12	564	562	2	99.64539	0.35461	6773	6172	601	11.05796	95.85327	84.79531	10.269551
13	564	564	0	100	0	7337	6736	601	12.06844	95.85327	83.784829	11.207987
14	565	562	3	99.46903	0.530973	7902	7298	604	13.07534	96.331738	83.2564	12.082781
15	564	561	3	99.46809	0.531915	8466	7859	607	14.08044	96.810207	82.729763	12.947282
16	565	565	0	100	0	9031	8424	607	15.09272	96.810207	81.71749	13.878089
17	564	563	1	99.8227	0.177305	9595	8987	608	16.10141	96.969697	80.868291	14.78125
18	565	565	0	100	0	10160	9552	608	17.11368	96.969697	79.856018	15.710526
19	564	564	0	100	0	10724	10116	608	18.12416	96.969697	78.845537	16.638158
20	564	564	0	100	0	11288	10680	608	19.13464	96.969697	77.835056	17.565789

Model Performance on Testing

Test	#Records	#Goods	#Bads	Fraud Rate								
	24190	23949	241	1.00%								
	Bin Statistics					Cumulative Statistics						
Population Bin %	# Records	# Goods	# Bads	% Goods	% Bads	Total # Records	Cumulative Goods	Cumulative Bads	% Goods	% Bads (FDR)	KS	FPR
1	242	85	157	35.12397	64.87603	242	85	157	0.354921	65.14522822	64.79031	0.541401
2	242	210	32	86.77686	13.22314	484	295	189	1.231784	78.42323651	77.19145	1.560847
3	242	230	12	95.04132	4.958678	726	525	201	2.192158	83.40248963	81.21033	2.61194
4	242	237	5	97.93388	2.066116	968	762	206	3.181761	85.47717842	82.29542	3.699029
5	242	235	7	97.10744	2.892562	1210	997	213	4.163013	88.38174274	84.21873	4.680751
6	241	235	6	97.51037	2.489627	1451	1232	219	5.144265	90.87136929	85.7271	5.625571
7	242	238	4	98.34711	1.652893	1693	1470	223	6.138043	92.53112033	86.39308	6.591928
8	242	241	1	99.58678	0.413223	1935	1711	224	7.144348	92.94605809	85.80171	7.638393
9	242	241	1	99.58678	0.413223	2177	1952	225	8.150653	93.36099585	85.21034	8.675556
10	242	242	0	100	0	2419	2194	225	9.161134	93.36099585	84.19986	9.751111
11	242	240	2	99.17355	0.826446	2661	2434	227	10.16326	94.19087137	84.02761	10.72247
12	242	241	1	99.58678	0.413223	2903	2675	228	11.16957	94.60580913	83.43624	11.73246
13	242	242	0	100	0	3145	2917	228	12.18005	94.60580913	82.42576	12.79386
14	242	240	2	99.17355	0.826446	3387	3157	230	13.18218	95.43568465	82.25351	13.72609
15	241	240	1	99.58506	0.414938	3628	3397	231	14.18431	95.85062241	81.66631	14.70563
16	242	242	0	100	0	3870	3639	231	15.19479	95.85062241	80.65583	15.75325
17	242	241	1	99.58678	0.413223	4112	3880	232	16.20109	96.26556017	80.06447	16.72414
18	242	242	0	100	0	4354	4122	232	17.21157	96.26556017	79.05399	17.76724
19	242	242	0	100	0	4596	4364	232	18.22206	96.26556017	78.0435	18.81034
20	242	241	1	99.58678	0.413223	4838	4605	233	19.22836	96.68049793	77.45214	19.76395

Model Performance on OOT

OOT	#Records	#Goods	#Bads	Fraud Rate								
	12427	12248	179	1.44%								
	Bin Statistics					Cumulative Statistics						
Population Bin %	# Records	# Goods	# Bads	% Goods	% Bads	Total # Records	Cumulative Goods	Cumulative Bads	% Goods	% Bads (FDR)	KS	FPR
1	124	57	67	45.96774	54.03226	124	57	67	0.465382103	37.4301676	36.96478549	0.850746269
2	125	92	33	73.6	26.4	249	149	100	1.216525147	55.86592179	54.64939664	1.49
3	124	114	10	91.93548	8.064516	373	263	110	2.147289353	61.45251397	59.30522461	2.390909091
4	124	120	4	96.77419	3.225806	497	383	114	3.12704115	63.68715084	60.56010969	3.359649123
5	124	121	3	97.58065	2.419355	621	504	117	4.114957544	65.36312849	61.24817095	4.307692308
6	125	121	4	96.8	3.2	746	625	121	5.102873939	67.59776536	62.49489142	5.165289256
7	124	120	4	96.77419	3.225806	870	745	125	6.082625735	69.83240223	63.7497765	5.96
8	124	119	5	95.96774	4.032258	994	864	130	7.054212933	72.62569832	65.57148539	6.646153846
9	124	123	1	99.19355	0.806452	1118	987	131	8.058458524	73.18435754	65.12589902	7.534351145
10	125	123	2	98.4	1.6	1243	1110	133	9.062704115	74.30167598	65.23897186	8.345864662
11	124	124	0	100	0	1367	1234	133	10.0751143	74.30167598	64.22656167	9.278195489
12	124	123	1	99.19355	0.806452	1491	1357	134	11.0793599	74.8603352	63.7809753	10.12686567
13	125	125	0	100	0	1616	1482	134	12.09993468	74.8603352	62.76040051	11.05970149
14	124	123	1	99.19355	0.806452	1740	1605	135	13.10418027	75.41899441	62.31481414	11.88888889
15	124	120	4	96.77419	3.225806	1864	1725	139	14.08393207	77.65363128	63.56969921	12.41007194
16	124	124	0	100	0	1988	1849	139	15.09634226	77.65363128	62.55728902	13.30215827
17	125	123	2	98.4	1.6	2113	1972	141	16.10058785	78.77094972	62.67036187	13.9858156
18	124	122	2	98.3871	1.612903	2237	2094	143	17.09666884	79.88826816	62.79159931	14.64335664
19	124	123	1	99.19355	0.806452	2361	2217	144	18.10091444	80.44692737	62.34601294	15.39583333
20	124	124	0	100	0	2485	2341	144	19.11332462	80.44692737	61.33360275	16.25694444

