Yinan Guo

guoy221@wfu.edu

2021/11/01

# Executive Summary

## Problem

The financial institution needs to find fraud, waste and abuse in their payment stream. My job is to build models to identify important factors that can lead to frauds and make predictions to decide whether there will be new frauds in other data.

## Key Findings

1. Email domain is an important factor to indicate fraud to happen since the p-value for email domain count is under 0.05. The more frequent the email domain is used, the more likely the transaction is fraud.
2. Billing postal is an important factor to indicate fraud to happen since the p-value for billing postal count is under 0.05. The more frequent the billing postal is used, the less likely the transaction is fraud.
3. Transactions happened in Oregon has a fraud rate 3-times higher than average fraud rate and transactions happened in Massachusetts has a fraud rate which is only 1/3 of average fraud rate.
4. Transactions using EURO has a fraud rate 1 time higher than transactions using CAD.

## Model Performance Summary & Interpretation

1. My final model is a logistic regression model with an accuracy of 0.983 and AUC of 0.972 for test dataset, which performs great. I select predicters other than some of insignificant factors like phone number and customer name, etc. The most important predictors are transaction adjusted amount, email domain and billing postal. With higher transaction adjusted amount, more frequently used email domain and billing postal, the more likely the transaction is to fraud.
2. At the FPR of 6%, the TPR is 93.2% and the threshold is 0.088. At this threshold, when the probability to fraud is higher than 0.088, we predict it as fraud and when the probability to fraud is lower than 0.088, we predict it as legit. By lowering the threshold from 0.5 to 0.088, the FPR increased since more situations are predicted as fraud while the TPR is also increased. The 6% of FPR is acceptable in change of increasement in TPR since it is more important for the institutions to predict or discover the fraud on time to prevent loss in profits while the loss cost by wrongly discover fraud is not so high.

## Recommendations

1. Pay attention to the transactions that happens with the cvv of A, B, J, P since they will probably cause fraud and the institution should make a phone call to the customers as soon as detecting such situations.
2. Since transactions happened in A, B, J, P and Y types of environments are highly likely to fraud, the institution can introduce some automatic fraud detection systems and ask customers to verify their transactions when it is in such environments to prevent fraud.

# Detailed Analysis & Steps

### File(s) Summary

| **File Name** | **Record count** | **Column count** | **Numeric columns** | **Character columns** | **Timestamp columns** |
| --- | --- | --- | --- | --- | --- |
| Project\_2\_training.csv | 125000 | 27 | 7 | 19 | 1 |
| Project\_2\_holdout.csv | 25000 | 27 | 8 | 18 | 1 |

