Qidi Zheng

Zhenq22@wfu.edu

12/09/2022

# Executive Summary

## Problem

The company is tasked with developing and comparing several machine learning models to identify which loans are likely to default. Also, management is curious about what predictors could impact the loan status.

## Key Findings

* Those who have higher range of loan amount from 5000 to 15000, longer term with 60 months tend to default. This could due to that since people who have larger loan amount and longer term do not necessarily have the ability to repay their loan thus contributing to default
* People who have lower LC assigned loan grade and lower boundary range of FICO tend to default. This could result from the fact that people who do not have a good habit of credit behavior could get a lower loan grade and FICO which lead to the result of default.
* Based on the local interpretation, the most recent month LC pulled credit for this loan could significantly affect the result of default. As a matter of fact, when the most recent month LC pulled credit is around 3 months, it is more likely to default. This may result from the fact that comparatively new borrowers could be more prone to default.
* From the break down profile, we could get profiles of loaner who are more likely to defaut: the most recent month LC pulled credit is around 3 months, last payment amount is between 290 to 747.36 and late fee between 14.9 to 37.4. This could suggest that people who default are prone to repay less loan and have higher late fee.

## Model Performance Summary & Interpretation

I used random forest model, XG Boosting model and neural network model to predict which loans are likely to default and found out XG Boosting model performs best, with an accuracy of 0.938 and AUC of 0.969 for the testing dataset, meaning it has a strong predicting ability. However, we would like to pay less attention to the accuracy since our dataset is imbalanced and major class which is not default would affect the predicting result.

## Recommendations

* Find a trade off between FP and TP. Based on the profiles of loaners who are more likely to default and who seem likely to default but actually not, the characteristics are very similar. For one thing, it costs a lot to detect default and loaners will also be disturbed if FPR is too high since lots of legit accounts will be recognized as default. Also, if the FPR is too high, it will not only hurt the company’s reputation but also the trust between the company and its loaners will be ruined. Thus, it’s more significant to pay attention to the threshold of classification to identify the true default since it might hurt to lose loaners who actually behave.
* Pay attention to the characteristic of loan contributing to default such as the most recent month LC pulled credit, term and other significant variables. Be more careful when reviewing those applications with the characteristics to mitigate the chances of default.
* Find a balanced value for precision and recall for better prediction When we look at the relationship between precision and recall, the precision increases, the recall may decrease. While we all want higher recall and precision to better predict the default cases, in reality, it’s not necessarily the case. In this case, recall rate is much more important to us since we would not want our loaners to be wrongly treated, and we also need a relatively high precision to detect those default cases. I recommend the company to have a relatively high value of recall and then look for a higher precision.
* Refine the model using more datasets or other variables to improve the recall and precision. I recommend the company could expand the dataset and find more variables impacting the target variable to refine the predict model as much as they can.
* From anomaly detection, we could find out outliers could affect the model predictions a lot. Therefore, digging into the dataset to detect anomaly or outliers and mitigate its impact to the whole model might be a better choice before modeling process.

# MODEL REPORT

# Detailed Analysis & Steps

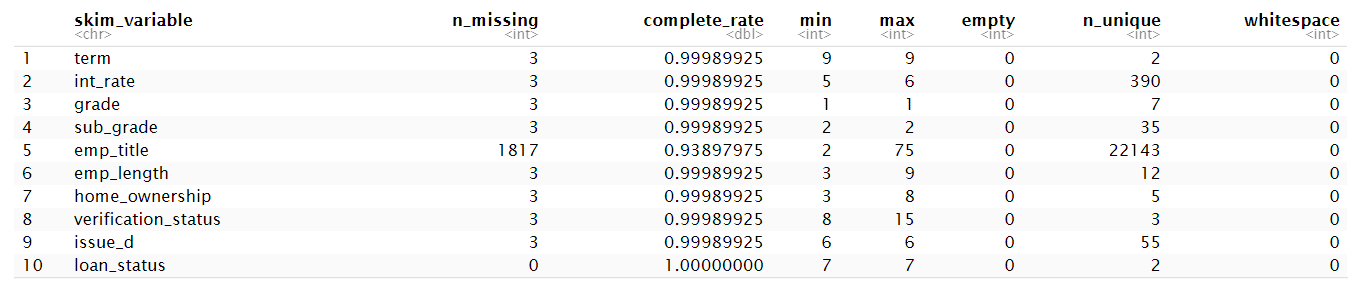
### File(s) Summary

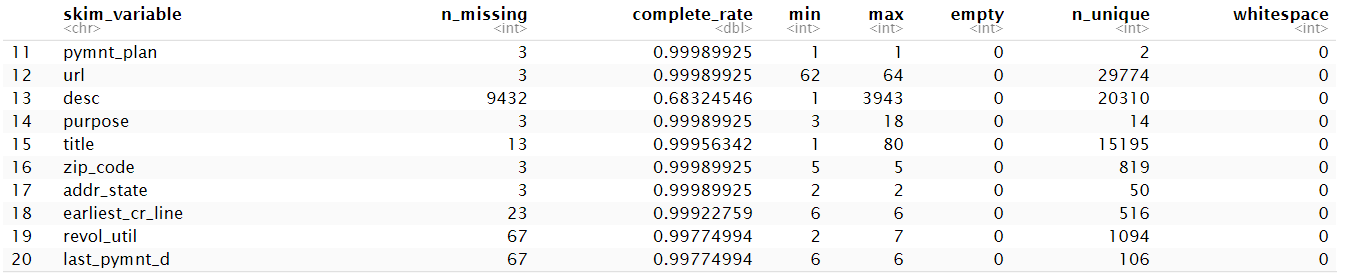
For each file you import you should provide a summary kind of like this

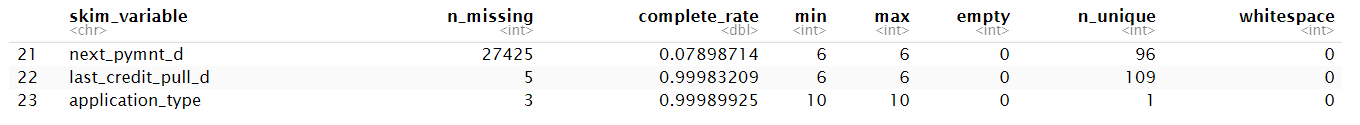
| **File Name** | **Record count** | **Column count** | **Numeric columns** | **Character columns** |
| --- | --- | --- | --- | --- |
| loan\_train.csv | 29777 | 52 | 29 | 23 |
| loan\_holdout.csv | 12761 | 51 | 29 | 22 |