Business Insights

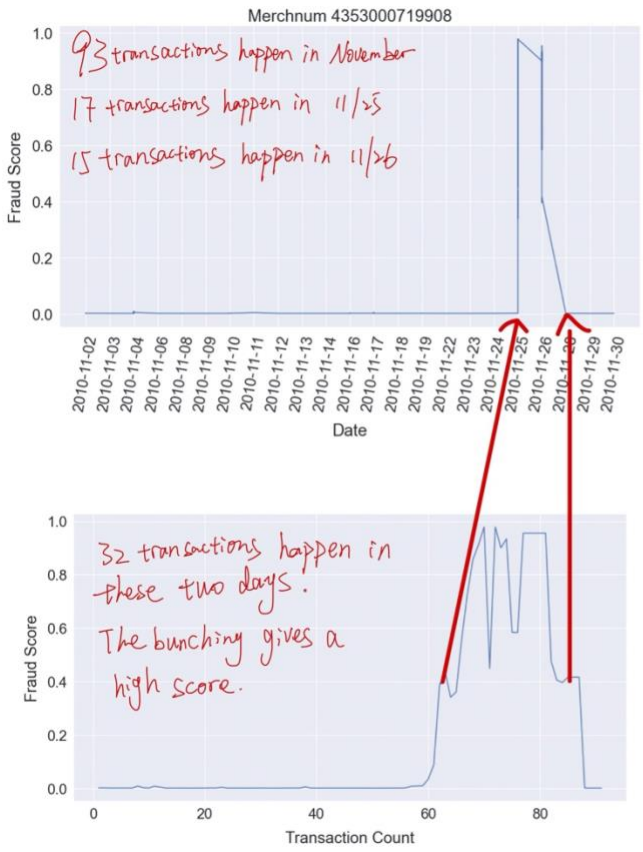
Fraud saving

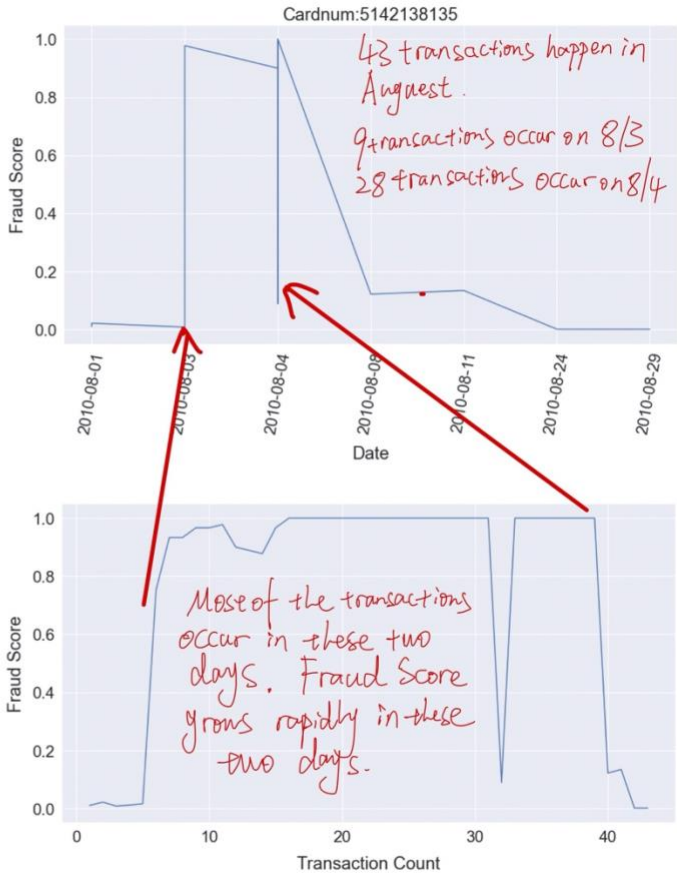
The implementation of the fraud model is expected to significantly reduce losses associated with frauds. In this specific problem, each fraud (True Positive) stopped is considered to bring an expected gain of $2,000 and each falsely identified transaction (False Positive) causes an expected loss of $50. It is with these assumptions that the following graph was drawn for 2 months of OOT data.

As the population bin progresses to the right, the lost sales proportionally increase at $50 each, while the Fraud Savings jumps initially but soon plateaus after a certain cut off. The difference between the two lines is then marked by the green curve which reaches a maximum around 2% total population bins at annual savings of $1.16 millions. This means that the card issuing institution will deny problematic transactions up to the top 2% of the population bin. After 2% population bin, we see a flat trend and then a downward trend of overall savings. These numbers are expected to boost should the model see any improvement, and the cut-off point is subject to change.

Fraud score

According to the following graphs, we can infer from that the burst of transactions in the short periods will boost up the fraud score for that entity.





Conclusion

In this project, we have covered all the steps including data preprocessing, feature creation, selection, and model building. For data preprocessing, we corrected as many records with unmatched state and zip as we can and filled the missing values by grouping on several different and meaningful entities. And then, we created hundreds of candidate variables related to fraud signals. By using filter and wrapper method, we selected 20 to 30 variables based on specific scoring criteria. We then applied different machine learning algorithms from baseline model to complex nonlinear models and selected final model based on average FDR at 3% across 10 runs. By having the best Random Forest as our final model, we generated bin statistics, fraud score analysis, and fraud saving analysis to provide business recommendation. we were able to capture a decent amount of the irregular activities via the machine learning methods, specifically, up to 61.45% of all frauds in the top 3 percent of all
transactions. This result is particularly encouraging considering there are efforts outside the scope of this project that can be made to further boost the percentage. Given the small sample size of the targeted variable, more efforts could be devoted into the sampling techniques designed to handle the imbalance of the classes. Secondly, we could consult with domain experts for insights to device more tailormade and thereby effective features to better represent the fraud activities. Lastly, in a real-world project we would expect the data set to be larger and more realistic pattern to be available wherein a simulated dataset is not found, so that variables involving the geo-coding for example, which captures the distance across different zip codes to play a more significant role in feature engineering.

At the same time, we realize that there are limitations to the application of these techniques. For example, not 100% of all fraud cases are observed by the consumers, and thereby reported to the associated entities. This invariably hinders our ability to depict a full picture of all fraudulent transactions. Financial institutions are known for not willing to hand over their user data, which is yet another barrier for data accessibility and subsequent maintenance of models and collection of feedback.

Despite these challenges, the lessons we learned in the class are invaluable in today's market. That which is applicable in the fraud detection problem could also be applied in a targeted marketing campaign. The difference lies in the domain knowledge, which is a business and communication

problem. The technical side is much the same, from data cleaning to feature engineering, from picking a wrapper to evaluating overfitting in the final model selection, and lastly from the calculation and dissection of key statistics to the eternally true demand for a more powerful computer in the field of machine learning.

Appendix

Data Quality Report

File Description

The Dataset, named as “card transactions data.csv”, is a data set of 12-month actual card purchases from a US government organization. The purpose of the dataset is to find card transaction fraud. The dataset contains 96,753 records. There are 10 fields, one numerical field, 1 date filed and 8 categorical fields.

Summary Table

Numeric Field:

Column Name	# of Records	% populated	Unique Values	Mean	Standard Deviation	Minimum Value	Maximum Value	# Zeros
Amount	96753	100	34909	427.8857	10006.14	0.01	3102046	0

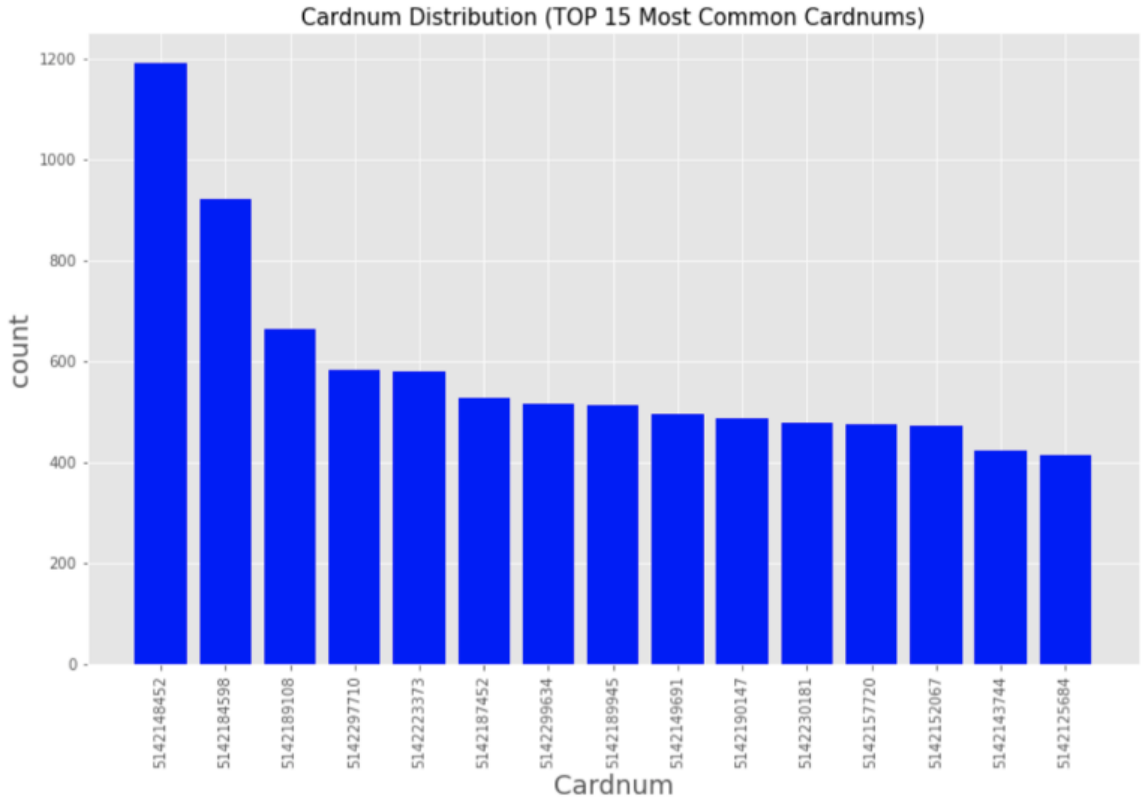
Categorical Fields:

Column Name	# of Records	% populated	Unique Values	Most Common Field Value
Recnum	96753	100	96753	2047
Cardnum	96753	100	1645	5142148452
Date	96753	100	365	2/28/10
Merchnum	93378	96.51	13091	930090121224
Merch description	96753	100	13126	GSA-FSS-ADV
Merch state	95558	98.76	227	TN
Merch zip	92097	95.19	4567	38118
Transtype	96753	100	4	P
Amount	96753	100	34909	3.62

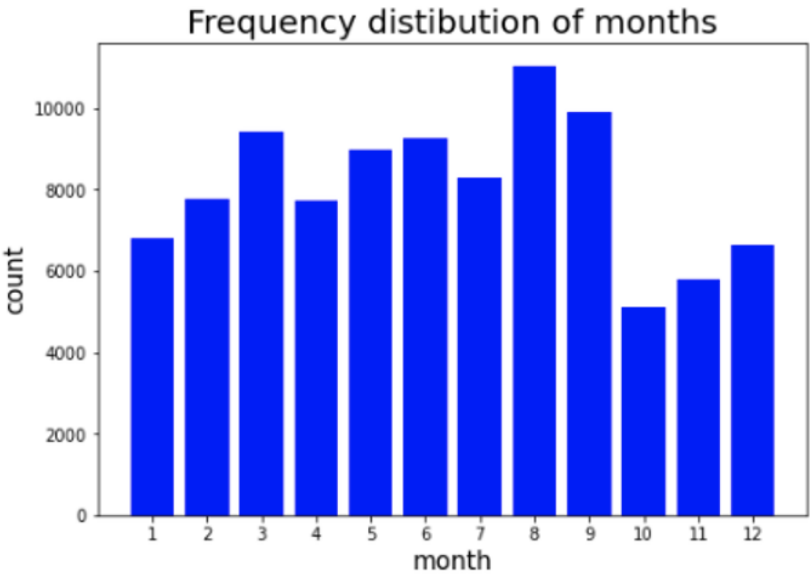
Fraud	96753	100	2	0

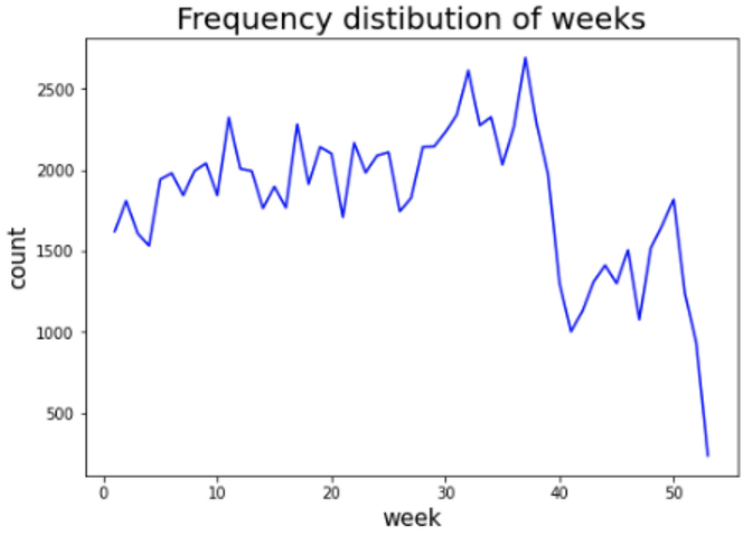
Field 1
Name: Recnum
Description: Unique identifier of each card transaction

Field 2
Name: CardNum
Description: Card Number
Data in the histogram shows top 15 most common card numbers

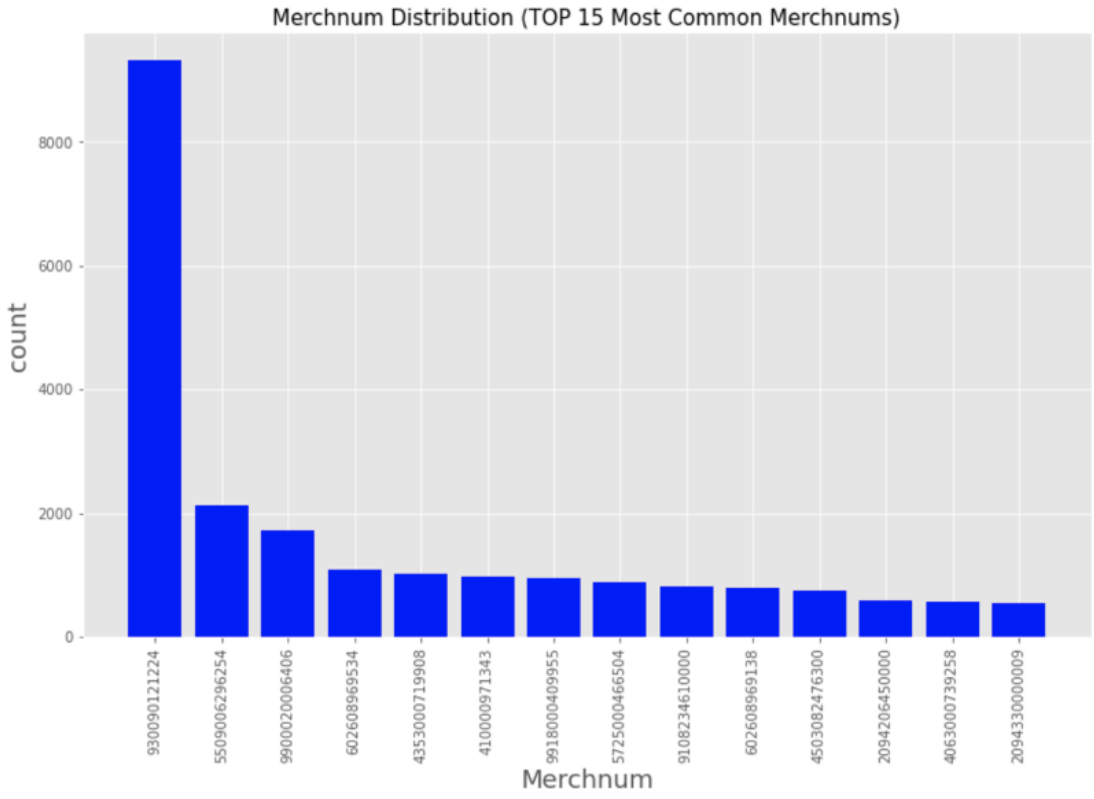


Field 3
Name: Date
Description: the date on which the card transaction occurred
Data in the histogram shows the distribution of January

