Executive Summary

Problem

The report is to help the financial institution to predict "loan status" is default or not and figure out the characteristic of the loans are most likely to be default. The purpose of doing this is to detect the potential default loans as many as possible. Based on the characteristic we found, we will give recommendation on how to prevent losing the commission of the loan value, which is typically around 1%-2% if the certain buyers cannot pay the bill.

Key Findings

There are several key findings on the variables influencing the loan status obviously.

1) **Having a payment plan effectively prevent loan default.** If a payment plan has been put in place for the loan, there's no default. All of default are under the situation that a payment plan has not been put in place for the loan, which is 15%
2) **As the grade increase from A to G, the possibility of default increases.** Grade A has 6% on the possibility of default, and grade G has 34% on the possibility of default, which is nearly 6 times.
3) **There are more defaults when the term is 60 months than 36 months,** which is 23% and 12% on the possibility of default respectively. It is nearly two times.
4) **The lower boundary range increase, the possibility of default decreases**. When the lower boundary range is under 650, the possibility of default is at least greater than 25%. When the lower boundary range is over 800, the possibility of default is lower than 10%.
5) **The purposes like small business and educational have a higher possibility of default**, which is 27% and 21%. The purpose like major purchase and wedding have the lowest possibility of default, which is 11% and 10%.

Model Performance Summary & Interpretation

XGboost model is the best among logistic, XGboost and Neural Network model, because it has the highest **AUC which is 95%.** That means the model can separate 95% of the loan status to default and current. What's more, XGboost model higher **precision and recall, which is 75% and 70%**. Precision measures the true default rate within detected default cases. Recall measures the model's ability to detect positive samples. They are also the important things to consider, as financial institution cares about covering as many as default rather than caring about the accuracy of the detected cases if this case amount is too small and ensure the detected default's accuracy. XGboost model shows strong capacity on detecting both enough amount and enough accuracy. In this case, I recommend selecting threshold 0.517 as nearly 70% of TPR and 4% FPR is accepted.

The top 5 variables of the model are

1) **The most recent month LC pulled credit for this loan.** When Sep-2016, it has highest impact to predictions.

2) **Last total payment amount received.** When last payment amount is under 100, it has highest impact to predictions and then decrease as the amount increase.
3) **The monthly payment owed by the borrower if the loan originates (installment).** The impact to predictions increases as the Loan installment increase.
4) **Total credit revolving balance.** The impact to predictions increases as the Loan revol balance increase.
5) **Self-reported annual income.** The impact to predictions decreases as the Loan self-reported annual income over $0, and then being stable.

There are variables more likely leaded to false positive and false negative cases. In top 10 false positive cases, when the most recent month LC pulled credit for this loan = 109, the number of payments on the loan = 2 and Last total payment amount received = 30.66, they contributed the most to the prediction. However, in false negative cases, these two variables shows the most negatively contributed to the prediction. It is reasonable as these two variables are important. Although they may lead to false positive and false negative cases, the rate of detecting wrong is acceptable.

Recommendations

1) **Set an alarming notice email or texting when customers' the total credit revolving balance when approaching $40,000.** It seems like when over $40,000, total credit revolving balance will significantly impact the default possibility of loan. It is necessary to notify the customers to pay the loan before they reach that dangerous line. And it is also helpful for the financial institution to figure out which customers are feeling frustrated on paying loans and adjust the loan plan for them.
2) **Check self-reported annual income more frequently during the loan process.** When self-reported annual income is $0, it will significantly impact the default possibility of loan. It is possible that customers may lose their income during the loan process, so they cannot pay the loan. To prevent more waste on letting owning amount be more and more, quickly realizing that is the most important thing for the financial institution. I would suggest check customers' credit, income and financial statement 2-3 times more, to make sure the current customers have at least stable cash flow to pay the loan.

MODEL REPORT

Detailed Analysis & Steps

File(s) Summary

File Name	Record count	Column count	Numeric columns	Character columns
Train.csv	29777	52	29	23
Evaluate.csv	12761	51	29	22

Field Summary

Categorical Variables

Name	Data Type	Feature Type	# missing	% missing	# unique
term	fctr	categorical	3	0.01%	2
int_rate	fctr	categorical	3	0.01%	390
grade	fctr	categorical	3	0.01%	7
sub_grade	fctr	categorical	3	0.01%	35
emp_title	fctr	Text	1817	6.10%	22143
emp_length	fctr	categorical	3	0.01%	12
home_ownership	fctr	categorical	3	0.01%	5
verification_status	fctr	categorical	3	0.01%	3
issue_d	fctr	categorical	3	0.01%	55
loan_status	fctr	Target	0	0.00%	2
pymnt_plan	fctr	categorical	3	0.01%	2
url	fctr	num	3	0.01%	29774
desc	fctr	Text	9432	31.68%	20310
purpose	fctr	categorical	3	0.01%	14
title	fctr	Text	13	0.04%	15200
zip_code	fctr	categorical	3	0.01%	819
addr_state	fctr	categorical	3	0.01%	50
earliest_cr_line	fctr	categorical	23	0.08%	516
revol_util	fctr	categorical	67	0.23%	1094
last_pymnt_d	fctr	categorical	67	0.23%	106
next_pymnt_d	fctr	categorical	27425	92.10%	96
last_credit_pull_d	fctr	categorical	5	0.02%	109
application_type	fctr	categorical	3	0.01%	1

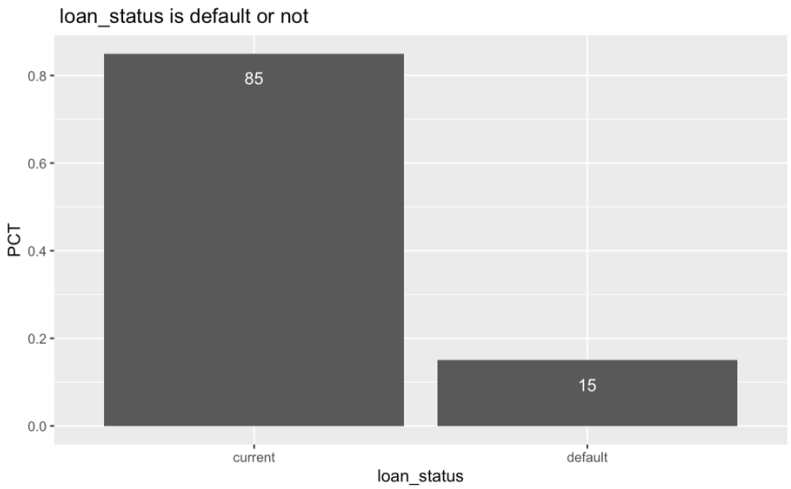
Numeric variables

Name	# missing	% missing	mean
id	3	0.01%	6.63E+05
member_id	3	0.01%	8.24E+05
loan_amnt	3	0.01%	1.11E+04
funded_amnt	3	0.01%	1.08E+04
funded_amnt_inv	3	0.01%	1.01E+04
installment	3	0.01%	3.24E+02
annual_inc	4	0.01%	6.92E+04
dti	3	0.01%	1.34E+01
delinq_2yrs	23	0.08%	1.55E-01
fico_range_low	3	0.01%	7.13E+02
fico_range_high	3	0.01%	7.17E+02
inq_last_6mths	23	0.08%	1.08E+00
mths_since_last_delinq	18907	63.50%	3.47E+01
mths_since_last_record	27208	91.37%	5.92E+01
open_acc	23	0.08%	9.34E+00
pub_rec	23	0.08%	5.85E-02
revol_bal	3	0.01%	1.43E+04
total_acc	23	0.08%	2.21E+01
out_prncp	3	0.01%	1.18E+01
out_prncp_inv	3	0.01%	1.18E+01
total_rec_late_fee	3	0.01%	1.50E+00
last_pymnt_amnt	3	0.01%	2.62E+03
collections_12_mths_ex_med	104	0.35%	0.00E+00
policy_code	3	0.01%	1.00E+00
acc_now_delinq	23	0.08%	1.34E-04
chargeoff_within_12_mths	104	0.35%	0.00E+00
delinq_amnt	23	0.08%	2.04E-01
pub_rec_bankruptcies	966	3.24%	4.53E-02
tax_liens	79	0.27%	3.37E-05

Target Summary

As shown below, there are 15% of loan status is default and 85% is current. The default accuracy is 15%.

