IDS 572 - ASSIGNMENT 1

SPRING 2020 BATCH

SUBMITTED BY AND CONTRIBUTION:

ASHOK BHATRAJU – 670248723 – 33.33%
ARCHANA SINGH – 668528470 – 33.33%
NIKITA BAWANE 661069000 – 33.33%

1. **Describe the business model for Lending Club. Consider the stakeholders and their roles, and what advantages Lending Club offers. How does the platform make money?**

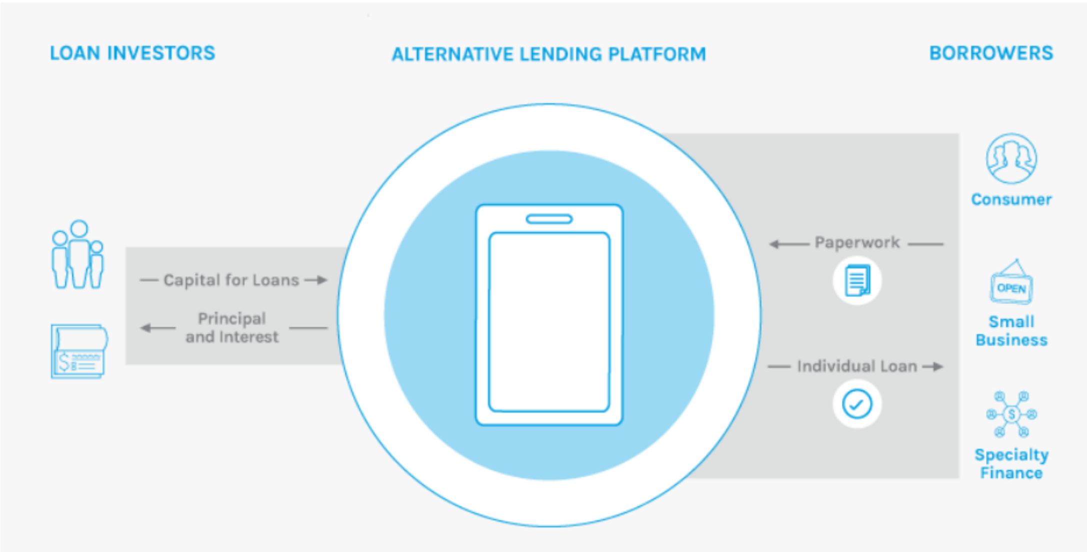
Lending club is a peer-to-peer lending platform. It enables borrowers to create unsecured personal loans ranging between $1000 to $40000. Each borrower is categorized into different grades(A-F) based on their credit score, credit history, desired loan amount and the borrower's debt-to-income ratio, Lending Club determines whether the borrower is credit worthy and assigns to its approved loans a credit grade that determines payable interest rate and fees.

Investors can search and browse the loan listings on Lending Club website and select loans that they want to invest based on the information supplied about the borrower, amount of loan, loan grade, and loan purpose. The loans can only be chosen at the interest rates assigned by Lending Club, but investors can decide how much to fund each borrower, with the minimum investment of $25 per note.

Investors make money from interest. Rates vary from 6.03% to 26.06%, depending on the credit grade assigned to the loan request. The grades assigned to these requests range alphabetically from A to G, with A being the highest-grade, lowest-interest loan. Each of these letter grades has five finer-grain sub-grades, numbered 1 to 5, with 1 being the highest sub-grade.

Stakeholders:

- Borrowers – Borrows money for various personal requirements.
- Lending club platform – rates the borrowers based on the available information. Shares this information with investors to help them with their investment.
- Investors – selects borrowers and invests based on their risk appetite.