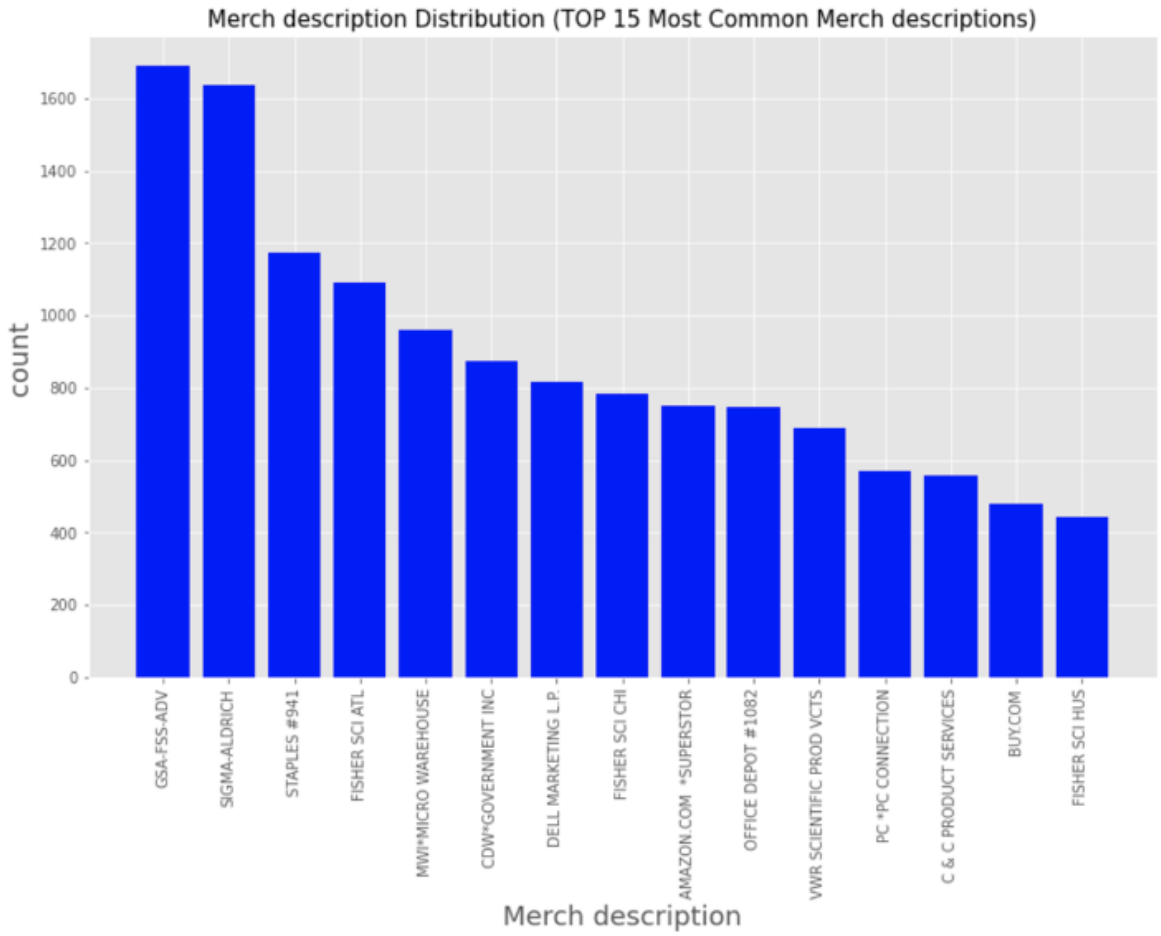




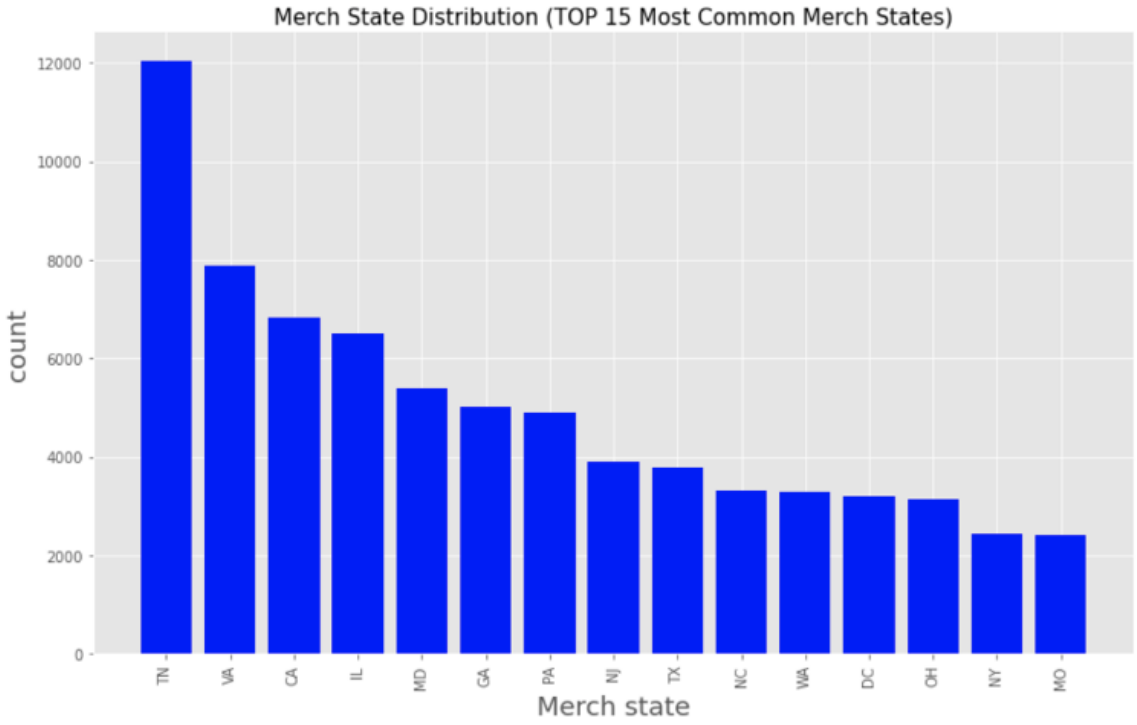
Field 4
Name: Merchnum
Description: The identifier of the merchandise
Data in the histogram shows top 15 most common merchant number



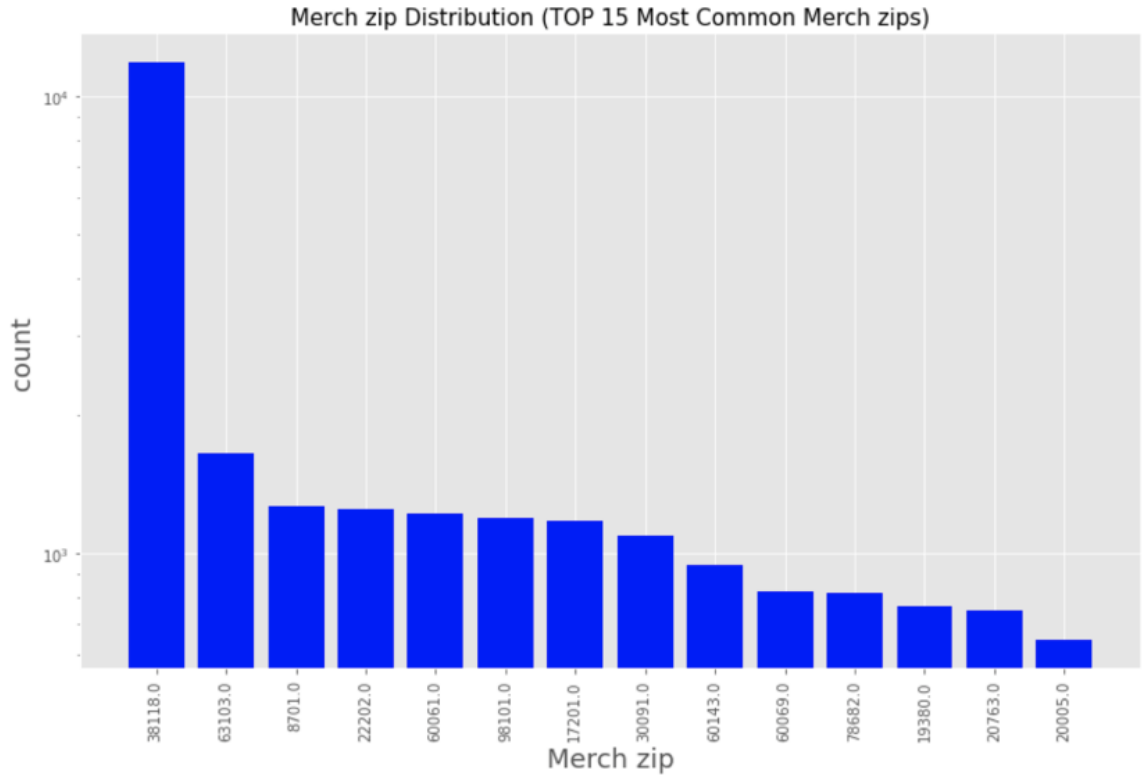
Field 5
Name: Merch description
Description: Description of the merchant
Data in the histogram shows top 15 most common merchant descriptions