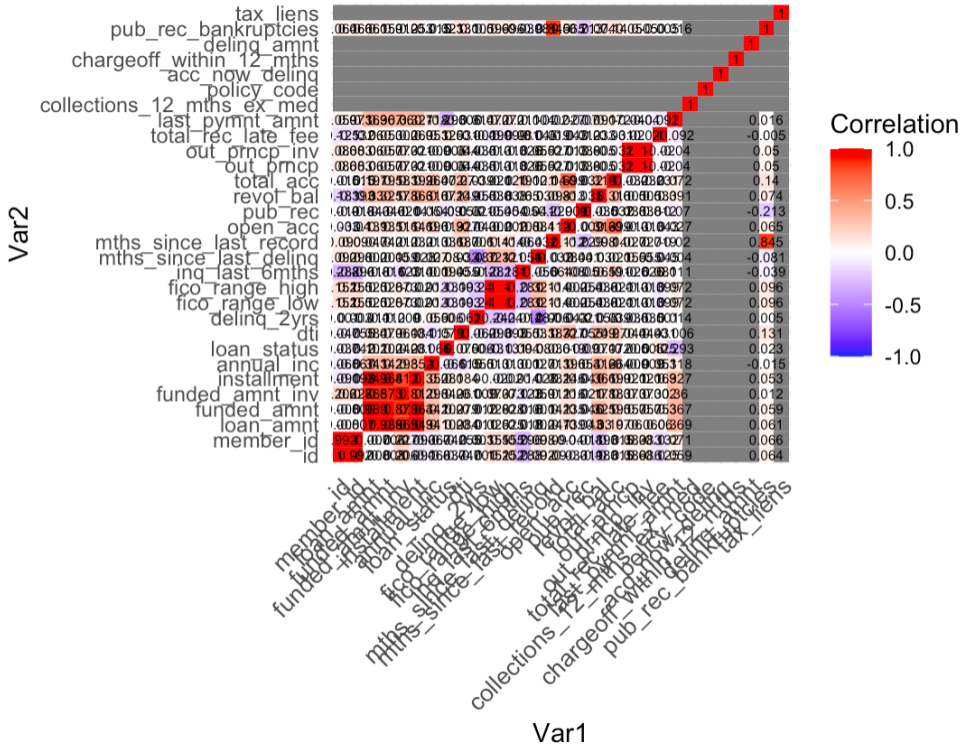
loan_status	n	pct
current	25300	0.8496491
default	4477	0.1503509



Exploratory Data Analysis & Screening

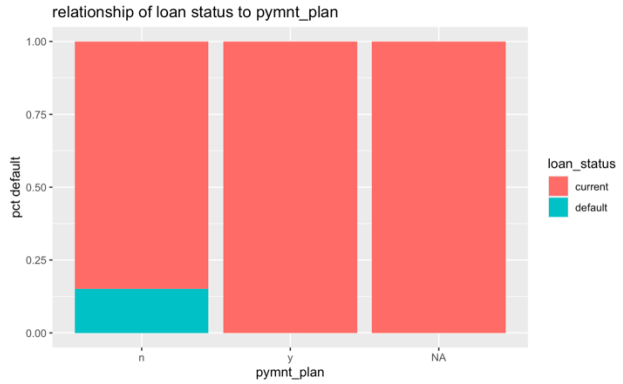
Correlation Analysis

Installment, the total amount committed by investors for that loan at that point in time, The total amount committed to that loan at that point in time and the listed amount of the loan applied for by the borrower are highly correlated. Last payment method is the most obvious negatively correlated to the loan status.

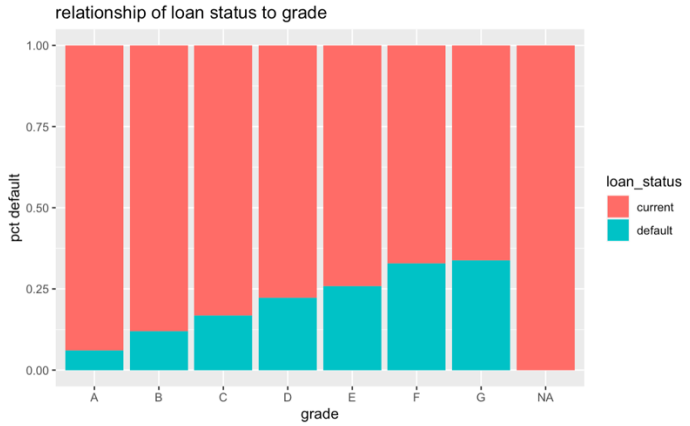


Initial Screening & Exploration

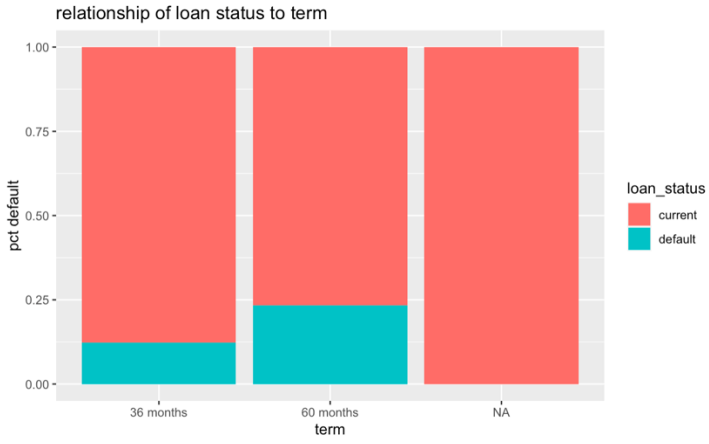
The chart below shows that if a payment plan has been put in place for the loan, there's no default. All of default are under the situation that a payment plan has not been put in place for the loan, which is 15%.



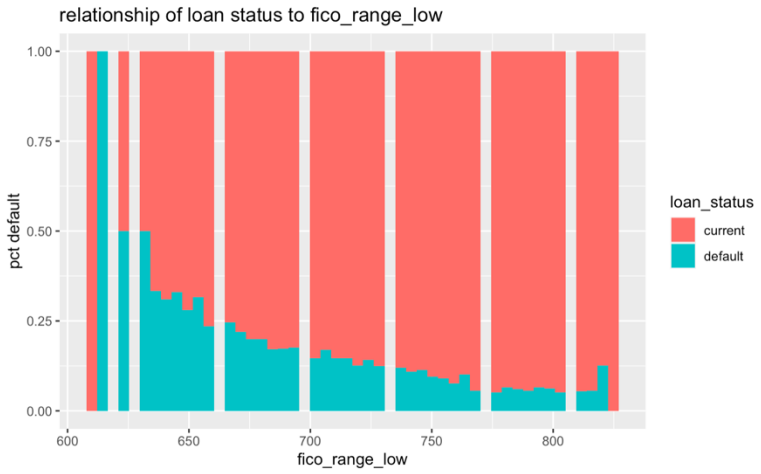
The chart below shows that it has a clear linear relationship between grade and loan status. As the grade increase from A to G, the possibility of default increases. A has 6% on the possibility of default, and G has 34% on the possibility of default, which is nearly 6 times.