### Field Summary

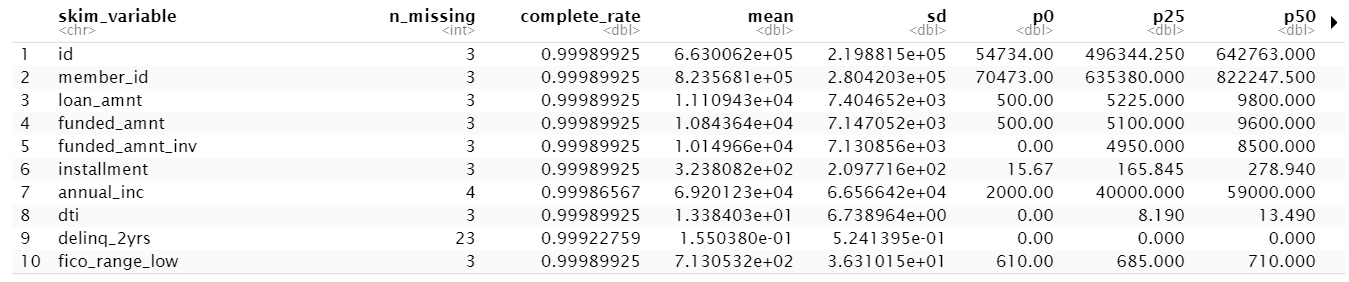
Categorical Variables Analysis

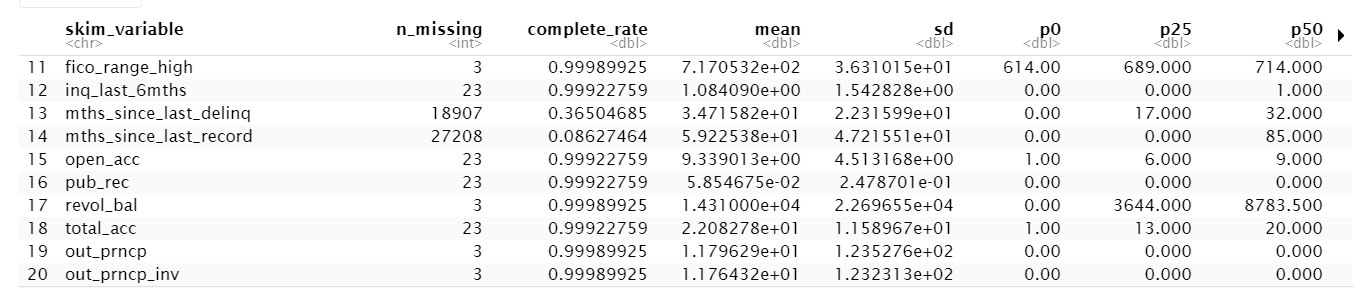


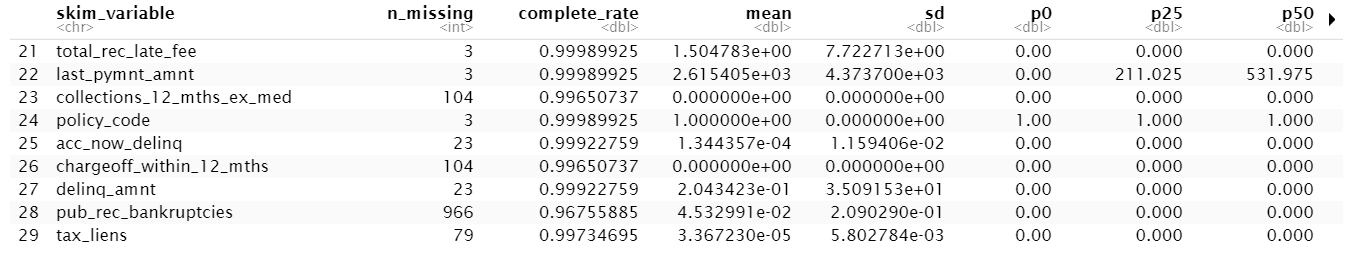




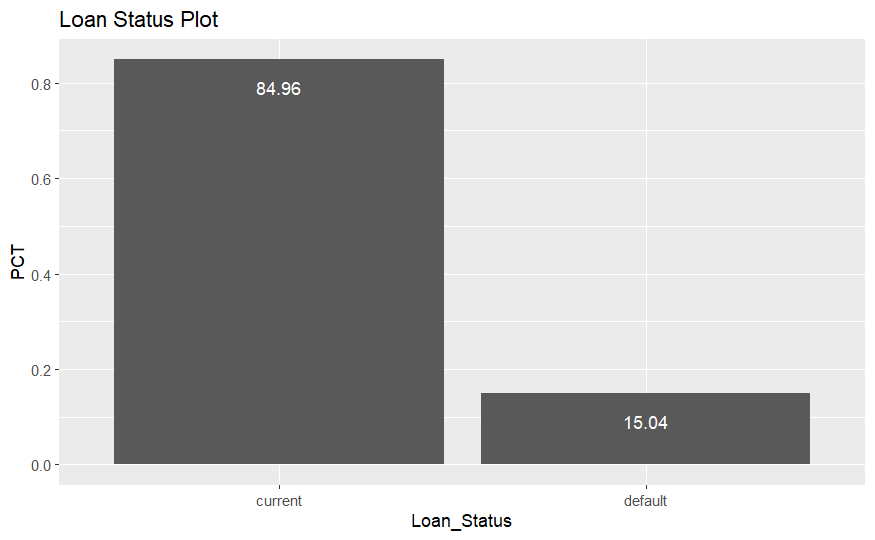
Numeric Table Analysis





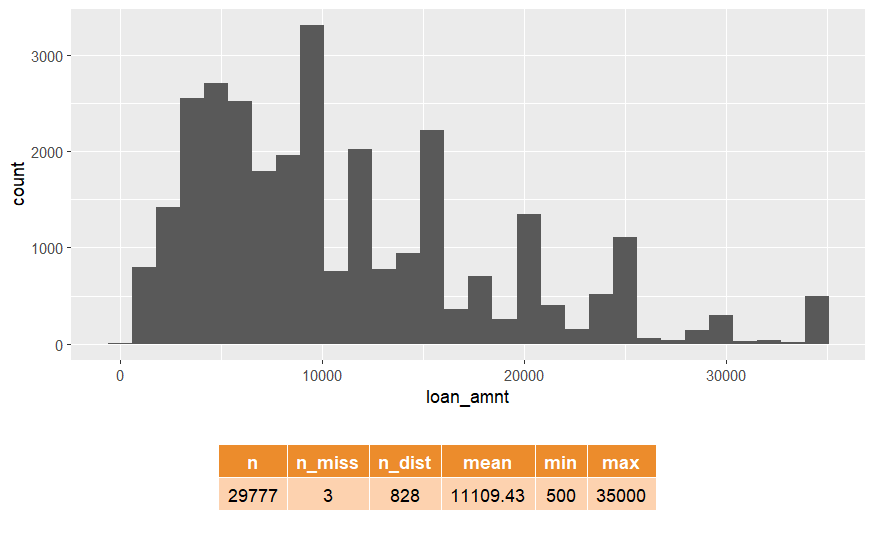


## Target Summary

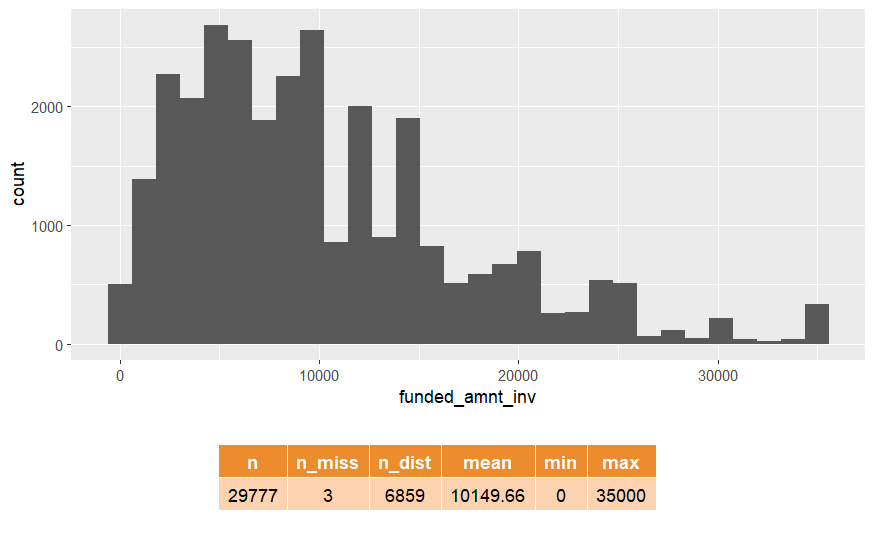
The chart above shows 84.96% of loan status are current, while 15.04% of loans are default. 

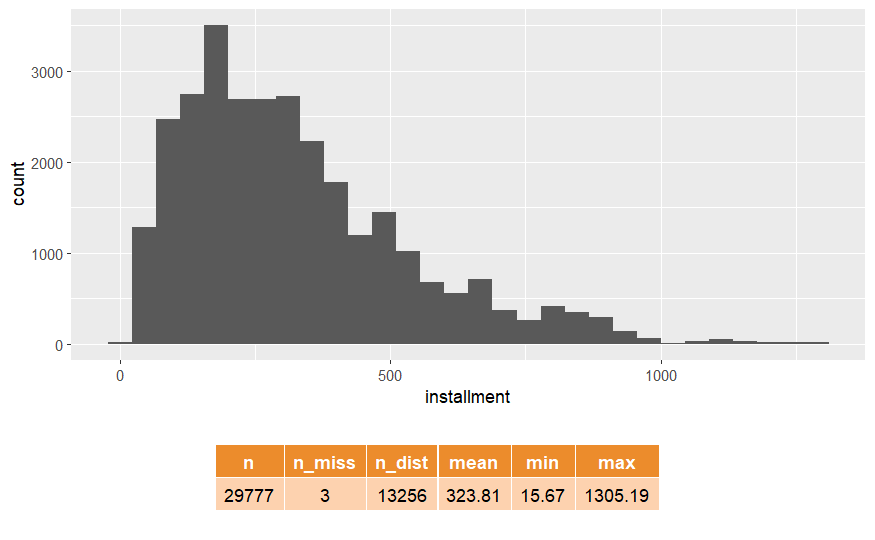
## Exploratory Data Analysis & Screening