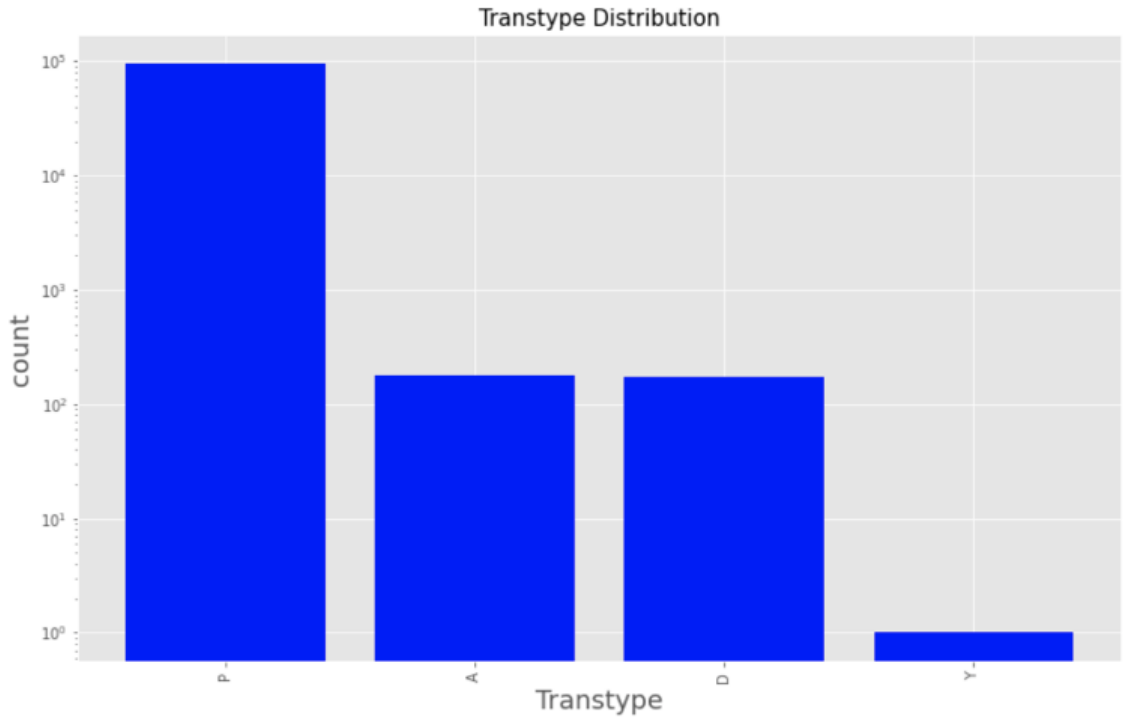
Field 6
Name: Merch state
Description: The location of the merchant branch where the purchase is made
Data in the histogram shows top 15 most common merchant locations



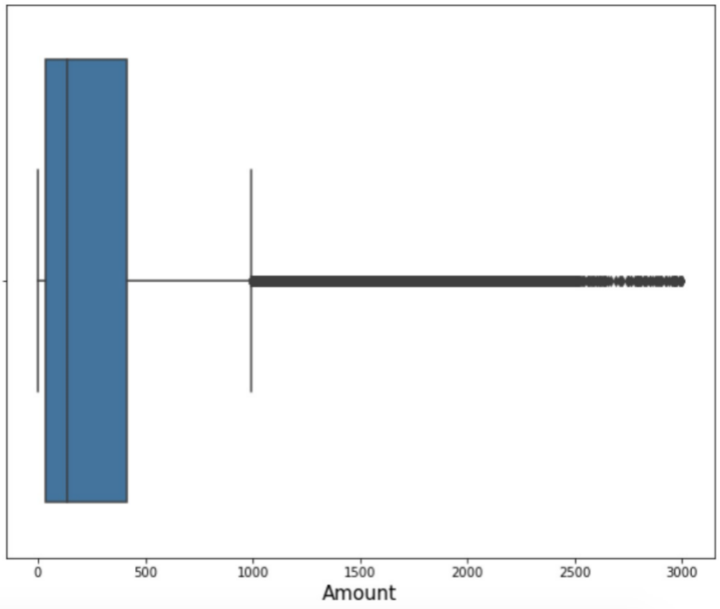
Field 7
Name: zip
Description: the zip code of the merchant address
Data in the histogram shows top 15 most common merchant zips

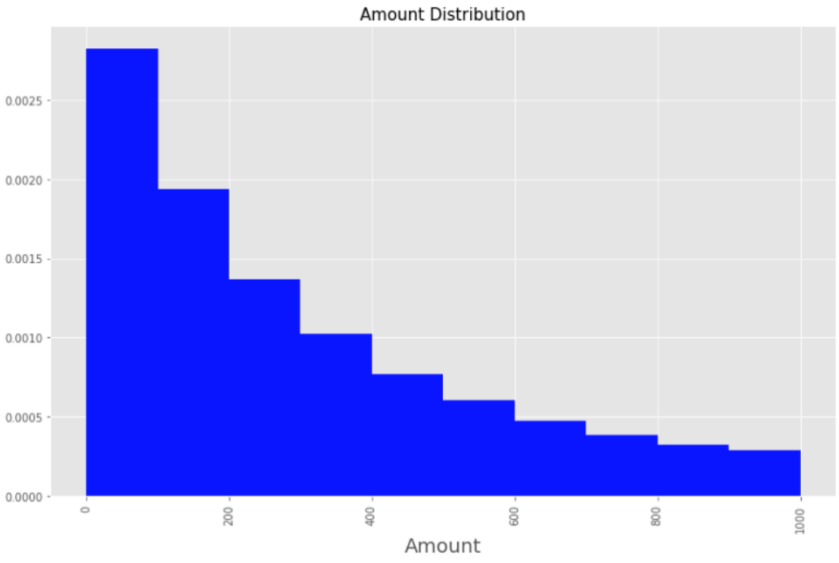


Field 8
Name: Transtype
Description: Transaction type



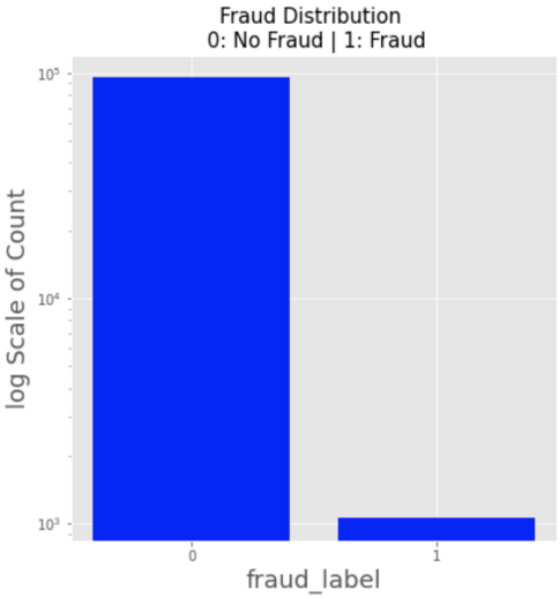
Field 9
Name: Amount
Description: The dollar amount of the transaction
The boxplot shows the overall distribution.
The histogram shows transaction amounts of less than 1000.