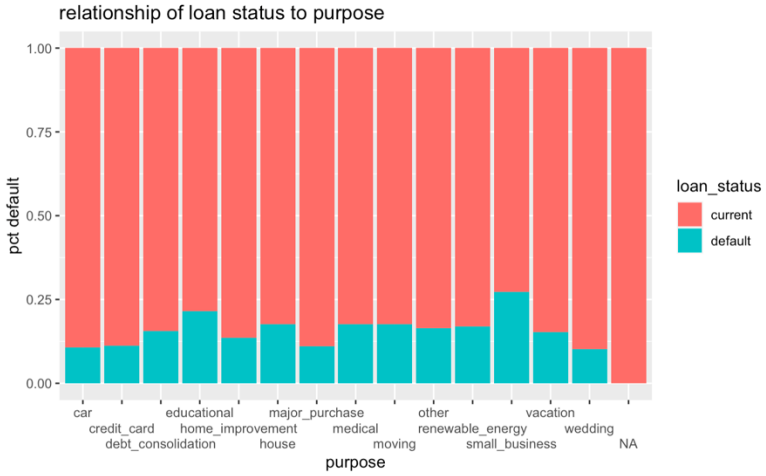
The chart below shows that there are more defaults when the term is 60 months than 36 months, which is 23% and 12% on the possibility of default respectively. It is nearly two times.



The chart below has a clear linear relationship between the lower boundary range the borrower,Äôs FICO at loan origination belongs to and loan status. The lower boundary range increase, the possibility of default decreases. When the lower boundary range is under 650, the possibility of default is at least greater than 25%. When the lower boundary range is over 800, the possibility of default is lower than 10%.

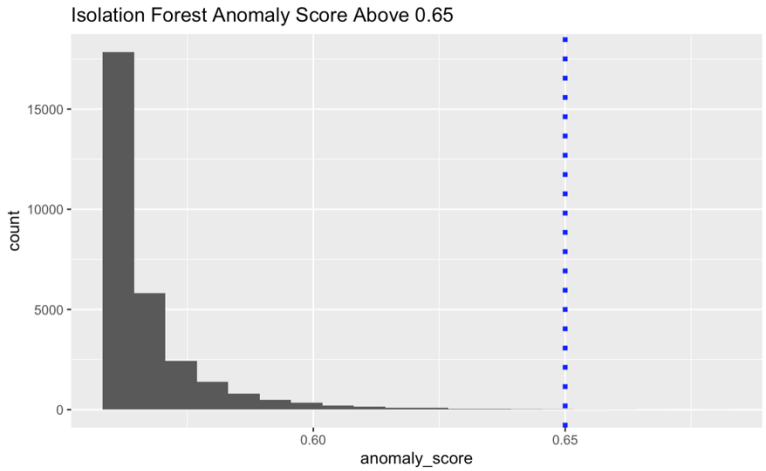


The chart below shows that the purposes like small business and educational have a higher possibility of default, which is 27% and 21%. The purpose like major purchase and wedding have the lowest possibility of default, which is 11% and 10%.

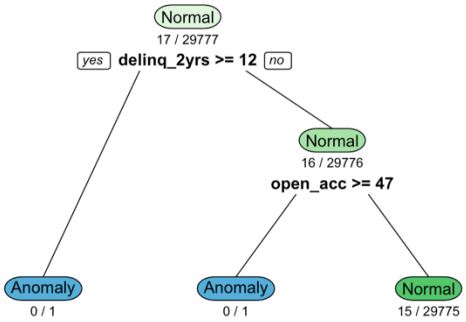


Anomaly detection

My anomaly point is 0.65.



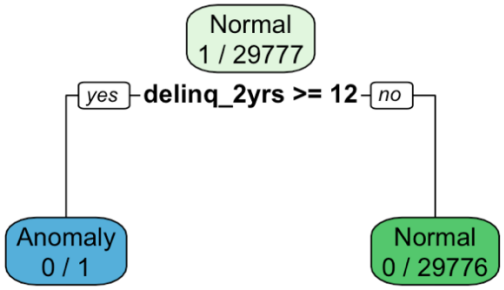
Global anomaly rules



	rule <chr>	cover <chr>
2	IF delinq_2yrs >= 12	0%
6	IF delinq_2yrs < 12 & open_acc >= 47	0%

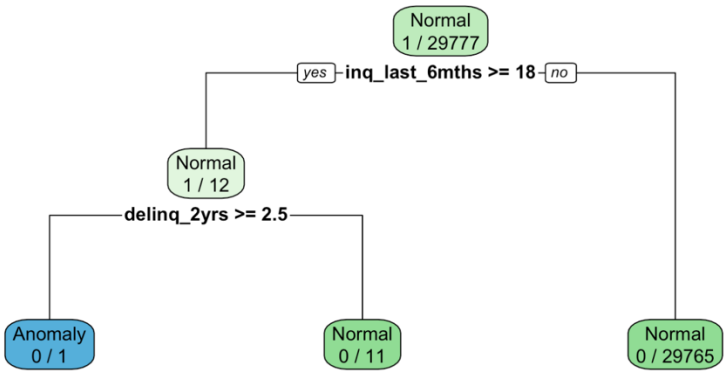
Top 5 anomaly rules

1)



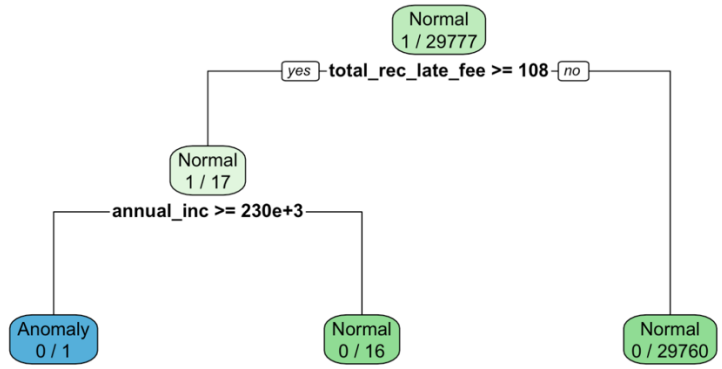
	rule <chr>	cover <chr>
2	IF delinq_2yrs >= 12	0%

2)



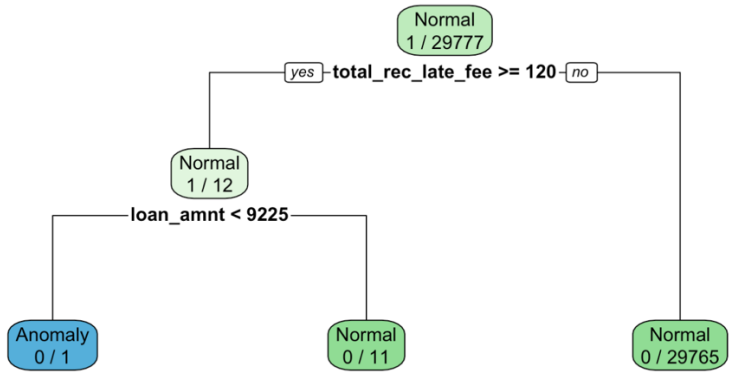
	rule <chr>	cover <chr>
4	IF inq_last_6mths >= 18 & delinq_2yrs >= 2.5	0%

3)



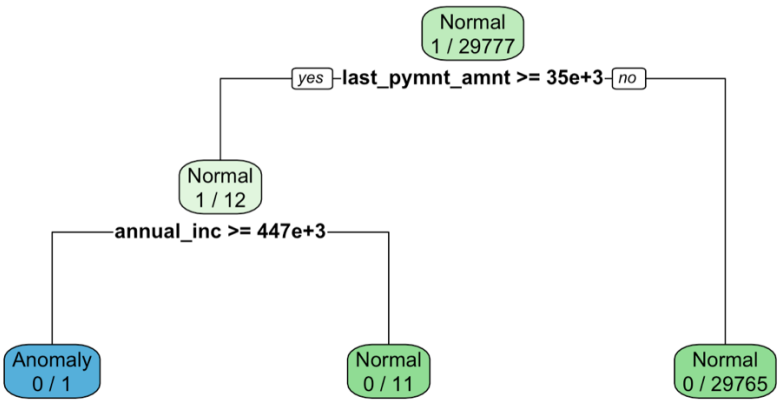
	rule <chr>	cover <chr>
4	IF total_rec_late_fee >= 108 & annual_inc >= 229750	0%

4)



	rule <chr>	cover <chr>
4	IF total_rec_late_fee >= 120 & loan_amnt < 9225	0%

5)



	rule <chr>	cover <chr>
4	IF last_pymnt_amnt >= 34537 & annual_inc >= 446500	0%

Data Preparation & Transformation

The target variable and all characters have been transformed to factor for using in the model. The recipe has included all the variables except identifiers like id, member_id, emp_title, url, desc and title. Variables like next_pymnt_d, mths_since_last_delinq and mths_since_last_record are also removed because of over 20% missing value. All missing in numeric predictors is imputed by median and all missing in numeric predictors are imputed by mode. All numeric data are centered because the models like neural network and XGBoost need normalized data. All categorical variables are dummied into 1 or 0 for using in the model. All variables that are highly sparse and unbalanced will be removed.