Advantages Lending club offer:

- Complete transparency about the borrowers.
- Competitive interest rates for both borrowers and investors
- Short term loan duration
- Flexible repayment options
- Better interest rates compared to majority of the lending options

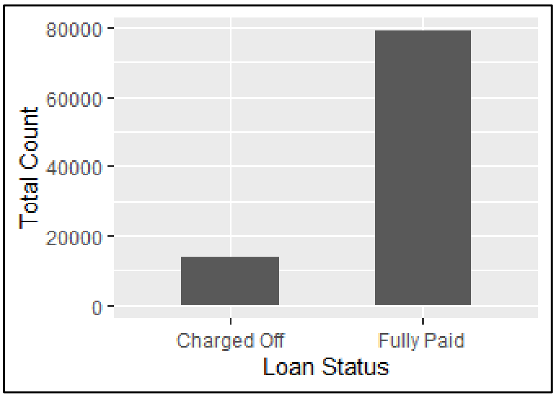
Lending club source of Money:

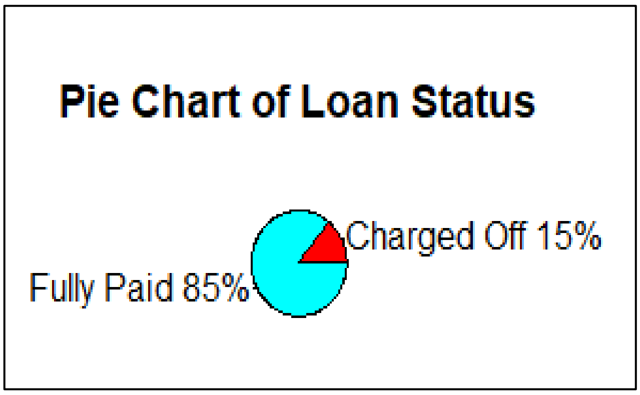
Lending Club makes money by charging borrowers an origination fee and investors a service fee. The size of the origination fee depends on the credit grade and ranges to be 1.1%-5.0% of the loan amount. The size of the service fee is 1% on all amounts the borrower pays.

Source: Lending Club Wikipedia

Data exploration (a)

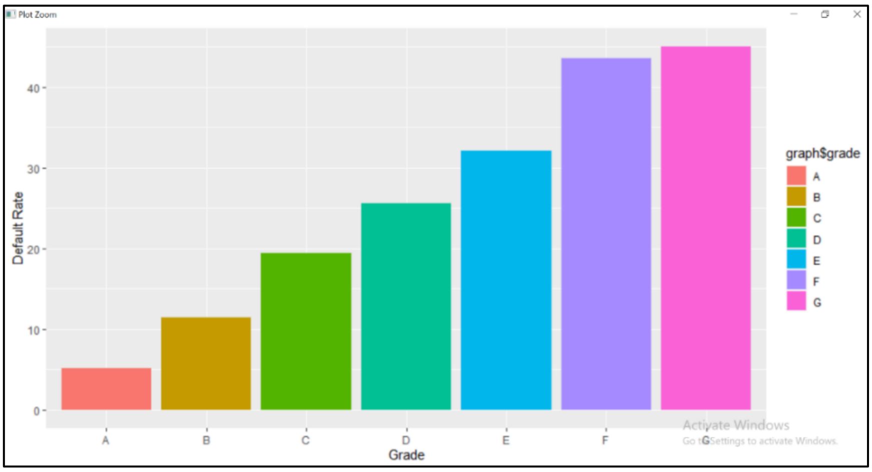
i) What is the proportion of defaults ('charged off' vs 'fully paid' loans) in the data? How does default rate vary with loan grade? Does it vary with sub-grade? And is this what you would expect, and why?