Filed 10
Name: Fraud
Description: whether the card transaction is fraud
0: No Fraud | 1: Fraud

	Fraud	count	percentage
0	0	95694	0.989055
1	1	1059	0.010945



Candidate Variables

1	cardnum_day_since	337	merchnum_zip_actual/avg_0
2	cardnum_count_0	338	merchnum_zip_actual/max_0
3	cardnum_avg_0	339	merchnum_zip_actual/med_0
4	cardnum_max_0	340	merchnum_zip_actual/toal_0
5	cardnum_med_0	341	merchnum_zip_count_1
6	cardnum_total_0	342	merchnum_zip_avg_1
7	cardnum_actual/avg_0	343	merchnum_zip_max_1
8	cardnum_actual/max_0	344	merchnum_zip_med_1
9	cardnum_actual/med_0	345	merchnum_zip_total_1
10	cardnum_actual/toal_0	346	merchnum_zip_actual/avg_1
11	cardnum_count_1	347	merchnum_zip_actual/max_1
12	cardnum_avg_1	348	merchnum_zip_actual/med_1
13	cardnum_max_1	349	merchnum_zip_actual/toal_1
14	cardnum_med_1	350	merchnum_zip_count_3
15	cardnum_total_1	351	merchnum_zip_avg_3
16	cardnum_actual/avg_1	352	merchnum_zip_max_3

17	cardnum_actual/max_1	353	merchnum_zip_med_3
18	cardnum_actual/med_1	354	merchnum_zip_total_3
19	cardnum_actual/toal_1	355	merchnum_zip_actual/avg_3
20	cardnum_count_3	356	merchnum_zip_actual/max_3
21	cardnum_avg_3	357	merchnum_zip_actual/med_3
22	cardnum_max_3	358	merchnum_zip_actual/toal_3
23	cardnum_med_3	359	merchnum_zip_count_7
24	cardnum_total_3	360	merchnum_zip_avg_7
25	cardnum_actual/avg_3	361	merchnum_zip_max_7
26	cardnum_actual/max_3	362	merchnum_zip_med_7
27	cardnum_actual/med_3	363	merchnum_zip_total_7
28	cardnum_actual/toal_3	364	merchnum_zip_actual/avg_7
29	cardnum_count_7	365	merchnum_zip_actual/max_7
30	cardnum_avg_7	366	merchnum_zip_actual/med_7
31	cardnum_max_7	367	merchnum_zip_actual/toal_7
32	cardnum_med_7	368	merchnum_zip_count_14
33	cardnum_total_7	369	merchnum_zip_avg_14
34	cardnum_actual/avg_7	370	merchnum_zip_max_14

35	cardnum_actual/max_7	371	merchnum_zip_med_14
36	cardnum_actual/med_7	372	merchnum_zip_total_14
37	cardnum_actual/toal_7	373	merchnum_zip_actual/avg_14
38	cardnum_count_14	374	merchnum_zip_actual/max_14
39	cardnum_avg_14	375	merchnum_zip_actual/med_14
40	cardnum_max_14	376	merchnum_zip_actual/toal_14
41	cardnum_med_14	377	merchnum_zip_count_30
42	cardnum_total_14	378	merchnum_zip_avg_30
43	cardnum_actual/avg_14	379	merchnum_zip_max_30
44	cardnum_actual/max_14	380	merchnum_zip_med_30
45	cardnum_actual/med_14	381	merchnum_zip_total_30
46	cardnum_actual/toal_14	382	merchnum_zip_actual/avg_30
47	cardnum_count_30	383	merchnum_zip_actual/max_30
48	cardnum_avg_30	384	merchnum_zip_actual/med_30
49	cardnum_max_30	385	merchnum_zip_actual/toal_30
50	cardnum_med_30	386	merchnum_state_day_since
51	cardnum_total_30	387	merchnum_state_count_0
52	cardnum_actual/avg_30	388	merchnum_state_avg_0

53	cardnum_actual/max_30	389	merchnum_state_max_0
54	cardnum_actual/med_30	390	merchnum_state_med_0
55	cardnum_actual/toal_30	391	merchnum_state_total_0
56	merchnum_day_since	392	merchnum_state_actual/avg_0
57	merchnum_count_0	393	merchnum_state_actual/max_0
58	merchnum_avg_0	394	merchnum_state_actual/med_0
59	merchnum_max_0	395	merchnum_state_actual/toal_0
60	merchnum_med_0	396	merchnum_state_count_1
61	merchnum_total_0	397	merchnum_state_avg_1
62	merchnum_actual/avg_0	398	merchnum_state_max_1
63	merchnum_actual/max_0	399	merchnum_state_med_1
64	merchnum_actual/med_0	400	merchnum_state_total_1
65	merchnum_actual/toal_0	401	merchnum_state_actual/avg_1
66	merchnum_count_1	402	merchnum_state_actual/max_1
67	merchnum_avg_1	403	merchnum_state_actual/med_1
68	merchnum_max_1	404	merchnum_state_actual/toal_1
69	merchnum_med_1	405	merchnum_state_count_3
70	merchnum_total_1	406	merchnum_state_avg_3

71 merchnum_actual/avg_1

72 merchnum_actual/max_1

73 merchnum_actual/med_1

74 merchnum_actual/toal_1

75 merchnum_count_3

76 merchnum_avg_3

77 merchnum_max_3

78 merchnum_med_3

79 merchnum_total_3

80 merchnum_actual/avg_3

81 merchnum_actual/max_3

82 merchnum_actual/med_3

83 merchnum_actual/toal_3

84 merchnum_count_7

85 merchnum_avg_7

86 merchnum_max_7

87 merchnum_med_7

88 merchnum_total_7

407 merchnum_state_max_3

408 merchnum_state_med_3

409 merchnum_state_total_3

410 merchnum_state_actual/avg_3

411 merchnum_state_actual/max_3

412 merchnum_state_actual/med_3

413 merchnum_state_actual/toal_3

414 merchnum_state_count_7

415 merchnum_state_avg_7

416 merchnum_state_max_7

417 merchnum_state_med_7

418 merchnum_state_total_7

419 merchnum_state_actual/avg_7

420 merchnum_state_actual/max_7

421 merchnum_state_actual/med_7

422 merchnum_state_actual/toal_7

423 merchnum_state_count_14

424 merchnum_state_avg_14

89 merchnum_actual/avg_7

90 merchnum_actual/max_7

91 merchnum_actual/med_7

92 merchnum_actual/toal_7

93 merchnum_count_14

94 merchnum_avg_14

95 merchnum_max_14

96 merchnum_med_14

97 merchnum_total_14

98 merchnum_actual/avg_14

99 merchnum_actual/max_14

100 merchnum_actual/med_14

101 merchnum_actual/toal_14

102 merchnum_count_30

103 merchnum_avg_30

104 merchnum_max_30

105 merchnum_med_30

106 merchnum_total_30

425 merchnum_state_max_14

426 merchnum_state_med_14

427 merchnum_state_total_14

428 merchnum_state_actual/avg_14

429 merchnum_state_actual/max_14

430 merchnum_state_actual/med_14

431 merchnum_state_actual/toal_14

432 merchnum_state_count_30

433 merchnum_state_avg_30

434 merchnum_state_max_30

435 merchnum_state_med_30

436 merchnum_state_total_30

437 merchnum_state_actual/avg_30

438 merchnum_state_actual/max_30

439 merchnum_state_actual/med_30

440 merchnum_state_actual/toal_30

441 cardnum_merchdes_day_since

442 cardnum_merchdes_count_0

107	merchnum_actual/avg_30	443	cardnum_merchdes_avg_0
108	merchnum_actual/max_30	444	cardnum_merchdes_max_0
109	merchnum_actual/med_30	445	cardnum_merchdes_med_0
110	merchnum_actual/toal_30	446	cardnum_merchdes_total_0
111	merch description_day_since	447	cardnum_merchdes_actual/avg_0
112	merch description_count_0	448	cardnum_merchdes_actual/max_0
113	merch description_avg_0	449	cardnum_merchdes_actual/med_0
114	merch description_max_0	450	cardnum_merchdes_actual/toal_0
115	merch description_med_0	451	cardnum_merchdes_count_1
116	merch description_total_0	452	cardnum_merchdes_avg_1
117	merch description_actual/avg_0	453	cardnum_merchdes_max_1
118	merch description_actual/max_0	454	cardnum_merchdes_med_1
119	merch description_actual/med_0	455	cardnum_merchdes_total_1
120	merch description_actual/toal_0	456	cardnum_merchdes_actual/avg_1
121	merch description_count_1	457	cardnum_merchdes_actual/max_1
122	merch description_avg_1	458	cardnum_merchdes_actual/med_1
123	merch description_max_1	459	cardnum_merchdes_actual/toal_1
124	merch description_med_1	460	cardnum_merchdes_count_3