Model Building

You will always need to build two (or more models) for comparison purposes. You will always need to partition your data in some form either a Train / Test split or use K-Fold cross validation.

1. Data partitioning
    - Split the data into 70/30 train/test split using random sampling
    - K-Fold Split to 5
2. Data preprocessing
    - Formula
        i. survived ~ full model except id, member_id, emp_title, url, desc, title, next_pymnt_d, mths_since_last_delinq and mths_since_last_record
    - Numeric Predictor Pre-Processing
        i. Replaced missing numeric variables with median

ii. Centered numeric predictors to have a mean of 0 and standard deviation of 1.
iii. Removed variables that are highly sparse and unbalanced

- Categorical Predictor Pre-Processing
  i. Replaced missing categorical variables with mode
  ii. Dummy encoded categories with 1s and 0s
  iii. Removed variables that are highly sparse and unbalanced

Data preprocessing

- Formula for Recipe
  - survived ~ full model except id, member_id, emp_title, url, desc, title, next_pymnt_d, mths_since_last_delinq and mths_since_last_record
- Numeric Predictor Pre-Processing
  - Replaced missing numeric variables with median
  - Centered numeric predictors to have a mean of 0 and standard deviation of 1.
  - Removed variables that are highly sparse and unbalanced
- Categorical Predictor Pre-Processing
  - Replaced missing categorical variables with mode
  - Dummy encoded categories with 1s and 0s
  - Removed variables that are highly sparse and unbalanced

Model fitting and Hyper parameters

1. **Logistic Regression:** Set engine("glm") and fit with baked train data
2. **XGBoost:** Set engine("xgboost") and "classification" mode. Tune model with kfold_splits =5, initial = 5, iter = 60, control_bayes (no_improve = 5), get the model that has highest AUC with hyper parameters including trees=1276, learn_rate = 0.05953159, tree_depth = 7
3. **Neural network model:** Scale train and test. Set engine("nnet") and "classification" mode. Tune model with kfold_splits =5, initial = 5, iter = 60, control_bayes (no_improve = 5), get the model that has highest AUC with hyper parameters including hidden_units = 2, penalty= 0.9866663, epochs = 993

Evaluate metrics on Train and Test:

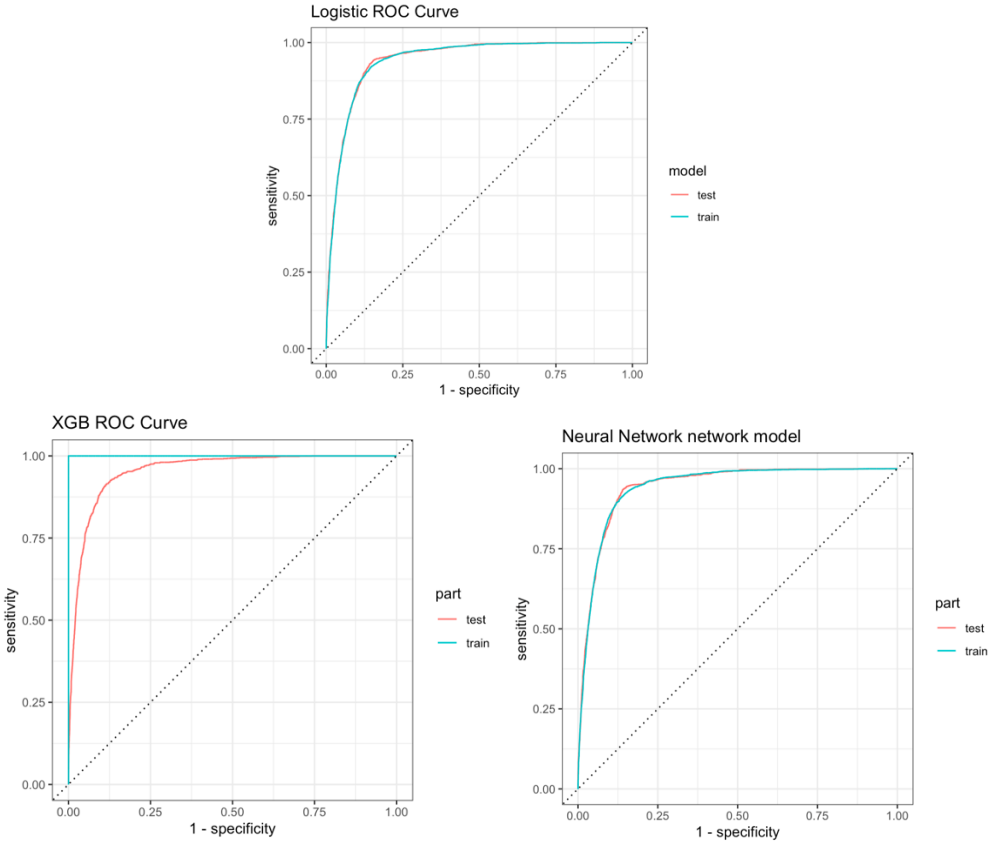
- Metrics:

		accuracy	mn_log_loss	roc_auc	precision	recall
logistic	train	0.90	0.21	0.94	0.69	0.64

	test	0.90	0.21	0.94	0.69	0.65
XGBoost	train	1.00	0.02	1.00	1.00	1.00
	test	0.92	0.21	0.95	0.75	0.70
NNT	train	0.90	0.41	0.94	0.69	0.66
	test	0.90	0.41	0.94	0.68	0.66

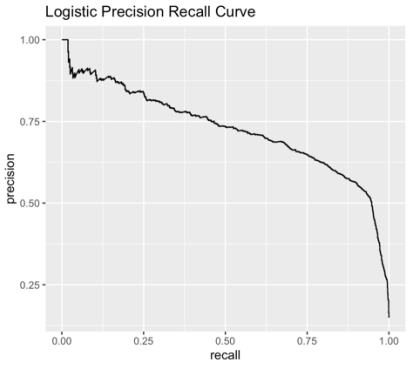
XGB is better because it has higher AUC as 95% and higher precision and recall

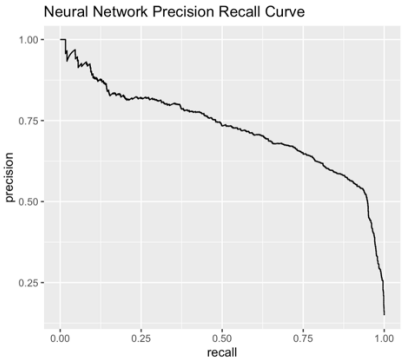
- ROC chart



XGB AUC curve is curved the most, which means it has the greatest ability to separate the classes.