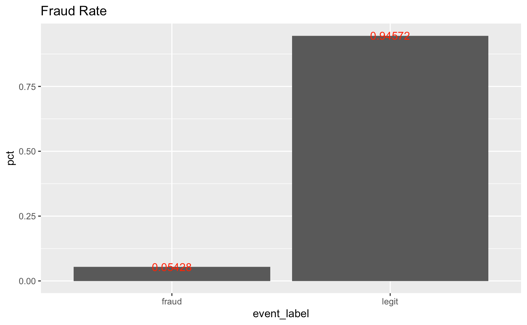
### Field Summary

| **Name** | **Data Type** | **Feature Type** | **Count** | **# Distinct** | **% Distinct** | **# Missing** | **% Missing** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| event\_label | char | target | 125000 | 2 | 0.00% | 0 | 0.00% |
| ip\_address | char | categorical | 125000 | 13314 | 10.65% | 0 | 0.00% |
| user\_agent | char | categorical | 125000 | 8571 | 6.86% | 0 | 0.00% |
| email\_domain | char | categorical | 125000 | 6992 | 5.59% | 0 | 0.00% |
| phone\_number | char | categorical | 125000 | 11928 | 9.54% | 0 | 0.00% |
| billing\_city | char | categorical | 125000 | 8980 | 7.18% | 0 | 0.00% |
| billing\_postal | char | categorical | 125000 | 11065 | 8.85% | 0 | 0.00% |
| billing\_state | char | categorical | 125000 | 51 | 0.04% | 0 | 0.00% |
| card\_bin | char | categorical | 125000 | 6322 | 5.06% | 0 | 0.00% |
| currency | char | categorical | 125000 | 4 | 0.00% | 0 | 0.00% |
| cvv | char | categorical | 125000 | 26 | 0.02% | 0 | 0.00% |
| signature\_image | char | categorical | 125000 | 27 | 0.02% | 0 | 0.00% |
| transaction\_type | char | categorical | 125000 | 27 | 0.02% | 0 | 0.00% |
| transaction\_env | char | categorical | 125000 | 27 | 0.02% | 0 | 0.00% |
| applicant\_name | char | categorical | 124876 | 84958 | 67.97% | 124 | 0.10% |
| billing\_address | char | categorical | 124889 | 124884 | 99.91% | 111 | 0.10% |
| merchant\_id | char | id | 124911 | 124904 | 99.92% | 89 | 0.10% |
| locale | char | categorical | 124885 | 293 | 0.23% | 115 | 0.10% |
| tranaction\_initiate | char | categorical | 124900 | 26 | 0.02% | 100 | 0.10% |
| event\_timestamp | POSIXct | TimeStamp | 124910 | 124685 | 99.45% | 90 | 0.07% |
|  |  |  |  |  |  |  |  |

For numeric variables:

| **column** | **n** | **dist** | **nmiss** | **mean** | **min** | **max** |
| --- | --- | --- | --- | --- | --- | --- |
| event\_id | 125000 | 125000 | 0 | 1500444 | 20 | 2999960 |
| account\_age\_days | 125000 | 6363 | 0 | 4642 | -1 | 9119 |
| transaction\_amt | 125000 | 3641 | 0 | 2520 | -1 | 4880 |
| transaction\_adj\_amt | 125000 | 93 | 0 | 54.1 | -1 | 99 |
| historic\_velocity | 125000 | 6633 | 0 | 4700 | -1 | 8875 |
| days\_since\_last\_logon | 124887 | 101 | 113 | 49.8 | 0 | 100 |
| inital\_amount | 124891 | 13999 | 109 | 8000 | 1000 | 15000 |

## Target Summary

* The fraud rate is 5.43% and the default rate is 94.57%.

|  |  |  |
| --- | --- | --- |
| Fraud | n | pct |
| No | 118215 | 0. 94572 |
| yes | 6785 | 0. 05428 |

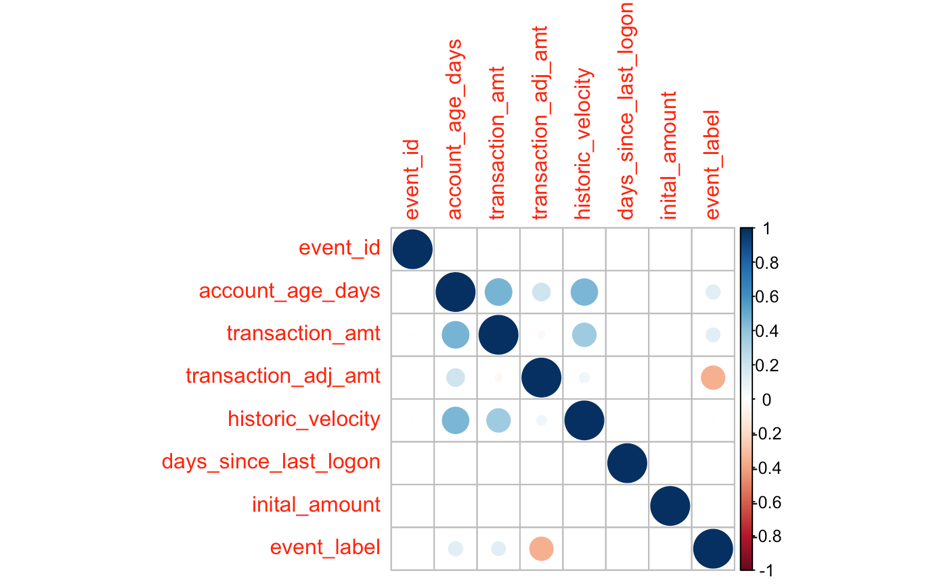
## Exploratory Data Analysis & Screening