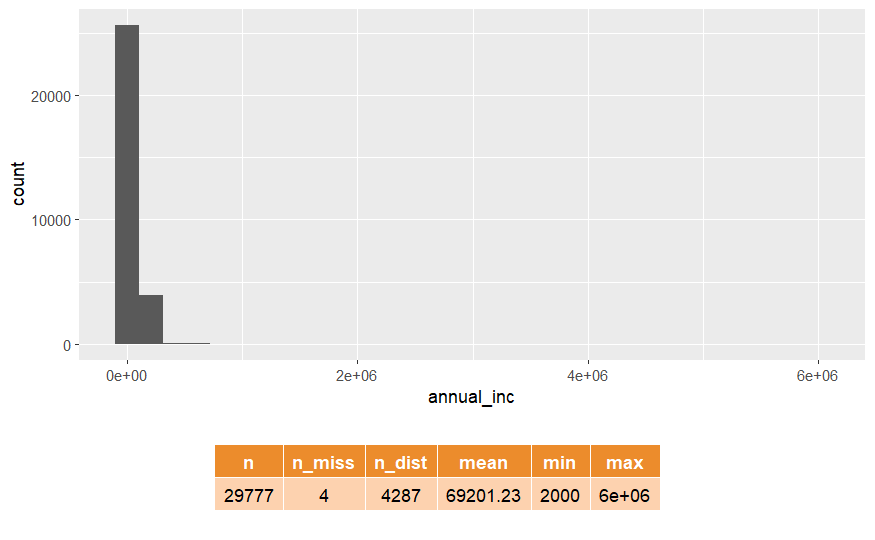
### Descriptive Statistics

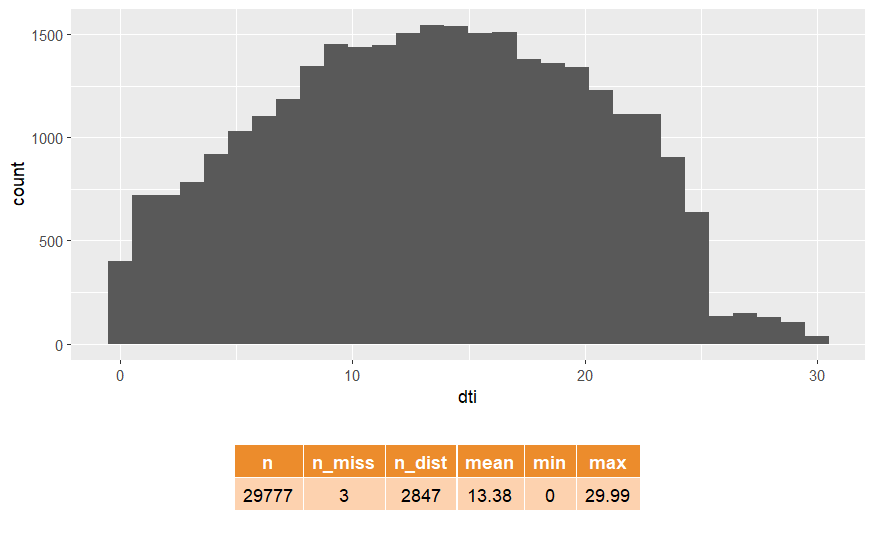


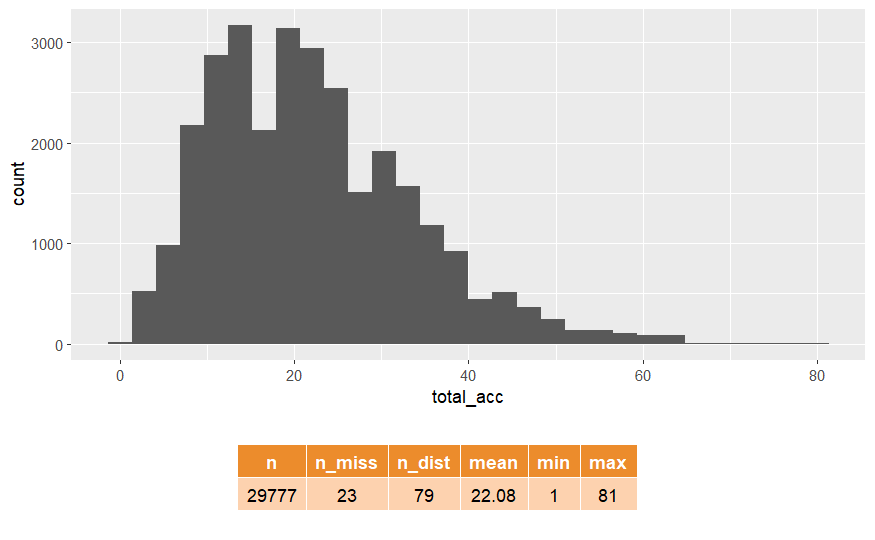
### 

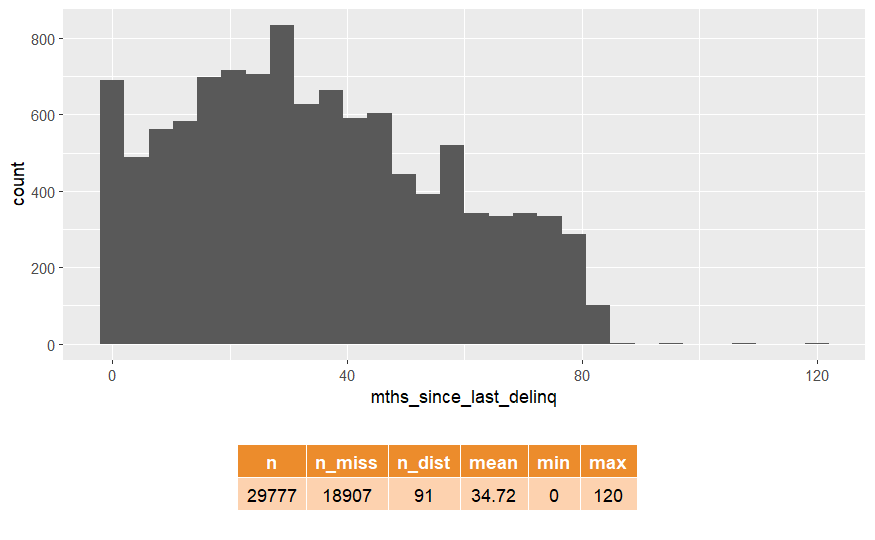


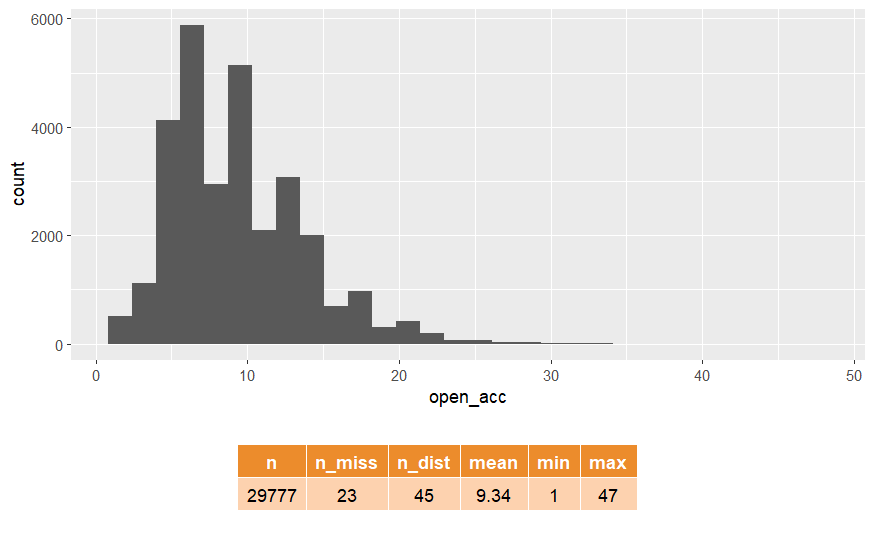






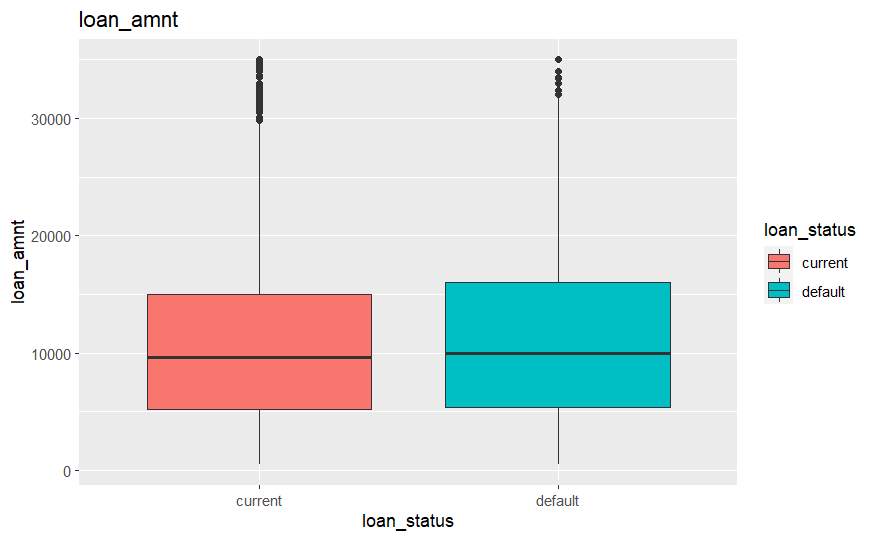


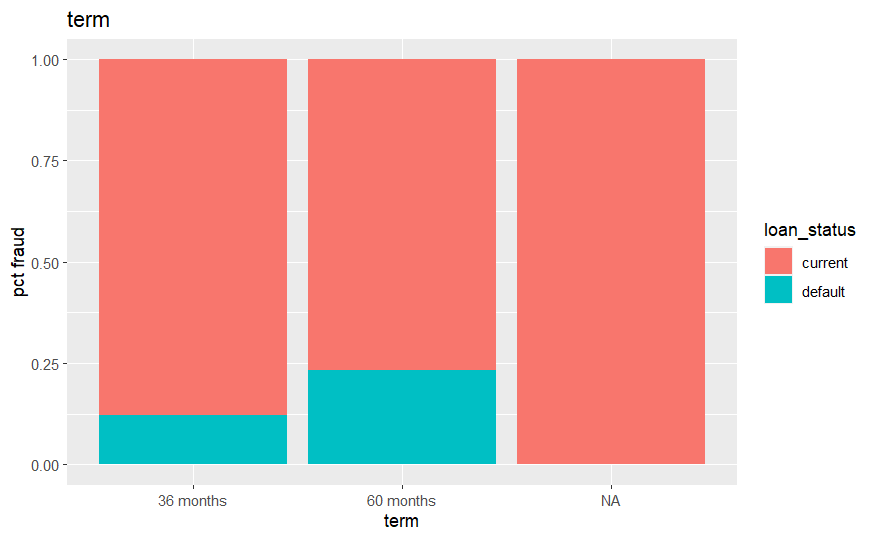


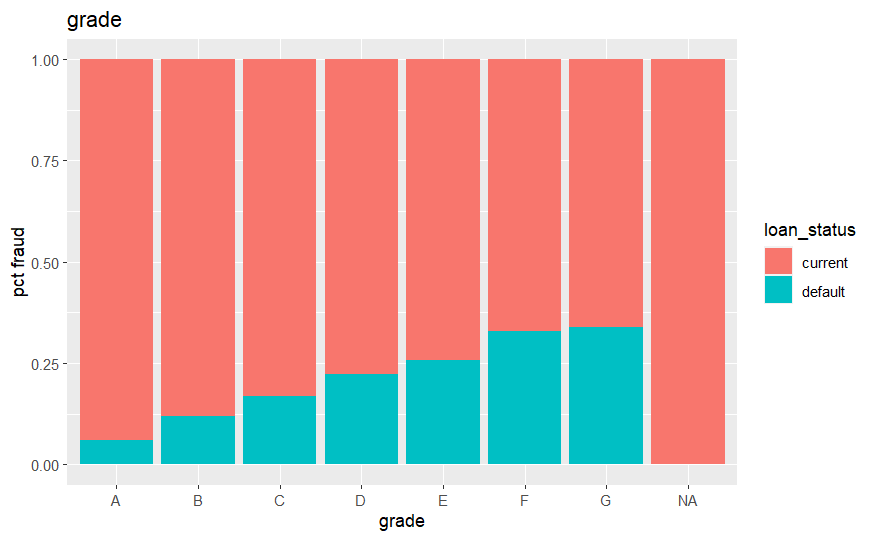


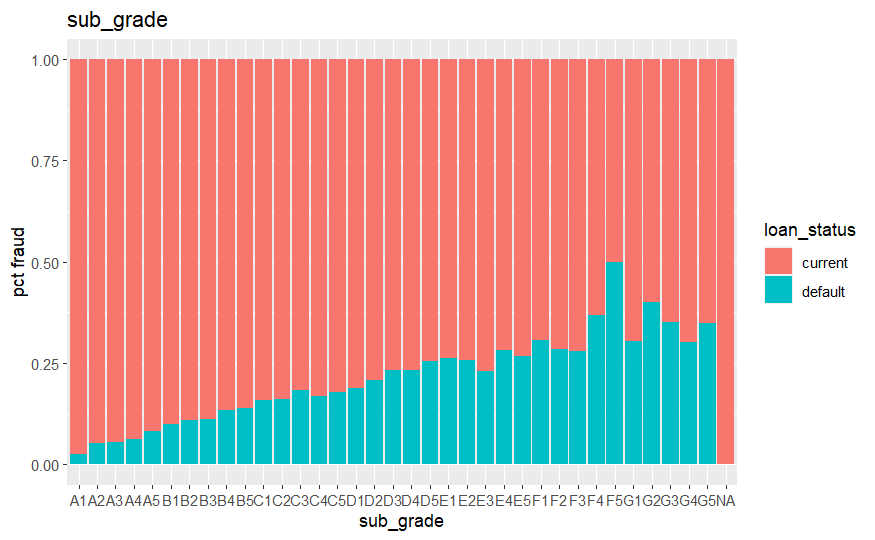
### Frequency Analysis

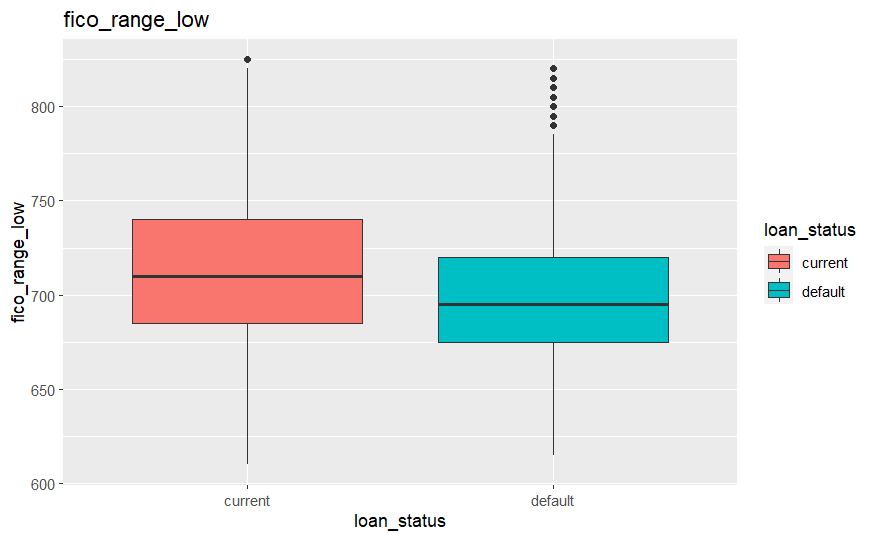
These charts present the relationship between our predictors and the target variables.