125 merch description_total_1

126 merch description_actual/avg_1

127 merch description_actual/max_1

128 merch description_actual/med_1

129 merch description_actual/toal_1

130 merch description_count_3

131 merch description_avg_3

132 merch description_max_3

133 merch description_med_3

134 merch description_total_3

135 merch description_actual/avg_3

136 merch description_actual/max_3

137 merch description_actual/med_3

138 merch description_actual/toal_3

139 merch description_count_7

140 merch description_avg_7

141 merch description_max_7

142 merch description_med_7

461 cardnum_merchdes_avg_3

462 cardnum_merchdes_max_3

463 cardnum_merchdes_med_3

464 cardnum_merchdes_total_3

465 cardnum_merchdes_actual/avg_3

466 cardnum_merchdes_actual/max_3

467 cardnum_merchdes_actual/med_3

468 cardnum_merchdes_actual/toal_3

469 cardnum_merchdes_count_7

470 cardnum_merchdes_avg_7

471 cardnum_merchdes_max_7

472 cardnum_merchdes_med_7

473 cardnum_merchdes_total_7

474 cardnum_merchdes_actual/avg_7

475 cardnum_merchdes_actual/max_7

476 cardnum_merchdes_actual/med_7

477 cardnum_merchdes_actual/toal_7

478 cardnum_merchdes_count_14

143 merch description_total_7

144 merch description_actual/avg_7

145 merch description_actual/max_7

146 merch description_actual/med_7

147 merch description_actual/toal_7

148 merch description_count_14

149 merch description_avg_14

150 merch description_max_14

151 merch description_med_14

152 merch description_total_14

153 merch description_actual/avg_14

154 merch description_actual/max_14

155 merch description_actual/med_14

156 merch description_actual/toal_14

157 merch description_count_30

158 merch description_avg_30

159 merch description_max_30

160 merch description_med_30

479 cardnum_merchdes_avg_14

480 cardnum_merchdes_max_14

481 cardnum_merchdes_med_14

482 cardnum_merchdes_total_14

483 cardnum_merchdes_actual/avg_14

484 cardnum_merchdes_actual/max_14

485 cardnum_merchdes_actual/med_14

486 cardnum_merchdes_actual/toal_14

487 cardnum_merchdes_count_30

488 cardnum_merchdes_avg_30

489 cardnum_merchdes_max_30

490 cardnum_merchdes_med_30

491 cardnum_merchdes_total_30

492 cardnum_merchdes_actual/avg_30

493 cardnum_merchdes_actual/max_30

494 cardnum_merchdes_actual/med_30

495 cardnum_merchdes_actual/toal_30

496 merchdes_zip_day_since

161	merch description_total_30	497	merchdes_zip_count_0
162	merch description_actual/avg_30	498	merchdes_zip_avg_0
163	merch description_actual/max_30	499	merchdes_zip_max_0
164	merch description_actual/med_30	500	merchdes_zip_med_0
165	merch description_actual/toal_30	501	merchdes_zip_total_0
166	cardnum_merch_day_since	502	merchdes_zip_actual/avg_0
167	cardnum_merch_count_0	503	merchdes_zip_actual/max_0
168	cardnum_merch_avg_0	504	merchdes_zip_actual/med_0
169	cardnum_merch_max_0	505	merchdes_zip_actual/toal_0
170	cardnum_merch_med_0	506	merchdes_zip_count_1
171	cardnum_merch_total_0	507	merchdes_zip_avg_1
172	cardnum_merch_actual/avg_0	508	merchdes_zip_max_1
173	cardnum_merch_actual/max_0	509	merchdes_zip_med_1
174	cardnum_merch_actual/med_0	510	merchdes_zip_total_1
175	cardnum_merch_actual/toal_0	511	merchdes_zip_actual/avg_1
176	cardnum_merch_count_1	512	merchdes_zip_actual/max_1
177	cardnum_merch_avg_1	513	merchdes_zip_actual/med_1
178	cardnum_merch_max_1	514	merchdes_zip_actual/toal_1

179	cardnum_merch_med_1	515	merchdes_zip_count_3
180	cardnum_merch_total_1	516	merchdes_zip_avg_3
181	cardnum_merch_actual/avg_1	517	merchdes_zip_max_3
182	cardnum_merch_actual/max_1	518	merchdes_zip_med_3
183	cardnum_merch_actual/med_1	519	merchdes_zip_total_3
184	cardnum_merch_actual/toal_1	520	merchdes_zip_actual/avg_3
185	cardnum_merch_count_3	521	merchdes_zip_actual/max_3
186	cardnum_merch_avg_3	522	merchdes_zip_actual/med_3
187	cardnum_merch_max_3	523	merchdes_zip_actual/toal_3
188	cardnum_merch_med_3	524	merchdes_zip_count_7
189	cardnum_merch_total_3	525	merchdes_zip_avg_7
190	cardnum_merch_actual/avg_3	526	merchdes_zip_max_7
191	cardnum_merch_actual/max_3	527	merchdes_zip_med_7
192	cardnum_merch_actual/med_3	528	merchdes_zip_total_7
193	cardnum_merch_actual/toal_3	529	merchdes_zip_actual/avg_7
194	cardnum_merch_count_7	530	merchdes_zip_actual/max_7
195	cardnum_merch_avg_7	531	merchdes_zip_actual/med_7
196	cardnum_merch_max_7	532	merchdes_zip_actual/toal_7

197	cardnum_merch_med_7	533	merchdes_zip_count_14
198	cardnum_merch_total_7	534	merchdes_zip_avg_14
199	cardnum_merch_actual/avg_7	535	merchdes_zip_max_14
200	cardnum_merch_actual/max_7	536	merchdes_zip_med_14
201	cardnum_merch_actual/med_7	537	merchdes_zip_total_14
202	cardnum_merch_actual/toal_7	538	merchdes_zip_actual/avg_14
203	cardnum_merch_count_14	539	merchdes_zip_actual/max_14
204	cardnum_merch_avg_14	540	merchdes_zip_actual/med_14
205	cardnum_merch_max_14	541	merchdes_zip_actual/toal_14
206	cardnum_merch_med_14	542	merchdes_zip_count_30
207	cardnum_merch_total_14	543	merchdes_zip_avg_30
208	cardnum_merch_actual/avg_14	544	merchdes_zip_max_30
209	cardnum_merch_actual/max_14	545	merchdes_zip_med_30
210	cardnum_merch_actual/med_14	546	merchdes_zip_total_30
211	cardnum_merch_actual/toal_14	547	merchdes_zip_actual/avg_30
212	cardnum_merch_count_30	548	merchdes_zip_actual/max_30
213	cardnum_merch_avg_30	549	merchdes_zip_actual/med_30
214	cardnum_merch_max_30	550	merchdes_zip_actual/toal_30

215	cardnum_merch_med_30	551	merchdes_state_day_since
216	cardnum_merch_total_30	552	merchdes_state_count_0
217	cardnum_merch_actual/avg_30	553	merchdes_state_avg_0
218	cardnum_merch_actual/max_30	554	merchdes_state_max_0
219	cardnum_merch_actual/med_30	555	merchdes_state_med_0
220	cardnum_merch_actual/toal_30	556	merchdes_state_total_0
221	cardnum_zip_day_since	557	merchdes_state_actual/avg_0
222	cardnum_zip_count_0	558	merchdes_state_actual/max_0
223	cardnum_zip_avg_0	559	merchdes_state_actual/med_0
224	cardnum_zip_max_0	560	merchdes_state_actual/toal_0
225	cardnum_zip_med_0	561	merchdes_state_count_1
226	cardnum_zip_total_0	562	merchdes_state_avg_1
227	cardnum_zip_actual/avg_0	563	merchdes_state_max_1
228	cardnum_zip_actual/max_0	564	merchdes_state_med_1
229	cardnum_zip_actual/med_0	565	merchdes_state_total_1
230	cardnum_zip_actual/toal_0	566	merchdes_state_actual/avg_1
231	cardnum_zip_count_1	567	merchdes_state_actual/max_1
232	cardnum_zip_avg_1	568	merchdes_state_actual/med_1