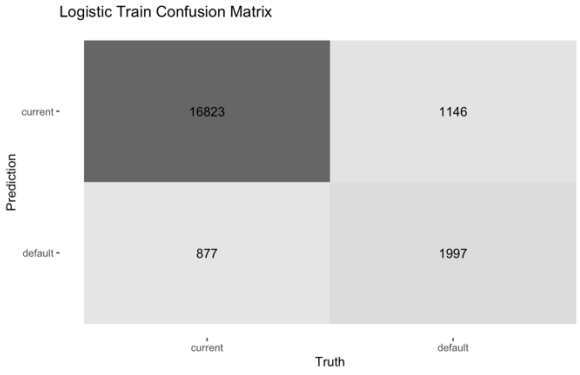
- Precision & Recall chart

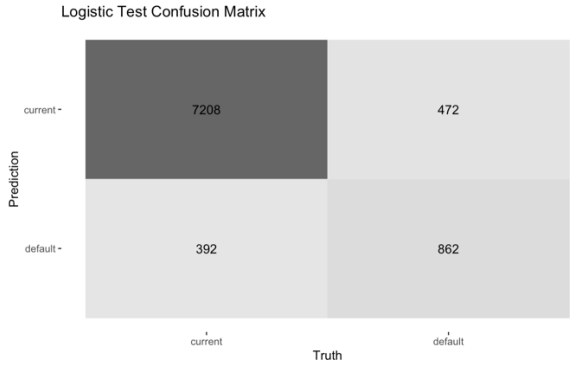


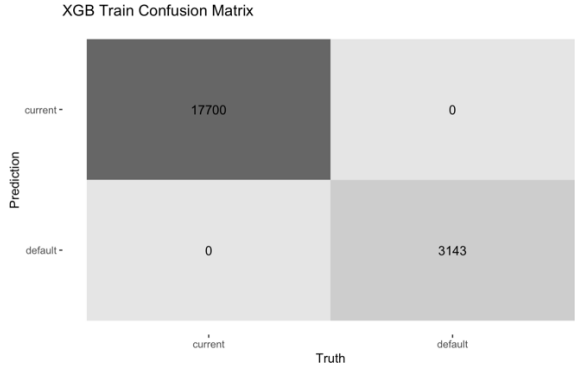


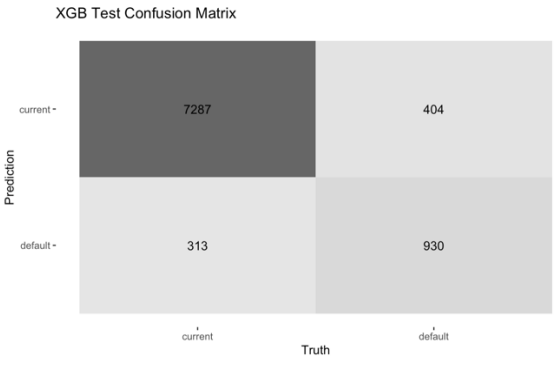
When precision increases, recall goes down. XGB model is the best because it has the point that let precision and recall both higher than 0.754. Others cannot.

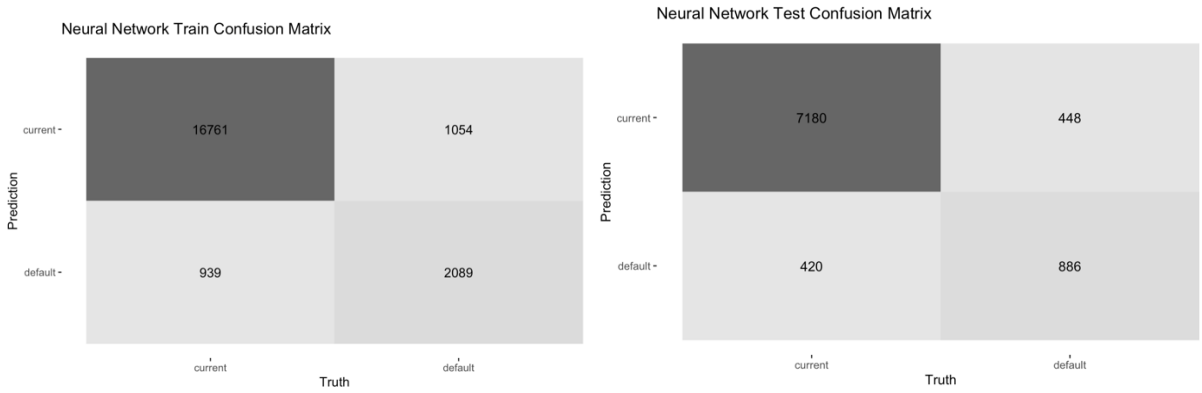
- Confusion matrix comparing train and test











XGB model is the best because its test set FPR is the lowest, which is 313, and FNR is the lowest as well, which is 404.

- Table of FPR/TPR/Precision and Score threshold
  - Logistic

fpr <dbl>	threshold <dbl>	tpr <dbl>
0.00	Inf	0.09294306
0.01	0.730	0.25683088
0.02	0.652	0.38767241
0.03	0.598	0.48751832
0.04	0.549	0.57281283
0.05	0.502	0.64462112
0.06	0.467	0.69657600
0.07	0.435	0.74070714
0.08	0.404	0.78095200
0.09	0.375	0.81335455

I recommend selecting threshold 0.435 as over 70% of TPR and 7% FPR is accepted.

  - XGB

fpr <dbl>	threshold <dbl>	tpr <dbl>
0.00	Inf	0.1198291
0.01	0.895	0.3249904
0.02	0.769	0.4873150
0.03	0.641	0.6069141
0.04	0.517	0.6850566
0.05	0.411	0.7471313
0.06	0.316	0.7917563
0.07	0.258	0.8198039
0.08	0.208	0.8441786
0.09	0.158	0.8686364

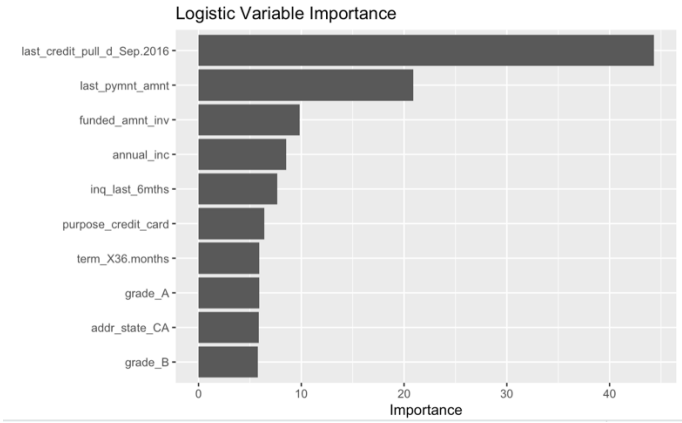
I recommend selecting threshold 0.517 as nearly 70% of TPR and 4% FPR is accepted.

  - Neural Network

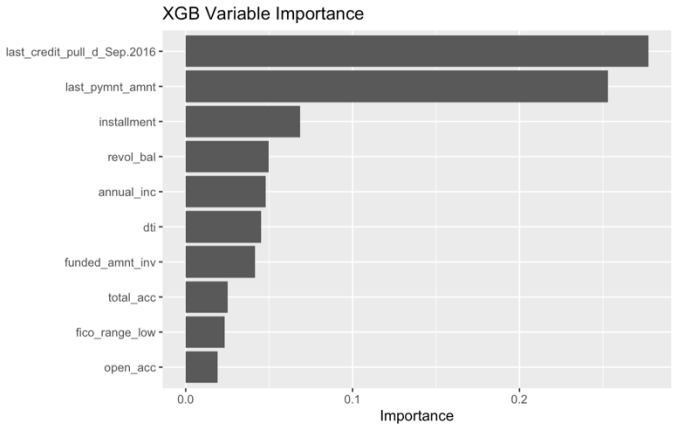
fpr <dbl>	threshold <dbl>	tpr <dbl>
0.00	Inf	0.07910084
0.01	0.620	0.24510123
0.02	0.584	0.39825862
0.03	0.560	0.49343646
0.04	0.535	0.57435829
0.05	0.512	0.63887500
0.06	0.489	0.69767273
0.07	0.471	0.74658915
0.08	0.453	0.78258974
0.09	0.437	0.81107407

I recommend selecting threshold 0.489 as nearly 70% of TPR and 6% FPR is accepted.