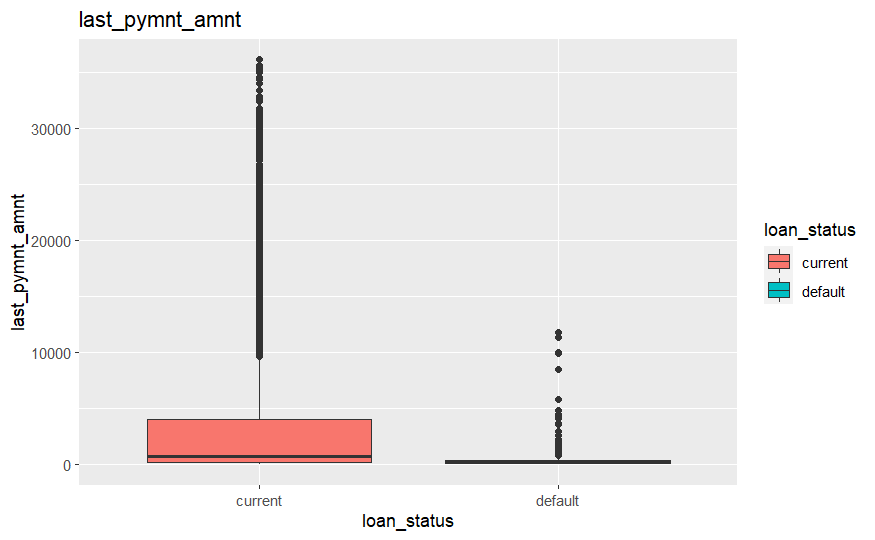






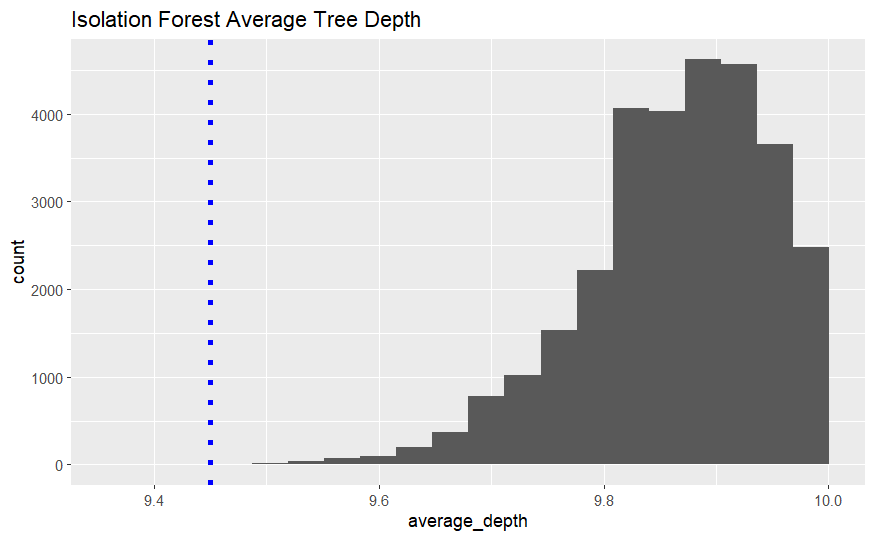


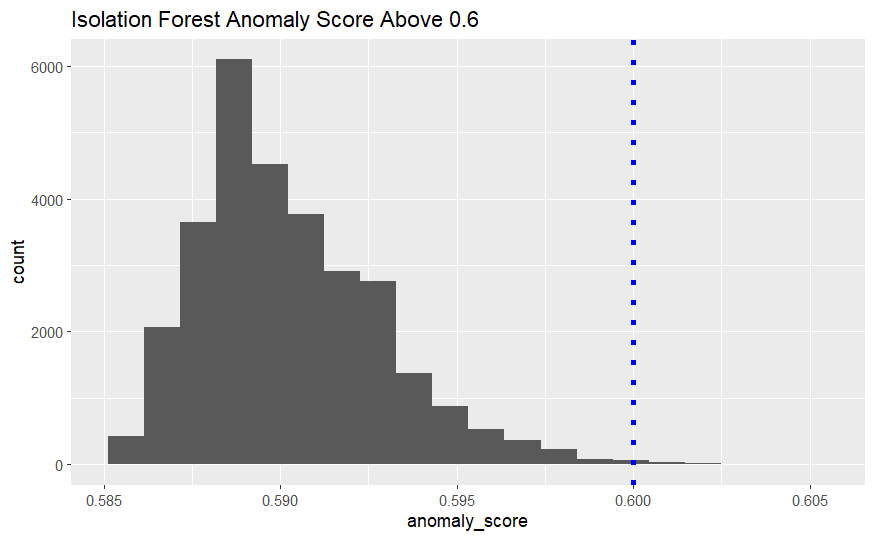




## Anomaly Detection

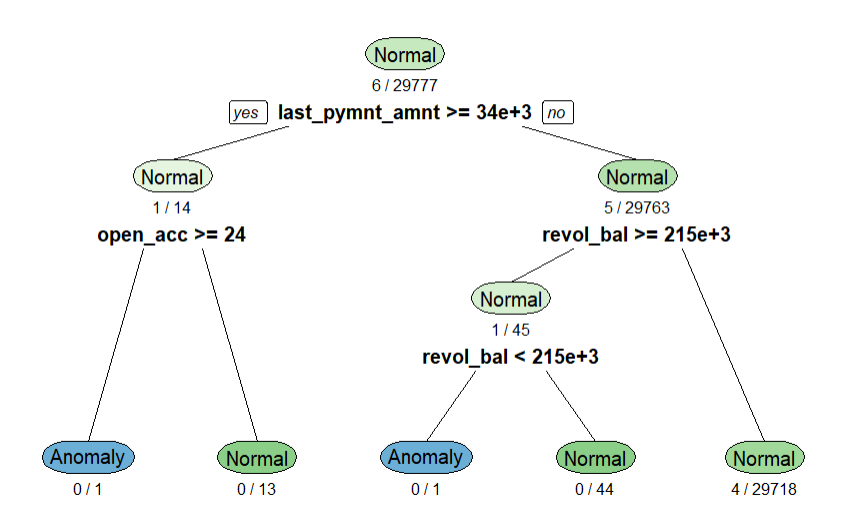
I use the isolation trees to detect anomalies. From a sample size of 1000 trees, average depth less than 9.45 and anomaly score above 0.6, we could notice there are about 6 anomalies.

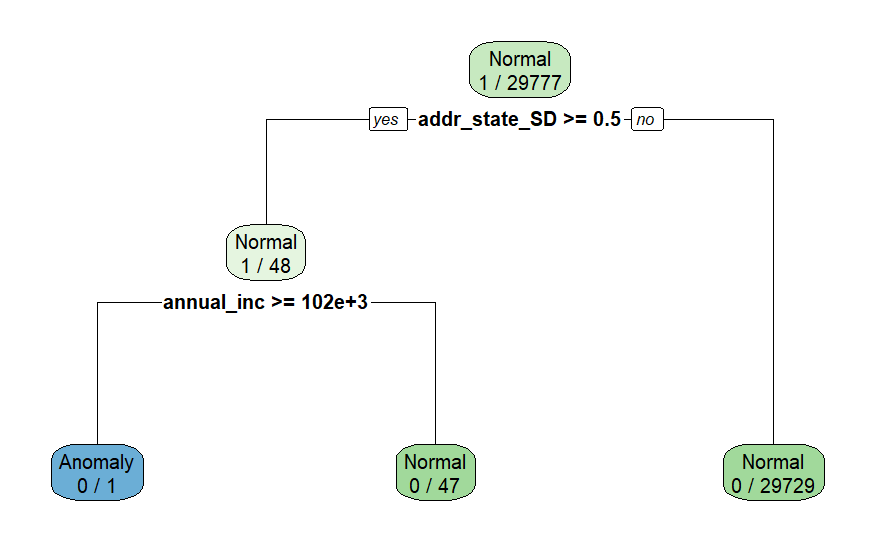




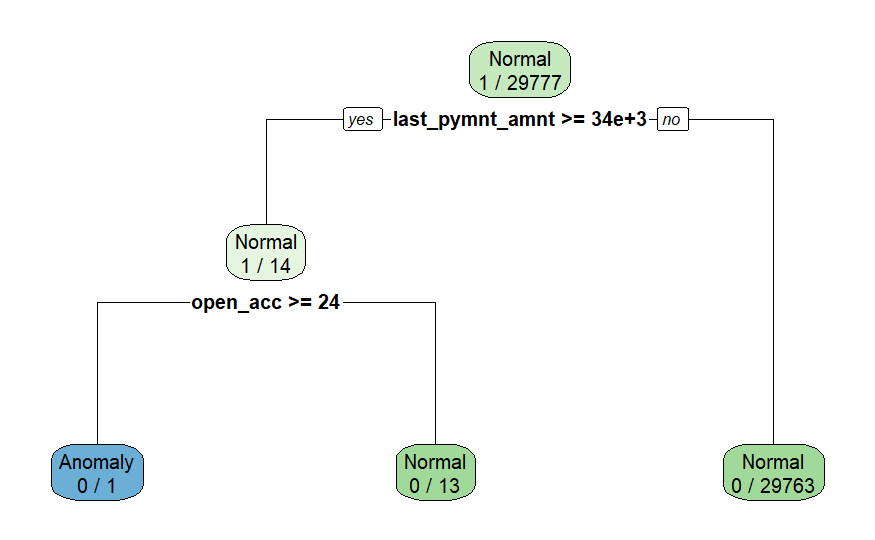
Then we could develop our outlier tree.

Global Anomaly rules:

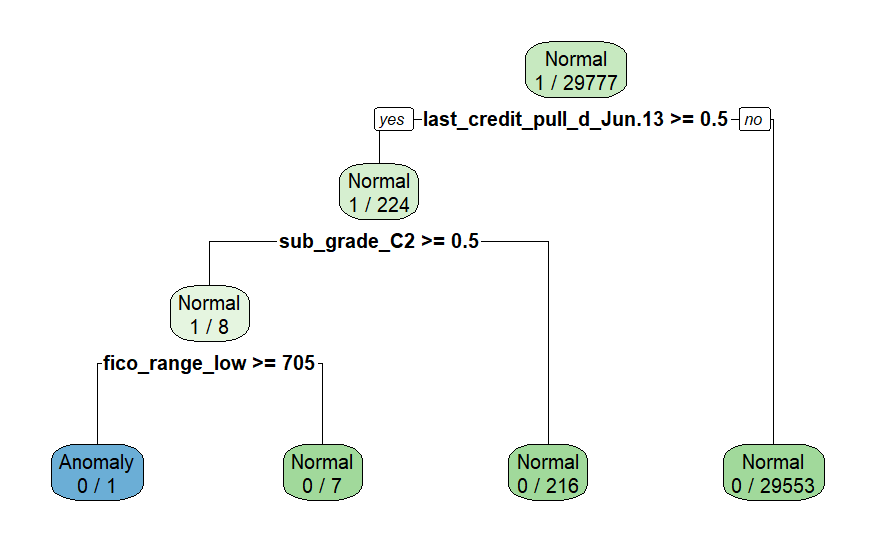
Local Anomaly Rules

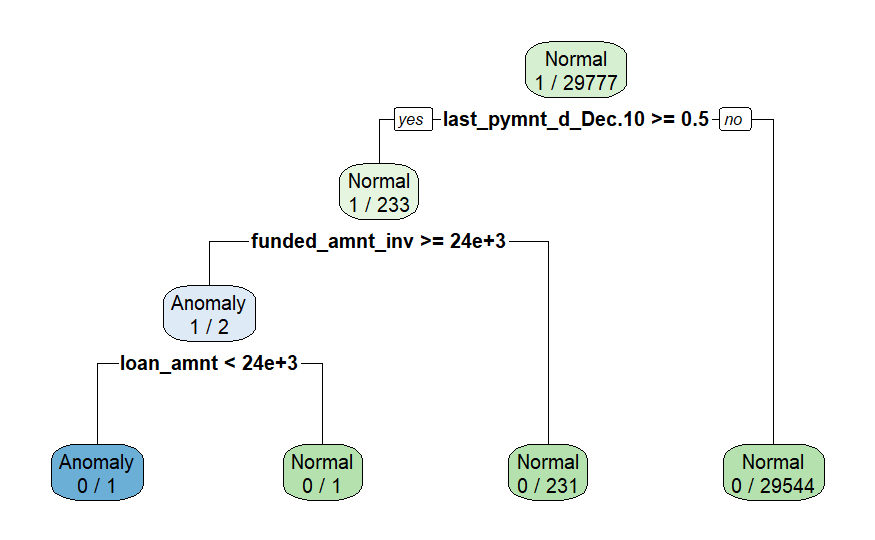
Based on the local anomaly rules, we could profile our 6 local anomaly detection rules:

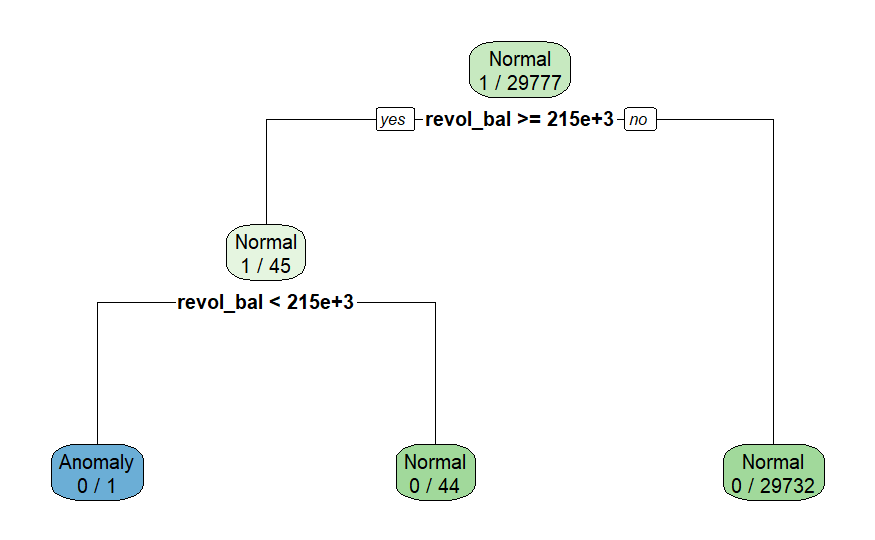
1. IF last\_pymnt\_amnt >= 34167 & open\_acc >= 24
2. IF sub\_grade\_G1 >= 0.5 & purpose\_medical >= 0.5
3. IF last\_credit\_pull\_d\_Jun.13 >= 0.5 & sub\_grade\_C2 >= 0.5 & fico\_range\_low >= 705
4. IF last\_pymnt\_d\_Dec.10 >= 0.5 & funded\_amnt\_inv >= 23591 & loan\_amnt < 24125
5. IF revol\_bal is 214867 to 215274
6. IF addr\_state\_SD >= 0.5 & annual\_inc >= 100000