233 cardnum_zip_max_1

234 cardnum_zip_med_1

235 cardnum_zip_total_1

236 cardnum_zip_actual/avg_1

237 cardnum_zip_actual/max_1

238 cardnum_zip_actual/med_1

239 cardnum_zip_actual/toal_1

240 cardnum_zip_count_3

241 cardnum_zip_avg_3

242 cardnum_zip_max_3

243 cardnum_zip_med_3

244 cardnum_zip_total_3

245 cardnum_zip_actual/avg_3

246 cardnum_zip_actual/max_3

247 cardnum_zip_actual/med_3

248 cardnum_zip_actual/toal_3

249 cardnum_zip_count_7

250 cardnum_zip_avg_7

569 merchdes_state_actual/toal_1

570 merchdes_state_count_3

571 merchdes_state_avg_3

572 merchdes_state_max_3

573 merchdes_state_med_3

574 merchdes_state_total_3

575 merchdes_state_actual/avg_3

576 merchdes_state_actual/max_3

577 merchdes_state_actual/med_3

578 merchdes_state_actual/toal_3

579 merchdes_state_count_7

580 merchdes_state_avg_7

581 merchdes_state_max_7

582 merchdes_state_med_7

583 merchdes_state_total_7

584 merchdes_state_actual/avg_7

585 merchdes_state_actual/max_7

586 merchdes_state_actual/med_7

251 cardnum_zip_max_7

252 cardnum_zip_med_7

253 cardnum_zip_total_7

254 cardnum_zip_actual/avg_7

255 cardnum_zip_actual/max_7

256 cardnum_zip_actual/med_7

257 cardnum_zip_actual/toal_7

258 cardnum_zip_count_14

259 cardnum_zip_avg_14

260 cardnum_zip_max_14

261 cardnum_zip_med_14

262 cardnum_zip_total_14

263 cardnum_zip_actual/avg_14

264 cardnum_zip_actual/max_14

265 cardnum_zip_actual/med_14

266 cardnum_zip_actual/toal_14

267 cardnum_zip_count_30

268 cardnum_zip_avg_30

587 merchdes_state_actual/toal_7

588 merchdes_state_count_14

589 merchdes_state_avg_14

590 merchdes_state_max_14

591 merchdes_state_med_14

592 merchdes_state_total_14

593 merchdes_state_actual/avg_14

594 merchdes_state_actual/max_14

595 merchdes_state_actual/med_14

596 merchdes_state_actual/toal_14

597 merchdes_state_count_30

598 merchdes_state_avg_30

599 merchdes_state_max_30

600 merchdes_state_med_30

601 merchdes_state_total_30

602 merchdes_state_actual/avg_30

603 merchdes_state_actual/max_30

604 merchdes_state_actual/med_30

269 cardnum_zip_max_30

270 cardnum_zip_med_30

271 cardnum_zip_total_30

272 cardnum_zip_actual/avg_30

273 cardnum_zip_actual/max_30

274 cardnum_zip_actual/med_30

275 cardnum_zip_actual/toal_30

276 cardnum_state_day_since

277 cardnum_state_count_0

278 cardnum_state_avg_0

279 cardnum_state_max_0

280 cardnum_state_med_0

281 cardnum_state_total_0

282 cardnum_state_actual/avg_0

283 cardnum_state_actual/max_0

284 cardnum_state_actual/med_0

285 cardnum_state_actual/toal_0

286 cardnum_state_count_1

605 merchdes_state_actual/toal_30

606 cardnum_count_0_by_7

607 cardnum_count_0_by_14

608 cardnum_count_0_by_30

609 cardnum_count_1_by_7

610 cardnum_count_1_by_14

611 cardnum_count_1_by_30

612 merchnum_count_0_by_7

613 merchnum_count_0_by_14

614 merchnum_count_0_by_30

615 merchnum_count_1_by_7

616 merchnum_count_1_by_14

617 merchnum_count_1_by_30

618 merch description_count_0_by_7

619 merch description_count_0_by_14

620 merch description_count_0_by_30

621 merch description_count_1_by_7

622 merch description_count_1_by_14

287	cardnum_state_avg_1	623	merch description_count_1_by_30
288	cardnum_state_max_1	624	cardnum_merch_count_0_by_7
289	cardnum_state_med_1	625	cardnum_merch_count_0_by_14
290	cardnum_state_total_1	626	cardnum_merch_count_0_by_30
291	cardnum_state_actual/avg_1	627	cardnum_merch_count_1_by_7
292	cardnum_state_actual/max_1	628	cardnum_merch_count_1_by_14
293	cardnum_state_actual/med_1	629	cardnum_merch_count_1_by_30
294	cardnum_state_actual/toal_1	630	cardnum_zip_count_0_by_7
295	cardnum_state_count_3	631	cardnum_zip_count_0_by_14
296	cardnum_state_avg_3	632	cardnum_zip_count_0_by_30
297	cardnum_state_max_3	633	cardnum_zip_count_1_by_7
298	cardnum_state_med_3	634	cardnum_zip_count_1_by_14
299	cardnum_state_total_3	635	cardnum_zip_count_1_by_30
300	cardnum_state_actual/avg_3	636	cardnum_state_count_0_by_7
301	cardnum_state_actual/max_3	637	cardnum_state_count_0_by_14
302	cardnum_state_actual/med_3	638	cardnum_state_count_0_by_30
303	cardnum_state_actual/toal_3	639	cardnum_state_count_1_by_7
304	cardnum_state_count_7	640	cardnum_state_count_1_by_14

305 cardnum_state_avg_7

306 cardnum_state_max_7

307 cardnum_state_med_7

308 cardnum_state_total_7

309 cardnum_state_actual/avg_7

310 cardnum_state_actual/max_7

311 cardnum_state_actual/med_7

312 cardnum_state_actual/toal_7

313 cardnum_state_count_14

314 cardnum_state_avg_14

315 cardnum_state_max_14

316 cardnum_state_med_14

317 cardnum_state_total_14

318 cardnum_state_actual/avg_14

319 cardnum_state_actual/max_14

320 cardnum_state_actual/med_14

321 cardnum_state_actual/toal_14

322 cardnum_state_count_30

641 cardnum_state_count_1_by_30

642 merchnum_zip_count_0_by_7

643 merchnum_zip_count_0_by_14

644 merchnum_zip_count_0_by_30

645 merchnum_zip_count_1_by_7

646 merchnum_zip_count_1_by_14

647 merchnum_zip_count_1_by_30

648 merchnum_state_count_0_by_7

649 merchnum_state_count_0_by_14

650 merchnum_state_count_0_by_30

651 merchnum_state_count_1_by_7

652 merchnum_state_count_1_by_14

653 merchnum_state_count_1_by_30

654 cardnum_merchdes_count_0_by_7

655 cardnum_merchdes_count_0_by_14

656 cardnum_merchdes_count_0_by_30

657 cardnum_merchdes_count_1_by_7

658 cardnum_merchdes_count_1_by_14

323 cardnum_state_avg_30

324 cardnum_state_max_30

325 cardnum_state_med_30

326 cardnum_state_total_30

327 cardnum_state_actual/avg_30

328 cardnum_state_actual/max_30

329 cardnum_state_actual/med_30

330 cardnum_state_actual/toal_30

331 merchnum_zip_day_since

332 merchnum_zip_count_0

333 merchnum_zip_avg_0

334 merchnum_zip_max_0

335 merchnum_zip_med_0

336 merchnum_zip_total_0

659 cardnum_merchdes_count_1_by_30

660 merchdes_zip_count_0_by_7

661 merchdes_zip_count_0_by_14

662 merchdes_zip_count_0_by_30

663 merchdes_zip_count_1_by_7

664 merchdes_zip_count_1_by_14

665 merchdes_zip_count_1_by_30

666 merchdes_state_count_0_by_7

667 merchdes_state_count_0_by_14

668 merchdes_state_count_0_by_30

669 merchdes_state_count_1_by_7

670 merchdes_state_count_1_by_14

671 merchdes_state_count_1_by_30