### Descriptive Statistics

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

Here are the boxplots for most of the numeric variables including variables mutated from characters like billing city. Among all the numeric variables, account age days, transaction amount, adjusted transaction amount, user agent count and email domain have different distributions within different groups divided by fraud or not, indicating that these may be important factors to affect frauds.

### Correlation Analysis



Among all the numeric variables, transaction adjusted amount has the largest correlation with fraud and they are negatively related. Also, historic velocity and transaction amount are positively related to account age days.

### Frequency Analysis

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  |  |  |
|  |  |  |

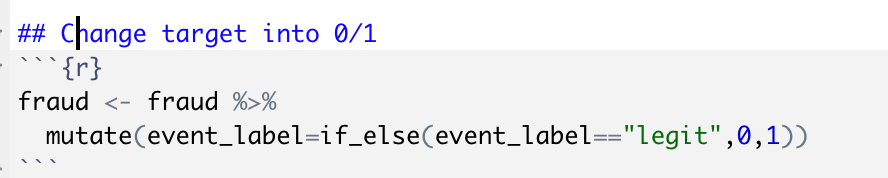
According to the histogram of characters, if different columns in a graph have different proportions of red part, which indicates the proportion of fraud, then the fraud rate is strongly correlated with this variable. (i.e. The barplot for transaction environment shows a strong difference in proportion of fraud in different environments, indicating that the transaction environment may be a great sign for fraud.)

## Initial Screening & Exploration

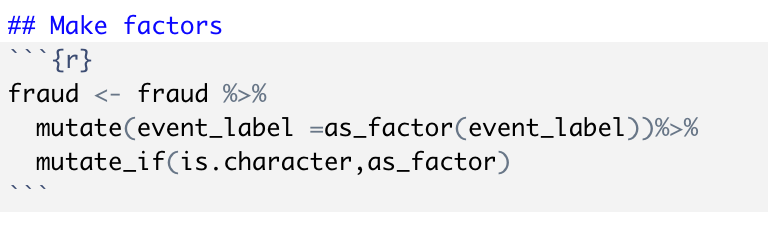
|  |  |
| --- | --- |
| Payments occur in A, P and Y environment have fraud rates about 1, which is really high and indicates that such environment will be dangerous signal for fraud. And L will be a safe transaction environment which has fraud rate of 0. | Transactions paid by EURO has the highest fraud rate, which is 2 times of those paid with USD. Thus, currency is an important predictor. |
| Transactions in type A has fraud rate that is 4 times of average fraud rate. And type B, J, P also has high fraud rate. Thus, transaction type is an important predictor. | Transactions in type A and type Y signature image have high rate of fraud, which is about 10 times of average. Thus, signature image is an important predictor. |
| Transactions that fraud has larger average number of days since the account was created. Thus, it may be a useful predictor. | The distribution for days since last logon in different groups are almost the same so that it is not a useful predictor. |