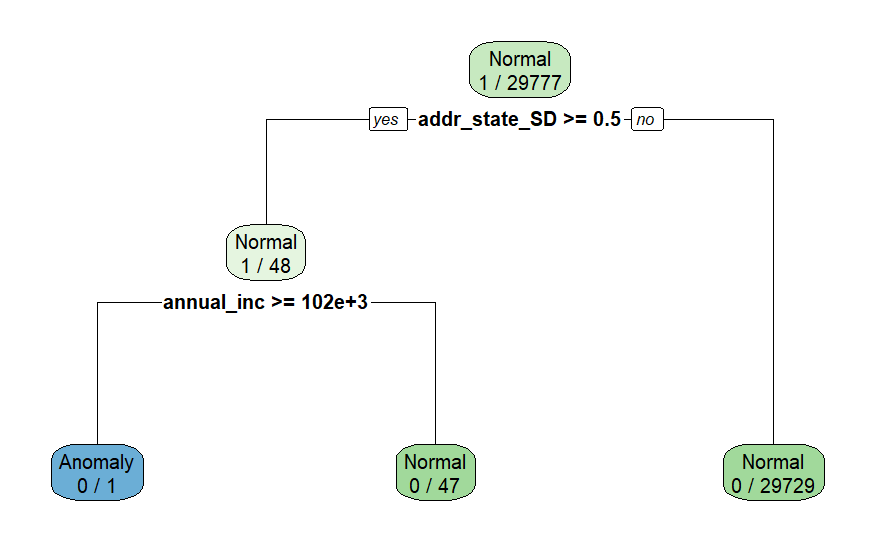


## 









## Data Preparation & Transformation

1. Data preprocessing
   * Formula
     1. loan\_status ~ loan\_amnt + funded\_amnt+funded\_amnt\_inv+term+installment+grade+sub\_grade+emp\_length+home\_ownership+annual\_inc+verification\_status+issue\_d+loan\_status+pymnt\_plan+purpose+addr\_state+dti+delinq\_2yrs+fico\_range\_low+fico\_range\_high+inq\_last\_6mths+open\_acc+pub\_rec+revol\_bal+total\_acc+out\_prncp+out\_prncp\_inv+total\_rec\_late\_fee+last\_pymnt\_d+last\_pymnt\_amnt+last\_credit\_pull\_d+collections\_12\_mths\_ex\_med+policy\_code+application\_type+acc\_now\_delinq+chargeoff\_within\_12\_mths+delinq\_amnt+pub\_rec\_bankruptcies+tax\_liens
   * Numeric Predictor Pre-Processing
     1. Replaced missing numeric variables with median
     2. Centered and scaled numeric predictors to have a mean of 0 and standard deviation of 1 because we are using NNET which is sensitive to varying scales of data.
     3. Remove variables that contain only a single value.
   * Categorical Predictor Pre-Processing
     1. Replaced missing categorical variables with “unknown”
     2. Dummy encoded categories with 1s and 0s
   * Sampling
     1. Downsample the imbalanced target data to 1:1

## Model Building

1. Model specification
   * Train Random Forest Model
   * Train Neural Network Model
   * Train XG Boosting Model
2. Data partitioning
   * Split the data into 70/30 train/test split using random sampling

## Model Training

* Model variables: loan\_status ~ loan\_amnt + funded\_amnt+funded\_amnt\_inv+term+installment+grade+sub\_grade+emp\_length+home\_ownership+annual\_inc+verification\_status+issue\_d+loan\_status+pymnt\_plan+purpose+addr\_state+dti+delinq\_2yrs+fico\_range\_low+fico\_range\_high+inq\_last\_6mths+open\_acc+pub\_rec+revol\_bal+total\_acc+out\_prncp+out\_prncp\_inv+total\_rec\_late\_fee+last\_pymnt\_d+last\_pymnt\_amnt+last\_credit\_pull\_d+collections\_12\_mths\_ex\_med+policy\_code+application\_type+acc\_now\_delinq+chargeoff\_within\_12\_mths+delinq\_amnt+pub\_rec\_bankruptcies+tax\_liensDefine your Recipe
* Define your Model
  + For 3 models, I use the same model variables
  + For hyperparameters, I use manually tuning method:

XG Boosting: trees=20

Random Forest: trees=20, num.threads=8

NNet model: hidden units=10, dropout=0.1, epchs=20

* Create a workflow and Fit the model
* Evaluate metrics on Train and Test:

We will evaluate the models using Accuracy, the percentage of passengers we correctly predict as surviving. Accuracy is one metric for evaluating classification models. Informally, accuracy is the fraction of predictions our model got right. Formally, accuracy has the following definition:

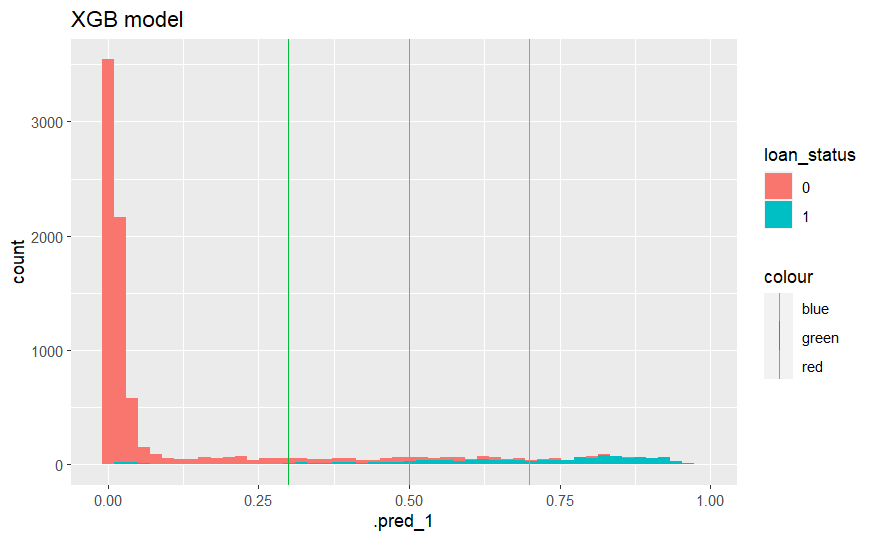
Where TP = True Positives, TN = True Negatives, FP = False Positives, and FN = False Negatives.

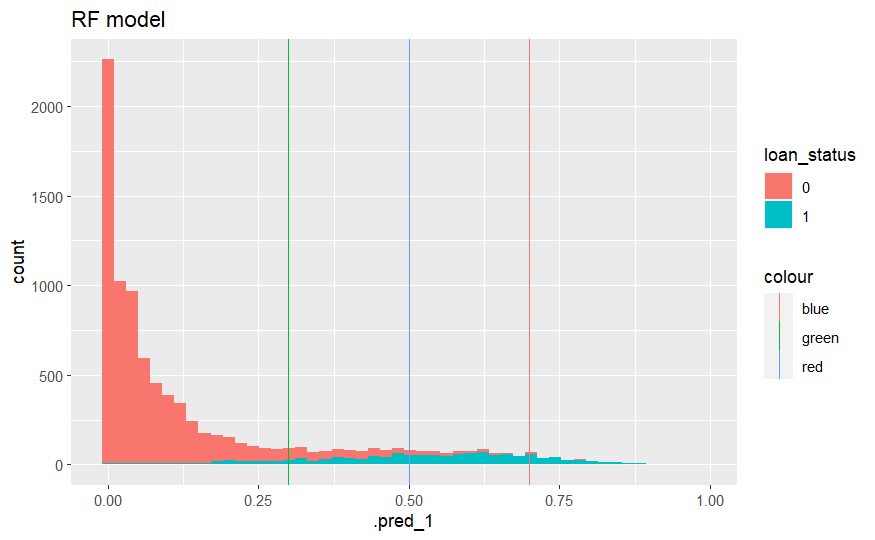
We will also use the AUC, which refers to the "Area under the ROC Curve" measuring performance across all possible classification thresholds. In other words, AUC is interpreted as the probability that the model ranks a random positive example more highly than a random negative example. It indicates the quality of the model’s predictions. The closer the AUC is to 1, the better the prediction the model has.

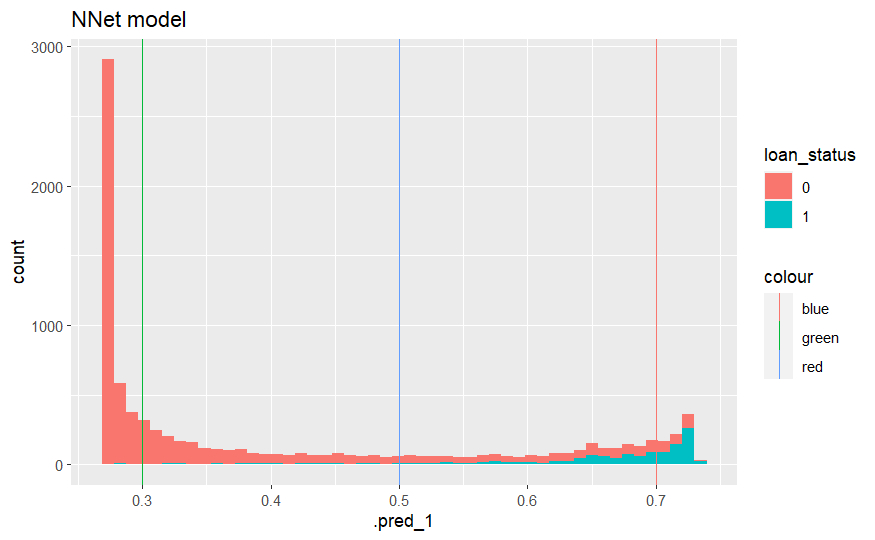
Precision describes what proportion of the positive identifications is actually correct.

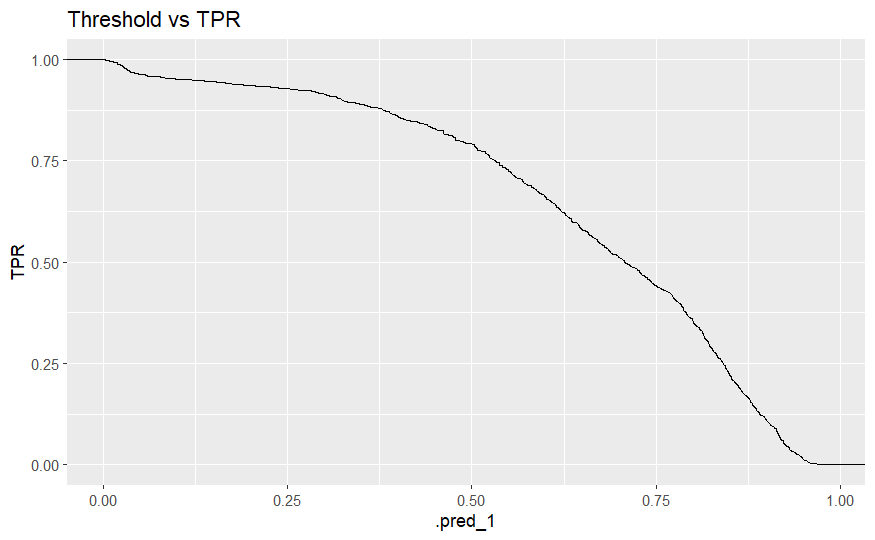
Recall describes what proportion of actual positives is actually correctly.