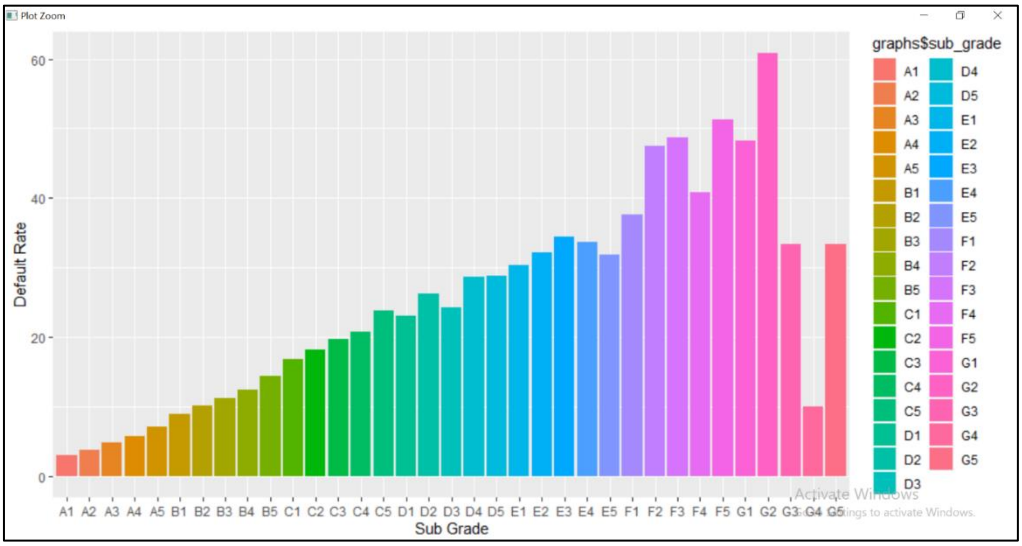
Interpretation: From plots above its clear that major part of borrowers paid off their loans. The dataset provided is an imbalanced dataset.

How does default rate vary with loan grade? Does it vary with sub-grade? And is this what you would expect, and why?



Interpretation:

Default rate increase from A to G. This is what we expected, as the interest rate and risk of investing increases from A to G.

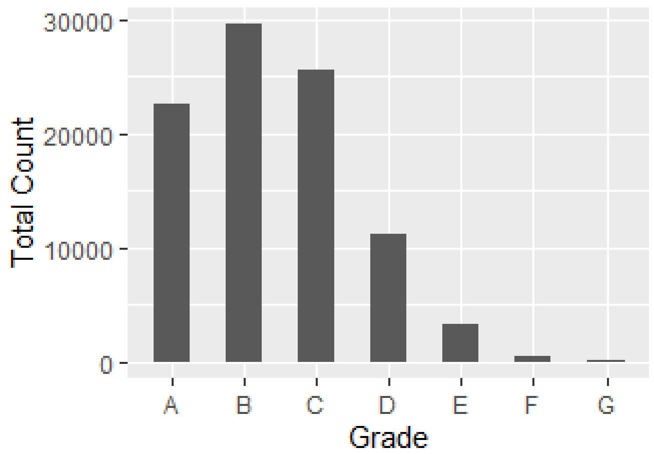


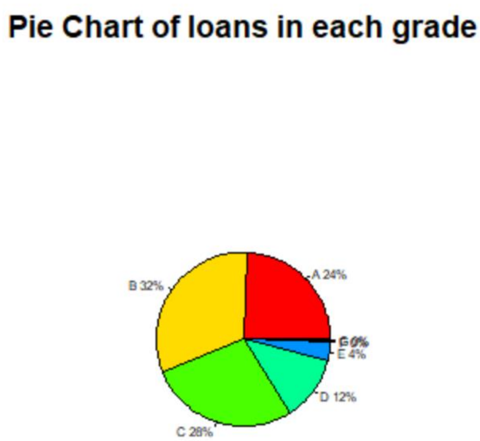
Interpretation:

As the earlier graph shown, default rate does increase from A to G. The pattern is also similar with the Sub grades. In sub grades of A, B and C, default rate increases from 1 to 5. In D the default rate of D3 is lower than D2, in E the default rate increases from E1 to E3 and decreases from there on till E5. In F, the default rate increases from F1 to F5 with F4 not following the pattern. In G the default rate is higher for G1 and G2 when compared to G3, G4 and G5.

Apart from G, default rate pattern is exactly what we expected it would be.