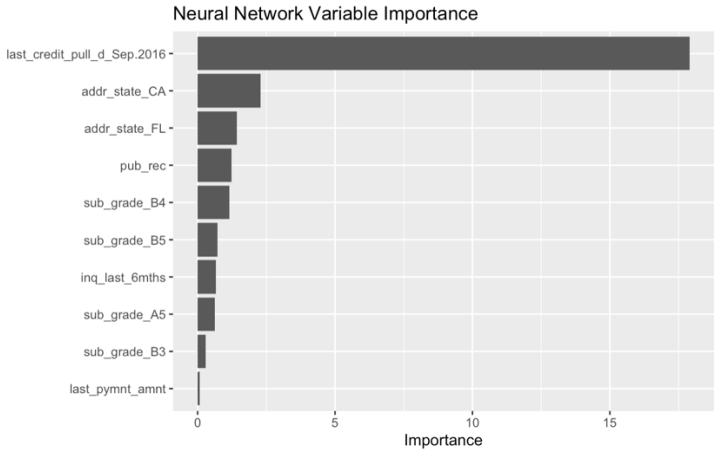
- Variable importance



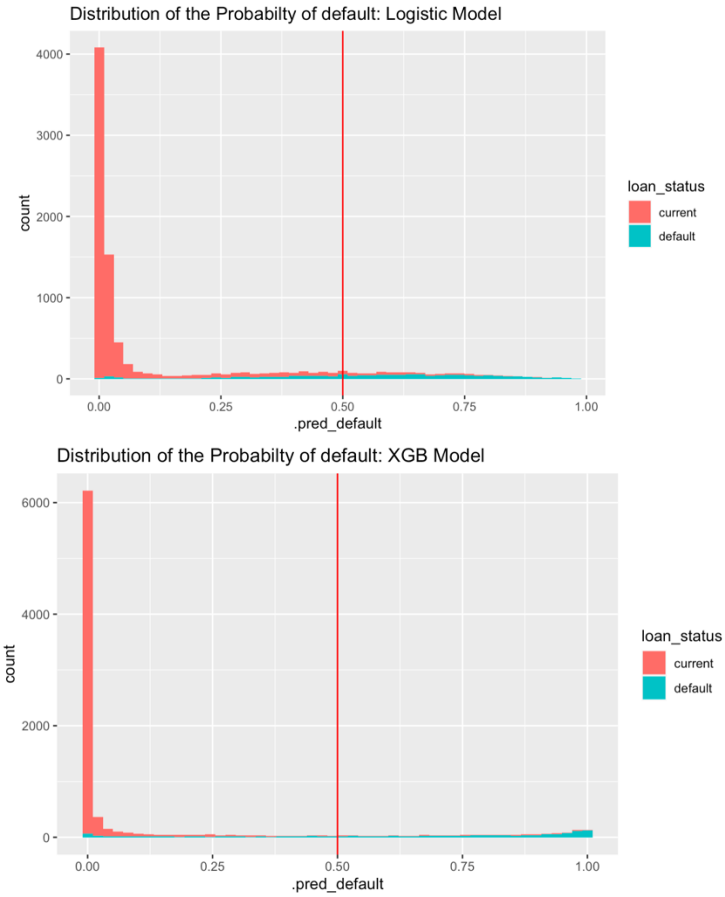
Top 5 variables are last_credit_pull_d, last_pymnt_amnt, installment, revol_bal and annual_inc.

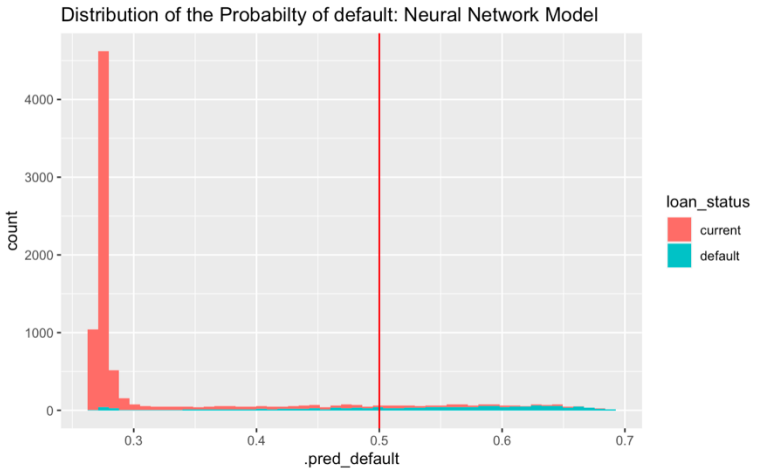


Top 5 variables are last_credit_pull_d, addr_state, pub_rec, sub_grade and inq_last_6mths.



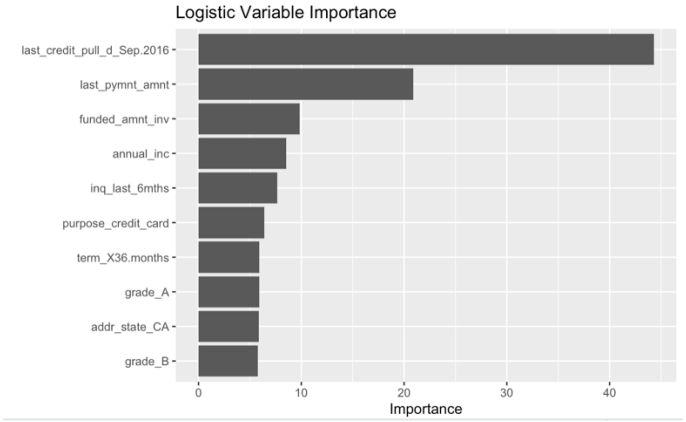
- Score distribution for test dataset





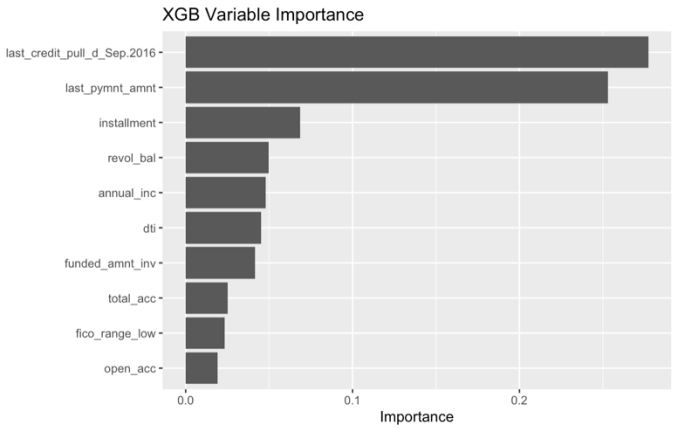
XGB model separates the loan status the best as most of the blue of default are located in 1 of .pred_default.

- Global explanations models and Top 5 partial dependence plot
  - Logistic

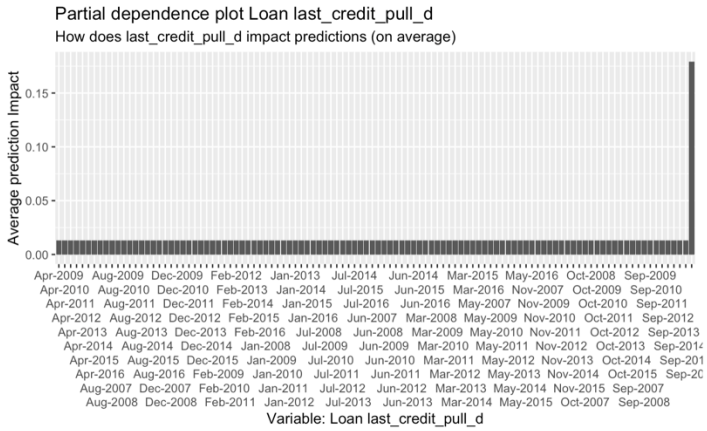


    - Because I did not use workflow() in logistic regression, so the partial dependence plot model_profile() is not available in here