As the threshold increases, the TPR decreases, we would choose 0.5 as our best threshold since the TPR decreases significantly after it passed 0.5.









## Model Comparison

* Compare and contrast 3 models

XG Boosting Model works the best, because it has a much higher AUC and Accuracy than other 2 models

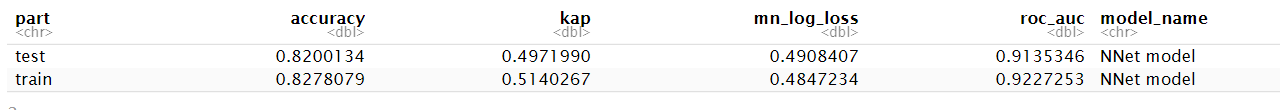
* Compare their auc/accuracy/precision / lift on training and test sets.

**AUC & Accuracy**

Random Forest



Neural Network

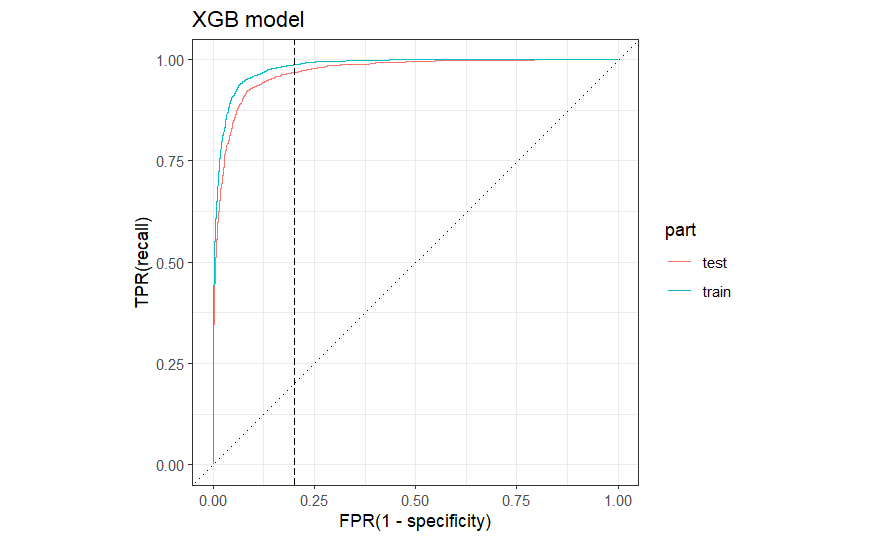


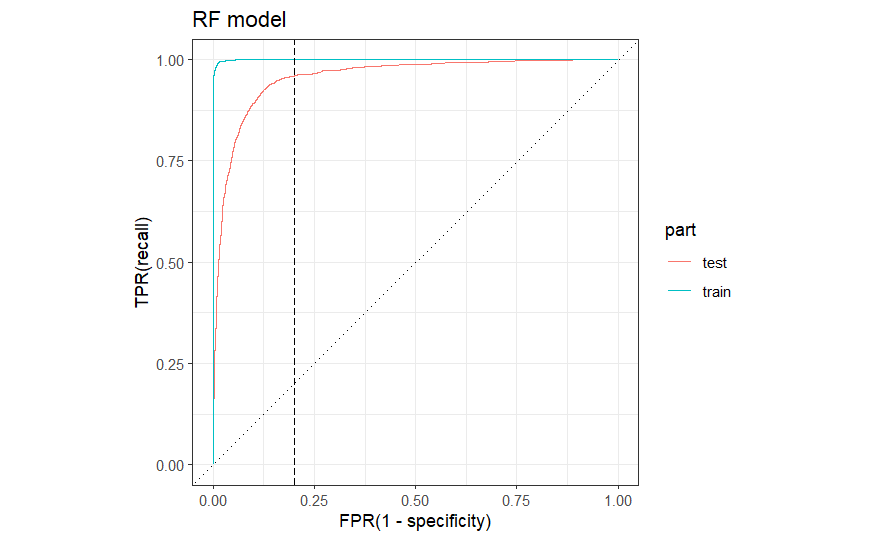
XG Boosting

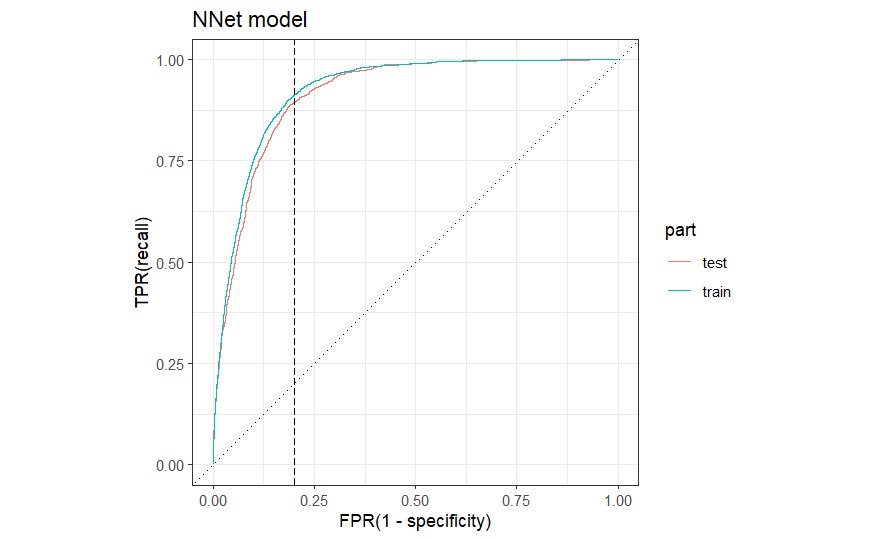


**AUC:**

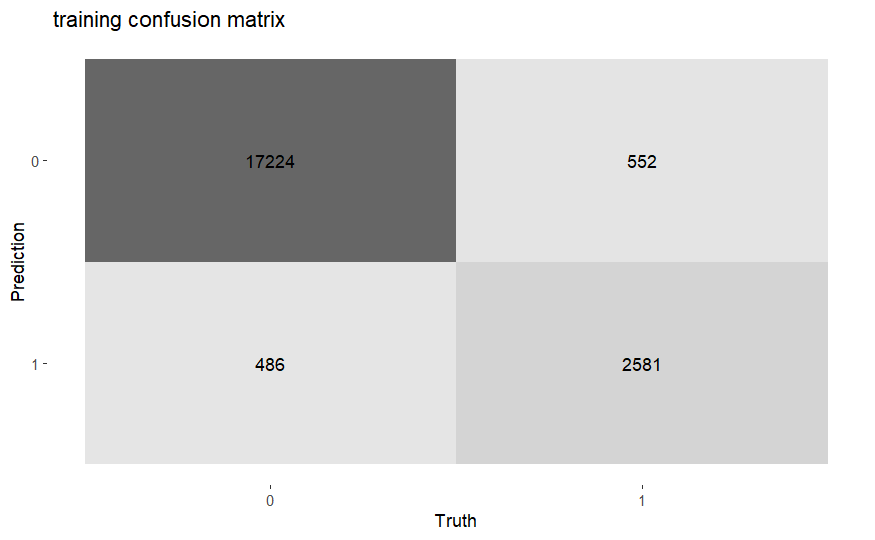
XGB reaches the highest AUC, suggesting it performs the best.

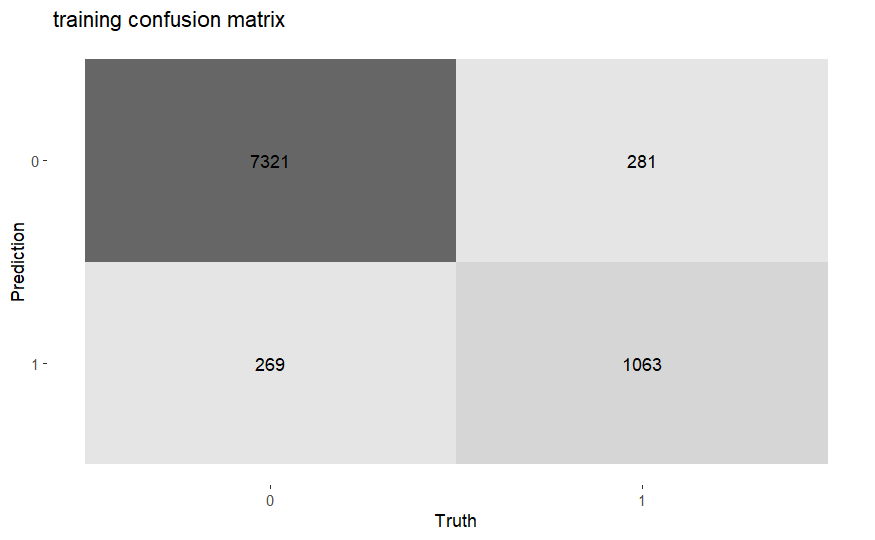






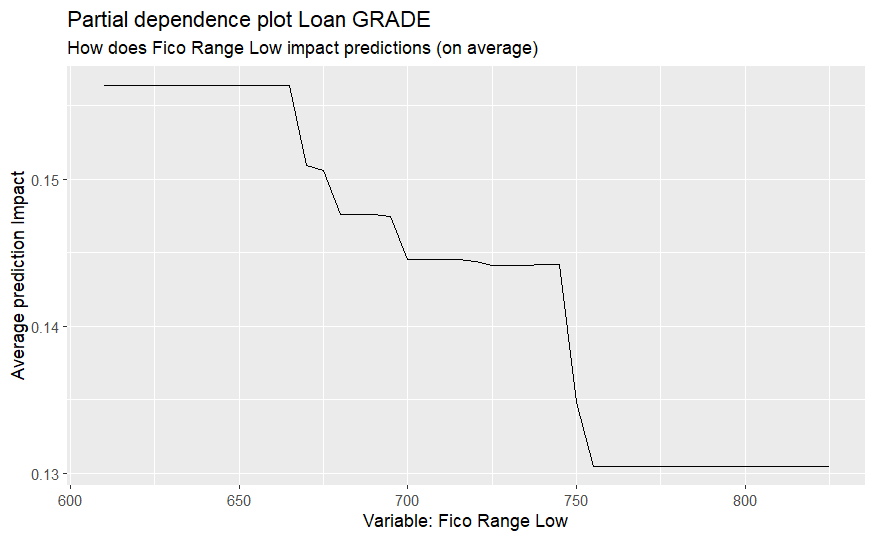
Confusion Matrics of XG Boosting Model:





## Global Variable Importance

## 

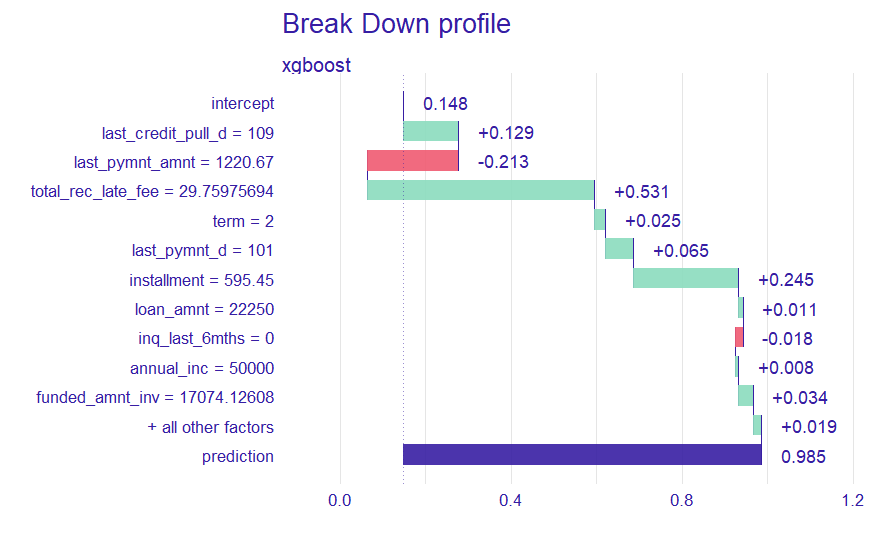


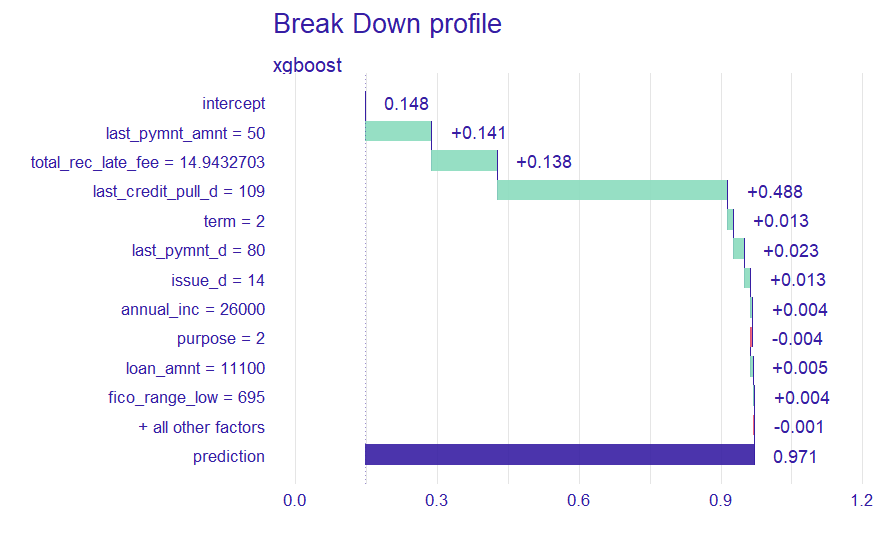
We could notice that the lower the fico range score is, the higher impact it will impact the target variable

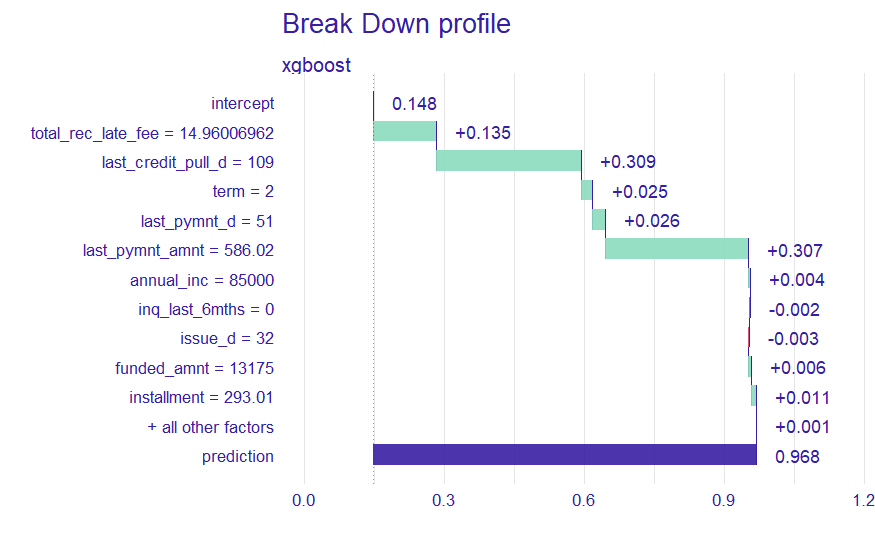
## Local Variable Importance

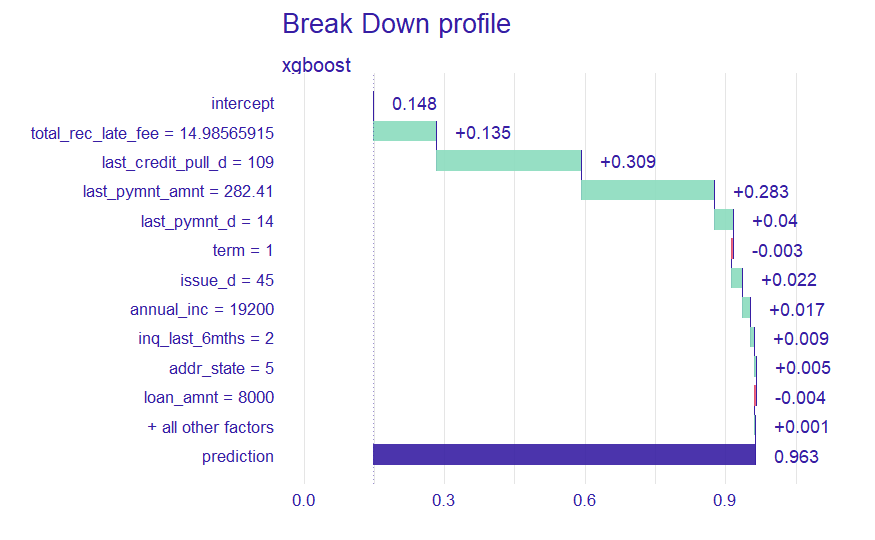
We identified top 5 true positive, false positive and false negative to look at how the combinations of variables contributed to the predictions.