## Data Preparation & Transformation

1. The target is not shown as 0/1 but legit/fraud. I mutate it into binary of 0/1.



1. And then turn characters into factors.



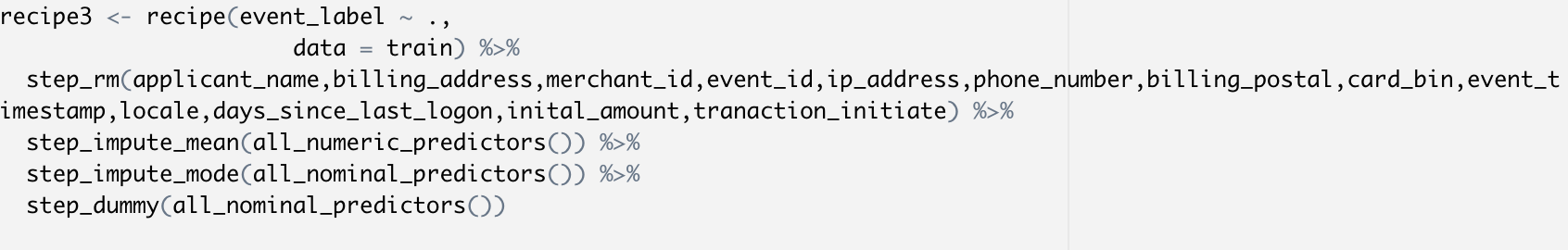
1. Using recipe to deal with variables

By step\_rm, i remove some unimportant variables. (This is an example for full logistic model.)

By step\_impute\_mean I deal with missing numeric variables to make all of them to be mean value of that variable.

By step\_impute\_mode, I deal with missing character variables to make all of them to be mode value of that variable.

By step\_dummy, I deal with character variables to make predictions.



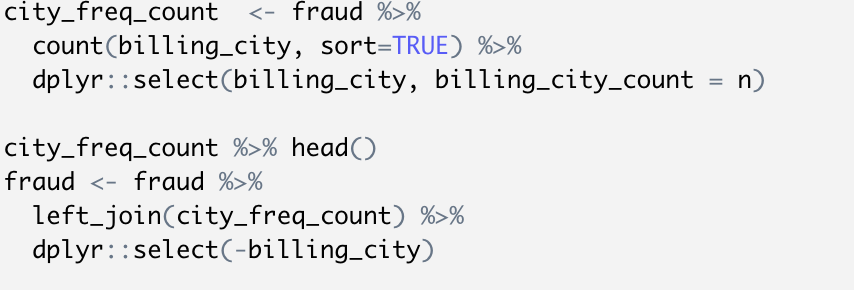
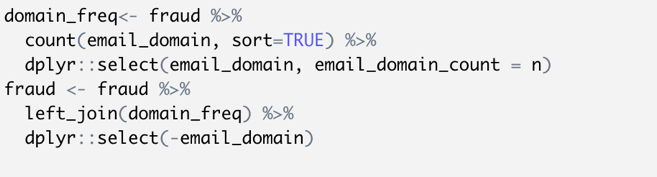
### Derive new variables

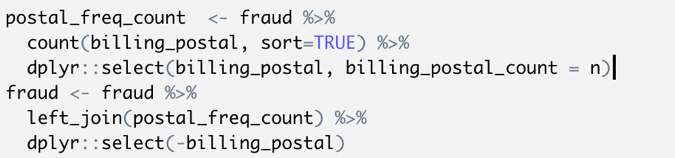
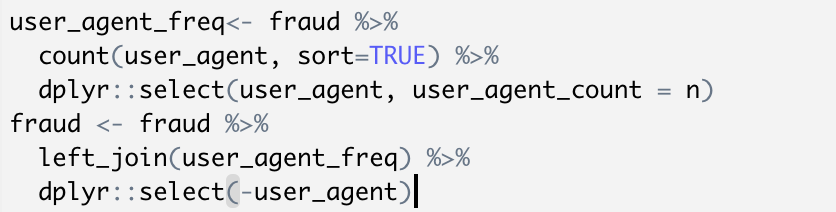
Some of characters with too many distinct values could be transformed in the following way. It is a way to utilize the frequency of the categories. In the cases where the frequency is related somewhat to the target variable.