  - XGB

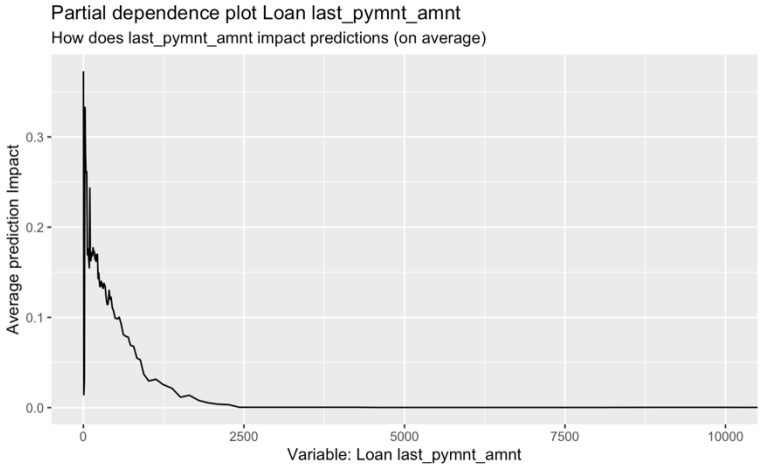


    - Last_credit_pull_d



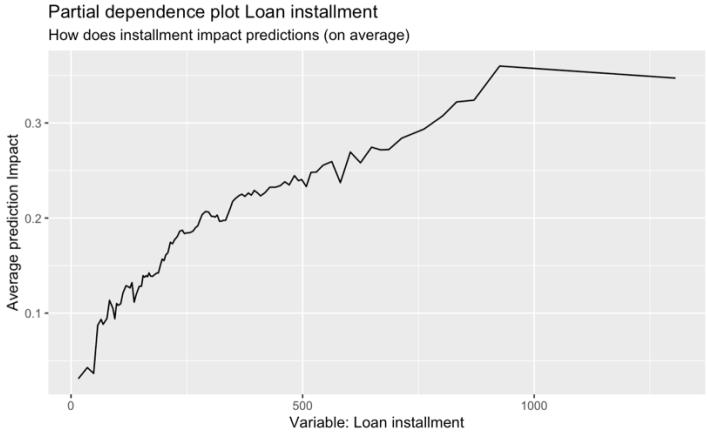
When Sep-2016, it has highest impact to predictions

- Last_pymnt_amnt



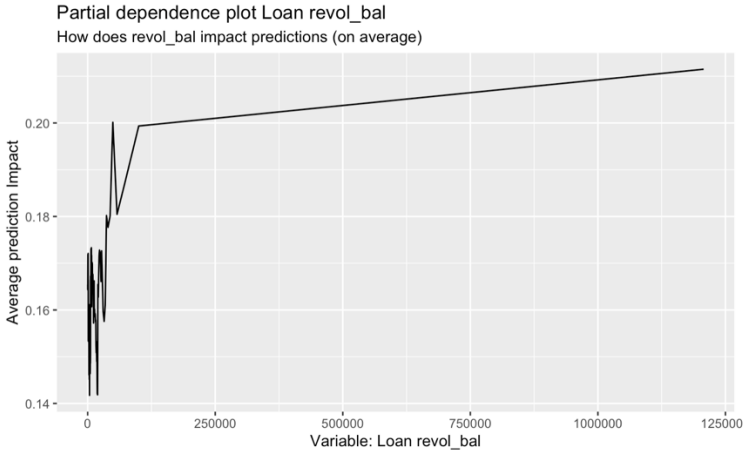
When last payment amount is under 100, it has highest impact to predictions and then decrease as the amount increase.

- Installment



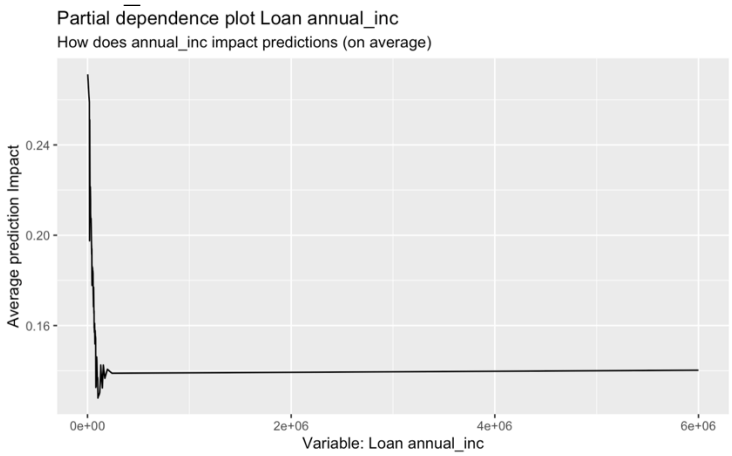
The impact to predictions increases as the Loan installment increase.

- Revol_bal



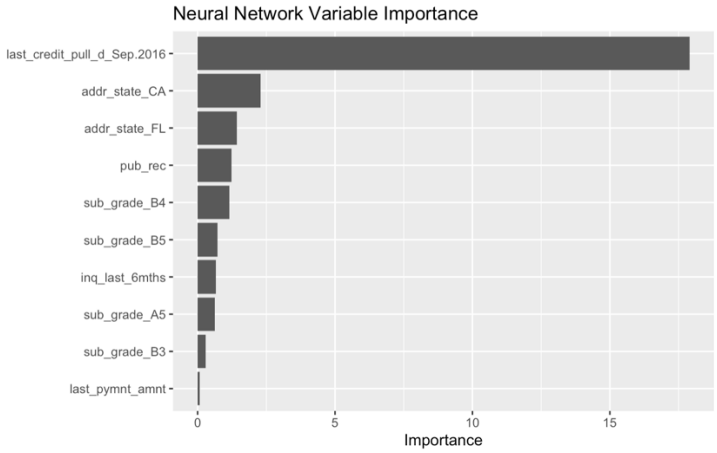
The impact to predictions roughly increases as the Loan revol balance increase.

- Annual_inc

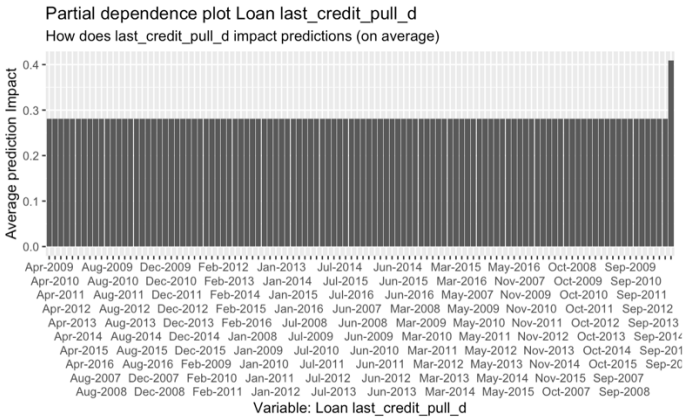


The impact to predictions decreases as the Loan self-reported annual income over $0, and then being stable.

- Neural Network

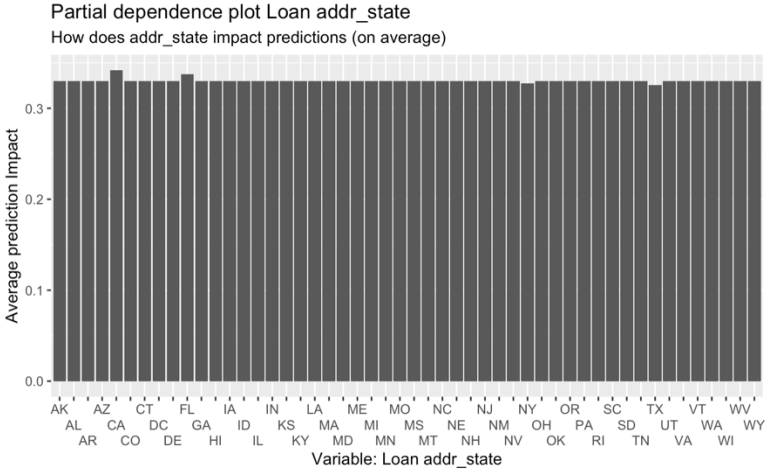


- Last_credit_pull_d



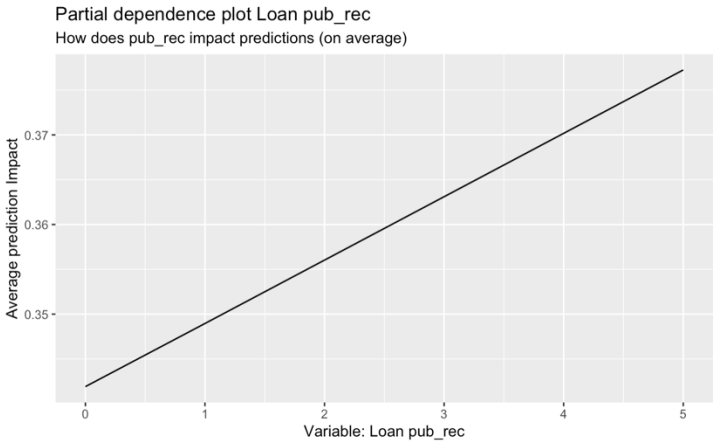
When Sep-2016, it has highest impact to predictions