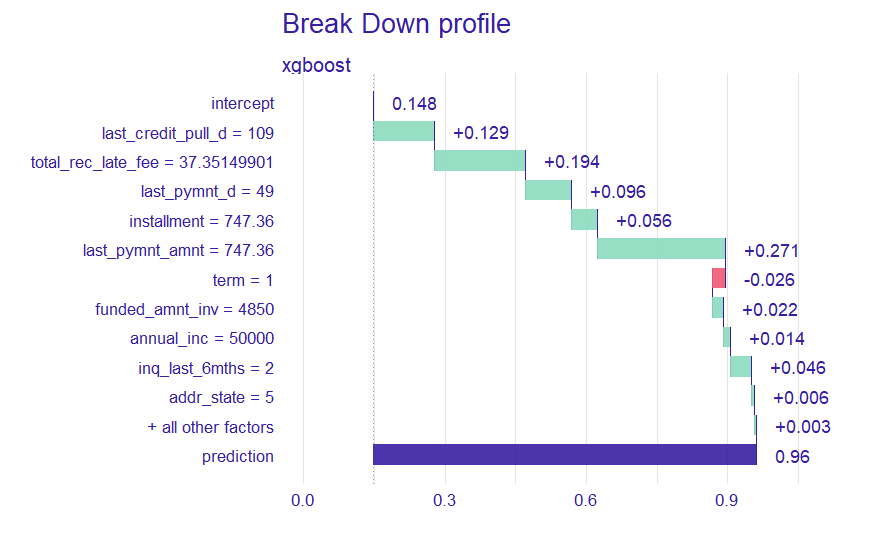
TOP 5 True Positive





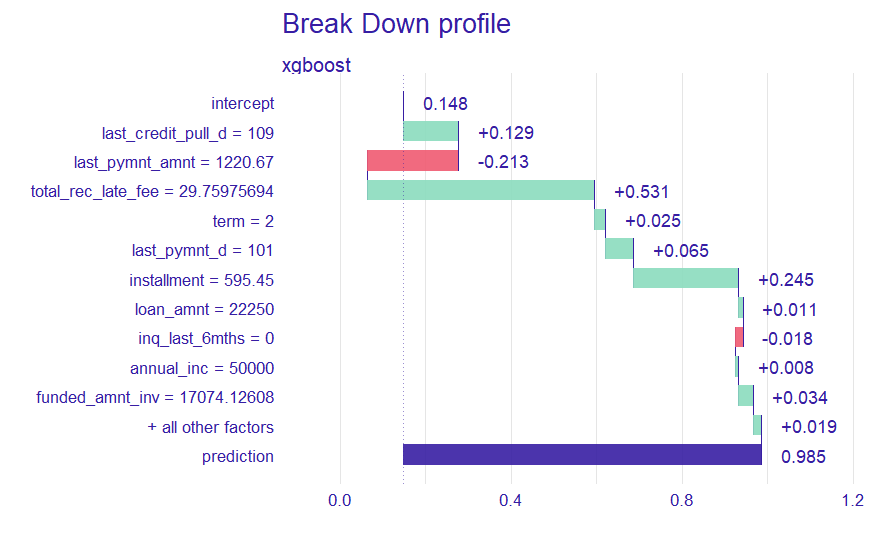


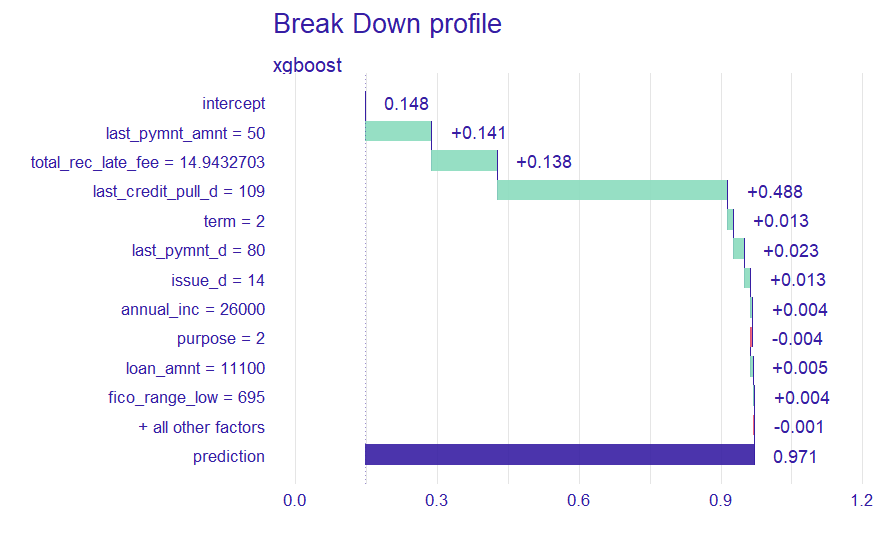


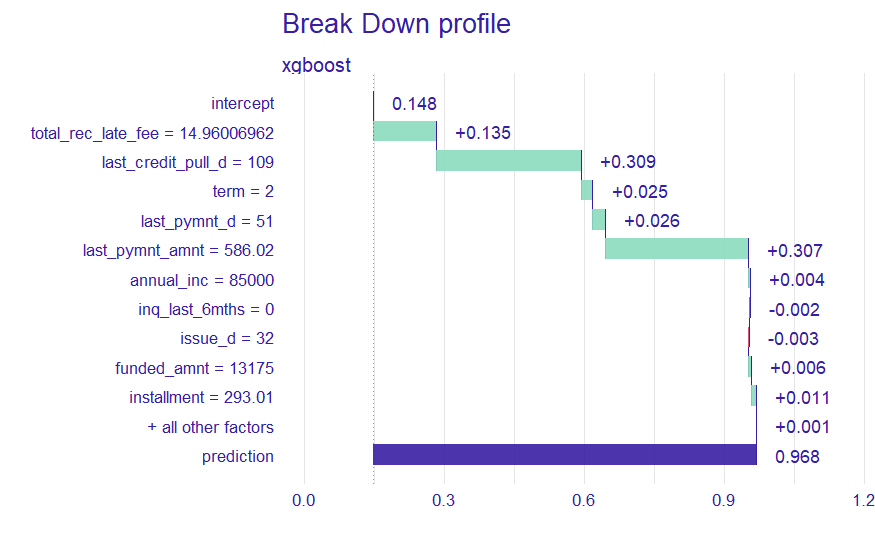


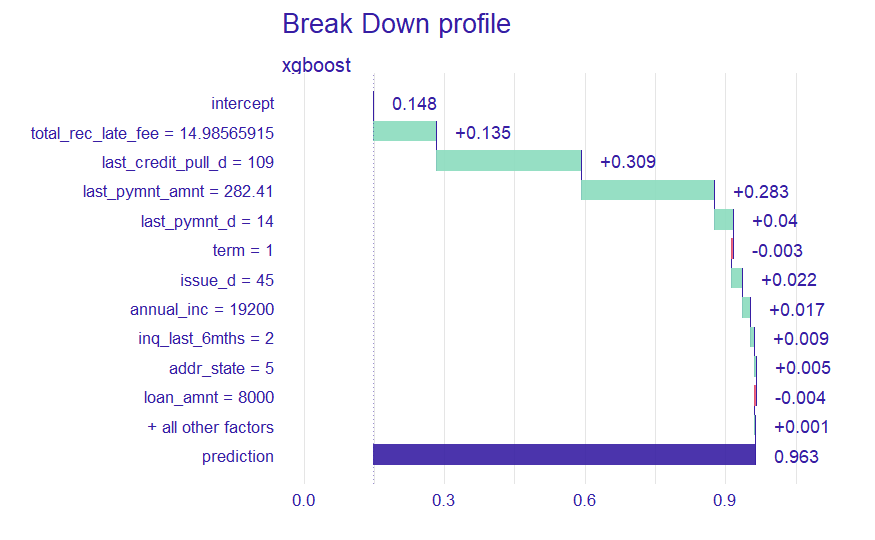
We could notice that to reach the prediction of default, based on the break down file above, last\_credit\_pull\_d around 3 months, last\_pymnt\_amnt around 290 to 747.36, and total\_rec\_late\_fee from 14.9 to 37.4 contributes to the result of default.

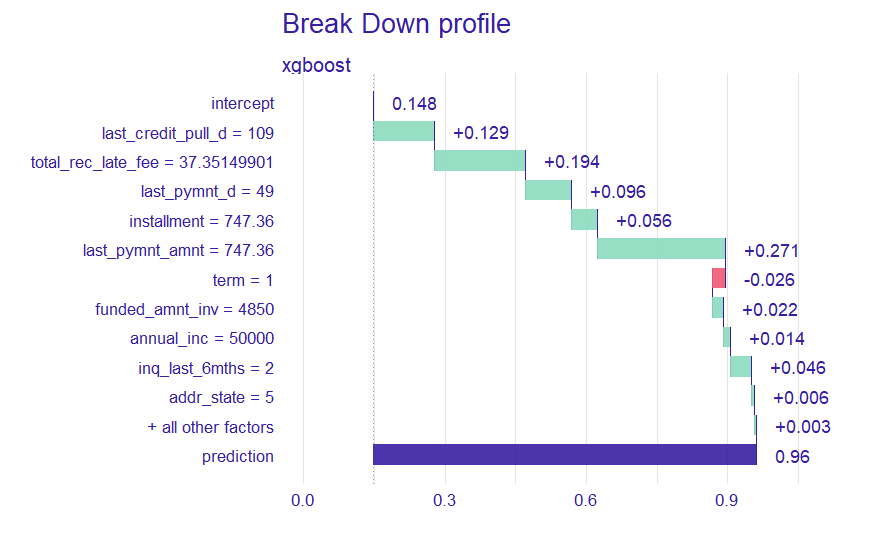
Top 5 False Positive





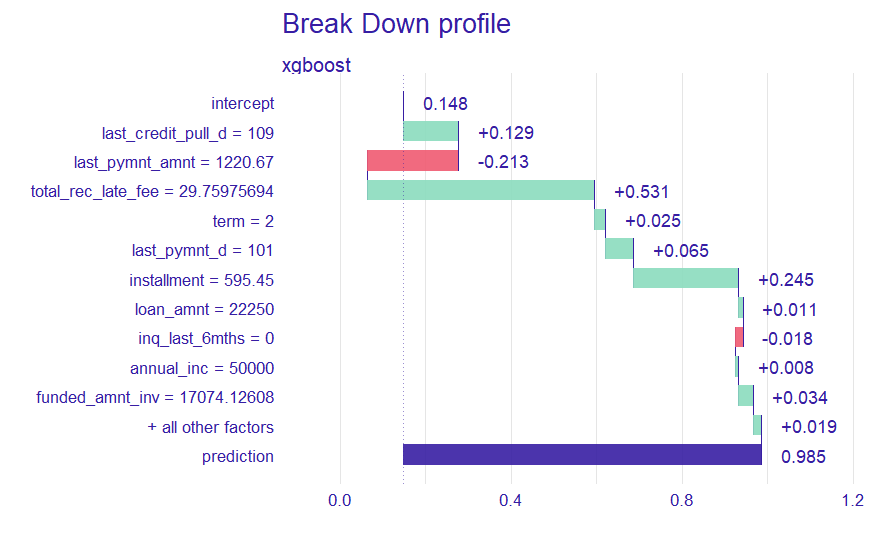


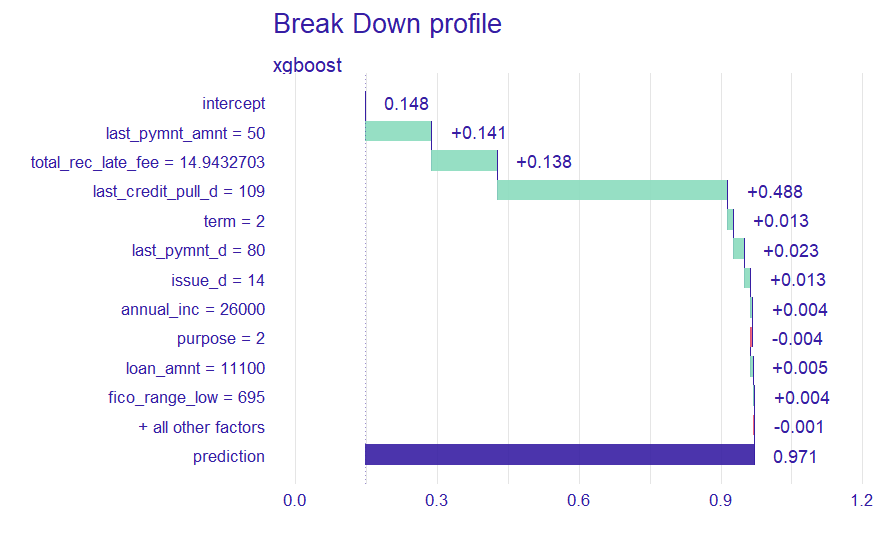


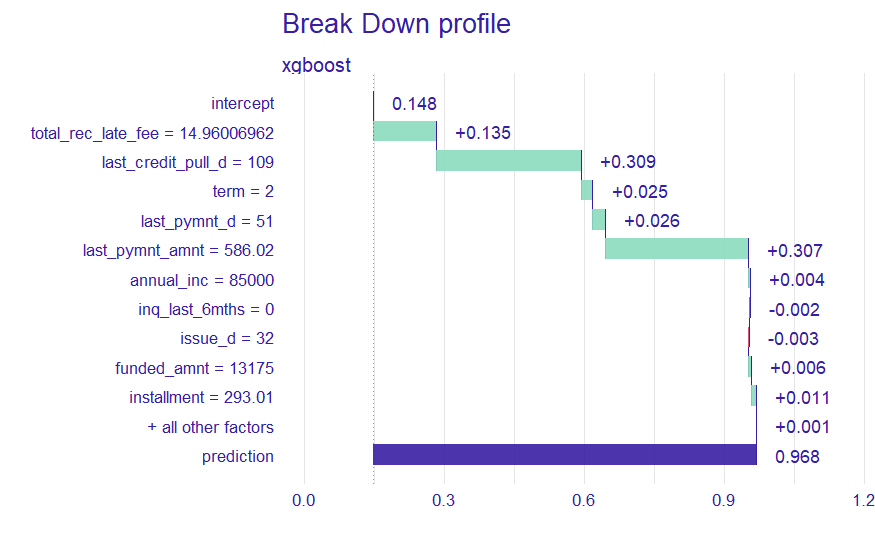


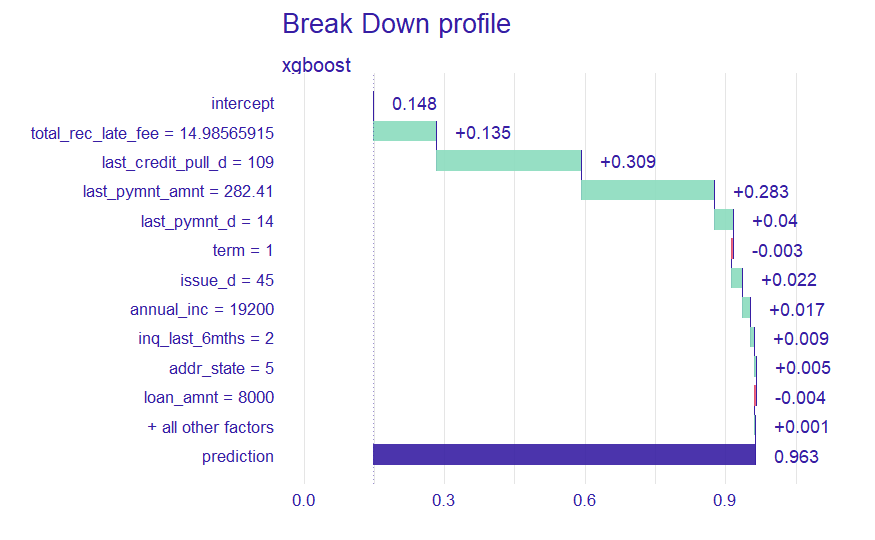
From the top 5 false positive, we could notice some variables which seems to contribute to default but actually not default. total\_rec\_late\_fee, installment, last\_credit\_pull\_d=109 and last\_rec\_late fee and last\_pymnt\_amnt are contributing to the false positive result. Also, we could notice that besides those features mentioned above affect the result, other variable do not play much part in contributing the false positive result.

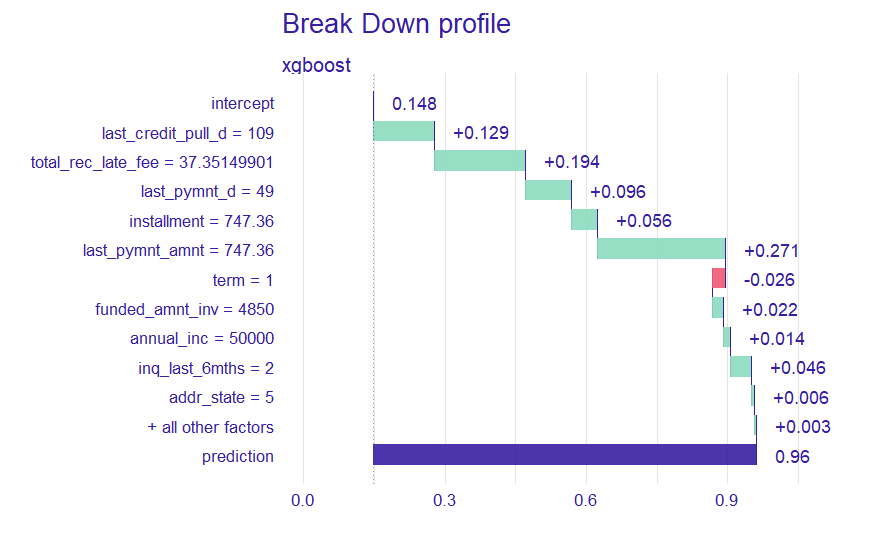
Top 5 False Negative











From the false positive result, we could see that those who have shorter months that LC pulled credit for this loan and have less last total payment amount and less late fees are less likely to default.