1. Select a categorical variable you would like to transform

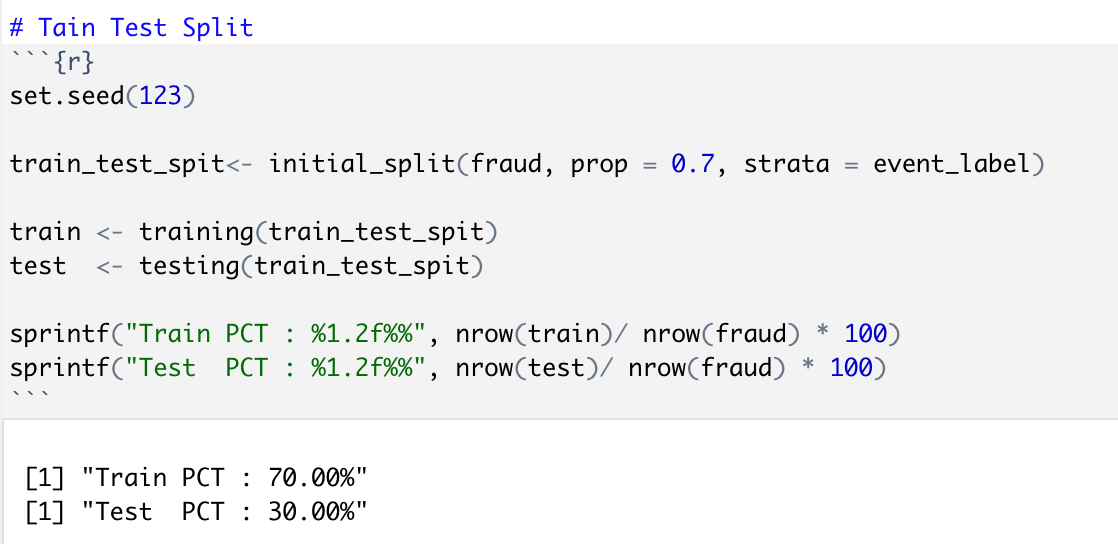
2. Group by the categorical variable and obtain counts of each category

3. Join it back with the training data set





## Model Building

70% of data are used to build the model and 30% are used for validation. 

## Model Training

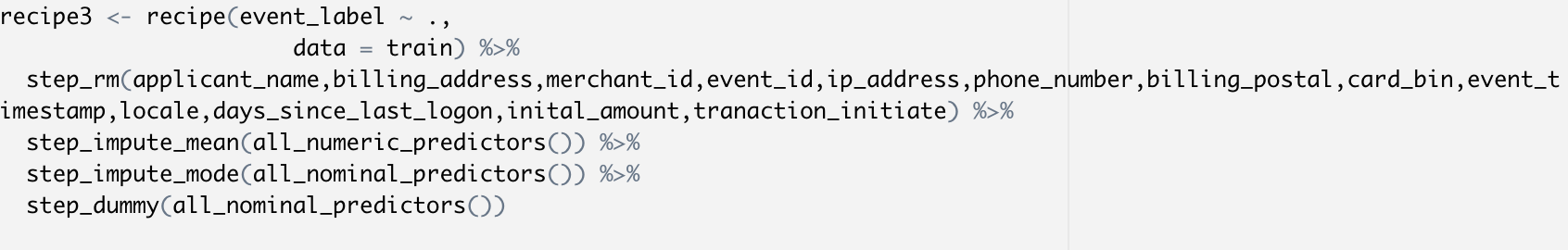
* Document the variables and the roles used

| **Name** | **Data Type** | **In models** |
| --- | --- | --- |
| event\_label | num (target) | / |
| ip\_address | char | 0 |
| user\_agent\_count | num | 1 |
| domain\_freq | num | 1 |
| phone\_number | char | 0 |
| billing\_city\_count | char | 1 |
| billing\_postal\_rate | char | 1 |
| billing\_state | char | 1 |
| card\_bin | char | 0 |
| currency | char | 1 |
| cvv | char | 1 |
| signature\_image | char | 1 |
| transaction\_type | char | 1 |
| transaction\_env | char | 1 |
| applicant\_name | char | 0 |
| billing\_address | char | 0 |
| merchant\_id | char | 0 |
| locale | char | 0 |
| tranaction\_initiate | char | 1 |
| event\_timestamp | POSIXct | 0 |
| event\_id | num | 0 |
| account\_age\_days | num | 1 |
| transaction\_amt | num | 1 |
| transaction\_adj\_amt | num | 1 |
| historic\_velocity | num | 1 |
| days\_since\_last\_logon | num | 0 |
| inital\_amount | num | 0 |

Some variables are removed because they have too many distinct values while they can’t provide useful information to predict fraud (ie. Billing\_address, customer\_name). Other variables are removed based on p-values in logistic regression model. (ie. Days\_since\_last\_logon, initial\_amount).