- Addr_state



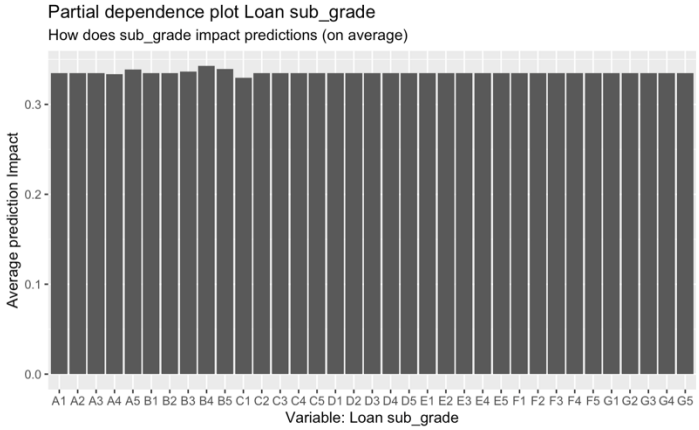
When state is AZ and GA, it has a little bit higher impact to predictions

- Pub_rec



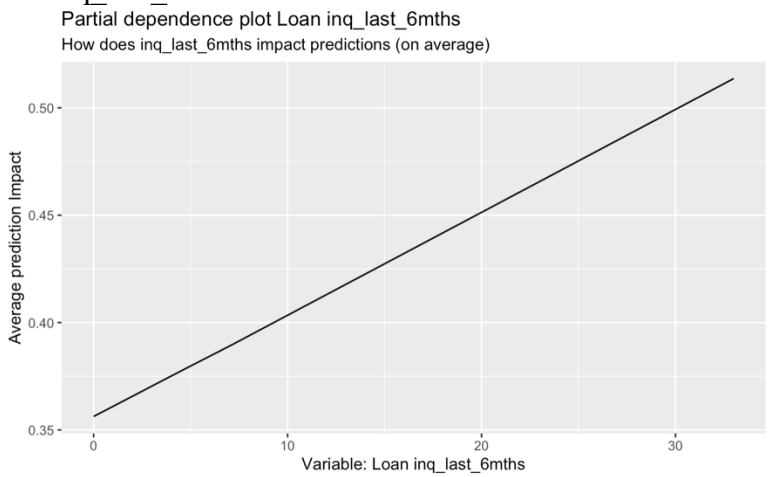
The impact to predictions increases as the Number of derogatory public records increase.

- sub_grade



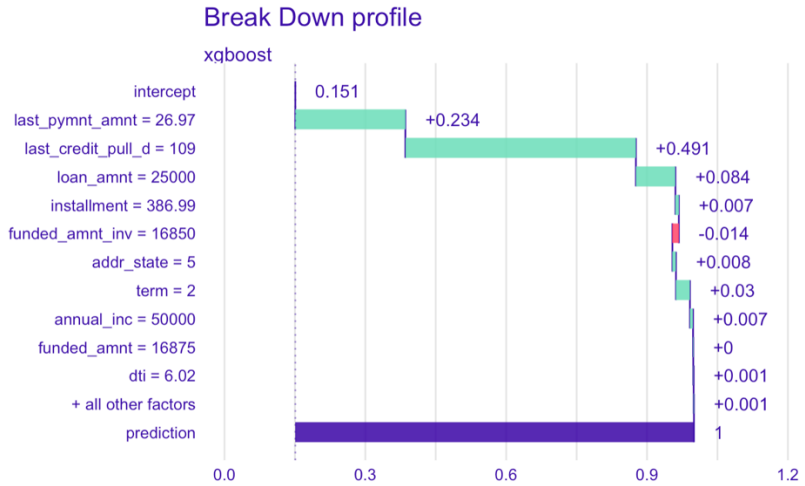
When sub grade is B4, B5 and A5, it has a little bit higher impact to predictions

- inq_last_6mths



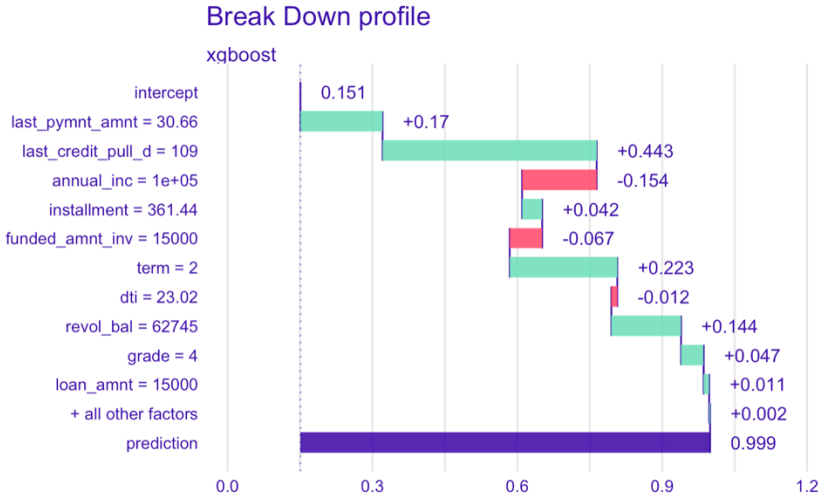
The impact to predictions increases as the number of inquiries in past 6 months increase.

- Local Explanations for best model
  - TP – top 10 true positives, loan default = 1 and ordered by pred_1 score DECENDING



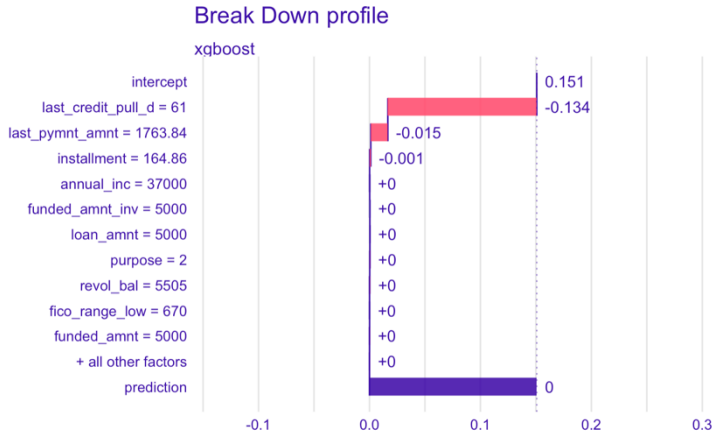
Except when the total amount committed by investors for that loan at that point in time is 16850 it negatively contributed to the True Positive prediction, all the rest variables are positively contributed. When last_credit_pull_d =109, it positively contributed the most.

  - FP – top 10 false positives, loan default = 0 and ordered by pred_1 score DECENDING (high scoring but actually didn't default)



Except self-reported annual income is 1e+05 and when the total amount committed by investors for that loan at that point in time is 15000, it negatively contributed to the False Positive prediction, all the rest variables are positively contributed. When last_credit_pull_d =109, it p positively contributed the most.

- FN - top 10 true negatives, loan default = 1 and ordered by pred_1 score ASCENDING (low scoring that did default)



When last_credit_pull_d =61, last_pymnt_amnt =1763.84 and installment =164.86, they are negatively contributed to the False Negative prediction.