* Define your Recipe

Recipe for all three models:



* Define your Model
  + Document your hyper parameters

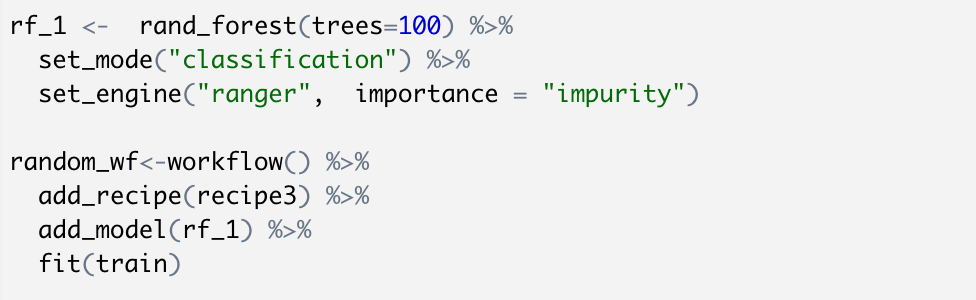
Except what has been removed in the recipes, all of other predictors are used in the model.

The top 5 important factors for each model are:

|  |  |  |
| --- | --- | --- |
| Logistic | Decision Tree | Random Forest |
| Transaction adjusted amount | Transaction adjusted amount | Transaction adjusted amount |
| Account age days | Account age days | Email domain |
| currency | Email domain | Billing postal |
| Email domain | Historic velocity | Account age days |
| Billing postal | User agent count | User agent |

* Create a workflow and fit the model

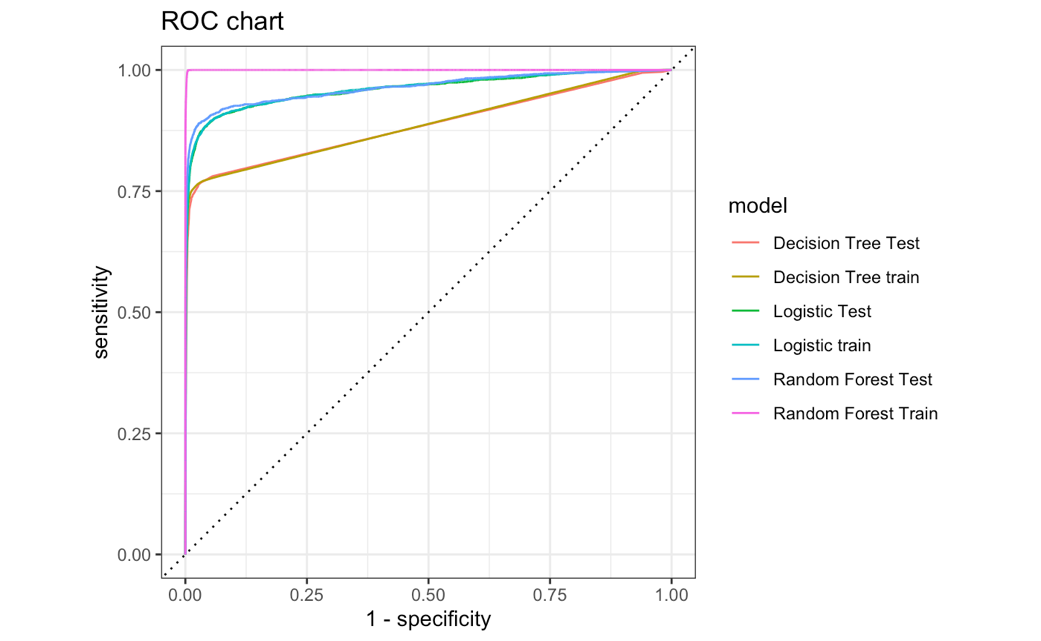
Codes are shown as following:



* Evaluate metrics on Train and Test:
  + Classification:
    - table of Metrics:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Accuracy | | AUC | |
| Train | Test | Train | Test |
| Logistic | 0.9827 | 0.9827 | 0.9736 | 0.9721 |
| Decision tree | 0.9803 | 0.9766 | 0.8870 | 0.8853 |
| Random Forest | 0.9920 | 0.9816 | 0.9998 | 0.9638 |

* + - ROC comparing train and test



* + - confusion matrixes comparing train and test

|  |  |
| --- | --- |
| Train | Test |
| Logistic Regression | |
|  |  |
| Decision Tree | |
|  |  |
| Random Forest | |
|  |  |

* + - a table of FPR/TPR/Precision and Score threshold

According to the requirement of a FPR of 6%, I can have a threshold for logistic regression of 0.056, a threshold for decision tree of 0.017, a threshold for random forest of 0.088.

|  |  |
| --- | --- |
| Logistic Regression | Decision tree |
|  |  |
| Random Forest |  |
|  |  |

* + - a chart / table of variable importance
* Top 10 important variables for models

|  |  |
| --- | --- |
| Logistic Regression | Decision tree |
|  |  |
| Random Forest |  |
|  |  |

* + - score distributions for test dataset

|  |  |
| --- | --- |
| Logistic Regression | Decision tree |
|  |  |
| Random Forest |  |
|  |  |

## Model Comparison

* Compare and contrast 3 models

Here is the chart showing accuracy, auc, precision and recall for all of the models.

* Accuracy is the rate for (TP+TN)/(TP+TN+FP+FN)
* Precision is the rate for TP/(TP+FP), which represents the positive predictive value and important to a model.
* Recall is the rate for TP/(TP+FN), which represents the true positive rate and important to a model.
* AUC is the area under the ROC.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  | Accuracy | AUC | Precision | Recall |
| Logistic Regression | train | 0.9827 | 0.9736 | 0.7176 | 0.9067 |
| test | 0.9827 | 0.9721 | 0.7157 | 0.8894 |
| Decision Tree | train | 0.9803 | 0.8870 | 0.6955 | 0.9225 |
| test | 0.9766 | 0.8853 | 0.6601 | 0.8783 |
| Random Forest | train | 0.9920 | 0.9998 | 0.8552 | 0.9971 |
| test | 0.9816 | 0.9638 | 0.6877 | 0.9615 |

* Which model performs better and why?

Logistic regression performs the best.

1. The logistic regression model has the largest AUC and accuracy for test dataset.
2. The precision is higher than decision tree while only slightly lower than random forest’s.
3. Though random forest has the largest accuracy and auc for train set, the overfitting problem hidden in this model is severe as the difference in auc between test and train is over 0.03 whereas the difference in auc and accuracy between train and test for logistic regression is really low.

* Compare their auc/accuracy/precision or lift on training and test sets.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  | Accuracy | AUC | Precision | Recall |
| Logistic Regression | train | 0.9827 | 0.9736 | 0.7176 | 0.9067 |
| test | 0.9827 | 0.9721 | 0.7157 | 0.8894 |
| Decision Tree | train | 0.9803 | 0.8870 | 0.6955 | 0.9225 |
| test | 0.9766 | 0.8853 | 0.6601 | 0.8783 |
| Random Forest | train | 0.9920 | 0.9998 | 0.8552 | 0.9971 |
| test | 0.9816 | 0.9638 | 0.6877 | 0.9615 |

# Plain Explanation to Random Forest

Random forest is a package of a bunch of independent decision trees. In each decision tree, variables are randomly selected to build the decision tree models. And in the end, the overall probability is the average probabilities that given by each decision trees. The more frequent a variable is chosen by decision trees models, the more important that variable is to predict the target.