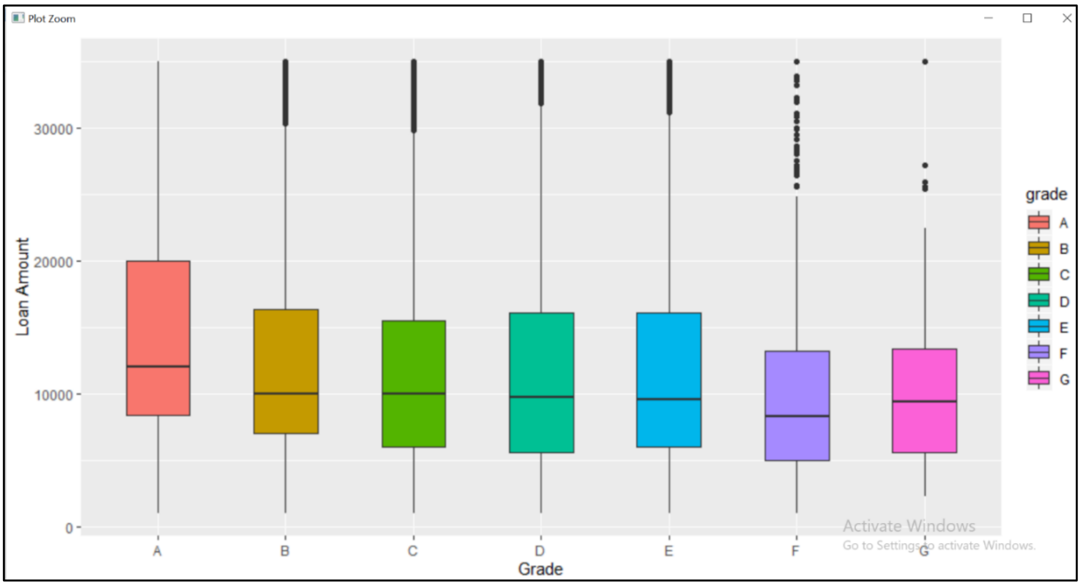
(ii) How many loans are there in each grade? And do loan amounts vary by grade? Does interest rate for loans vary with grade, subgrade? And is this what you expect, and why?





Interpretation:

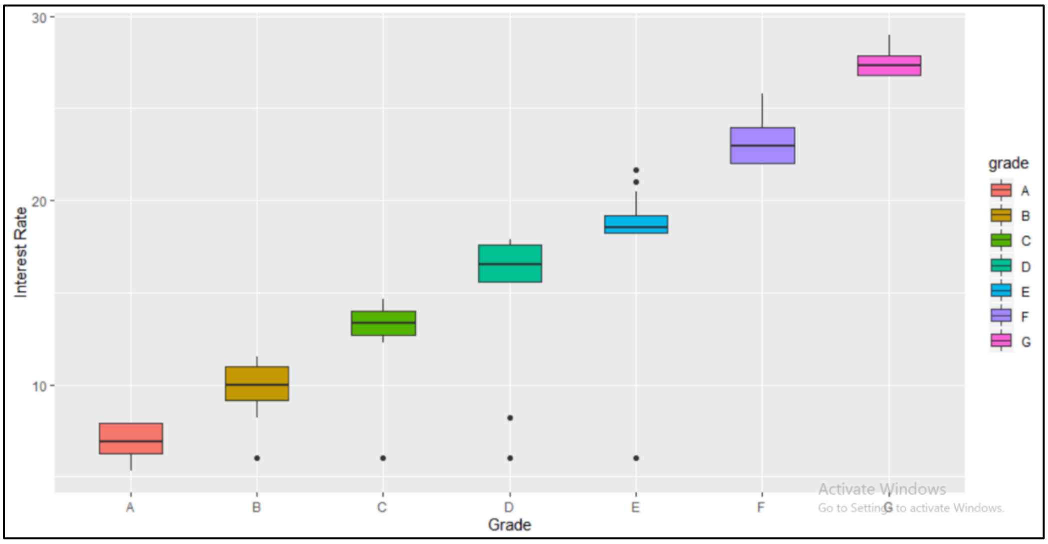
Count wise grade B has the highest number of loans in our given data and G has the least.



Interpretation:

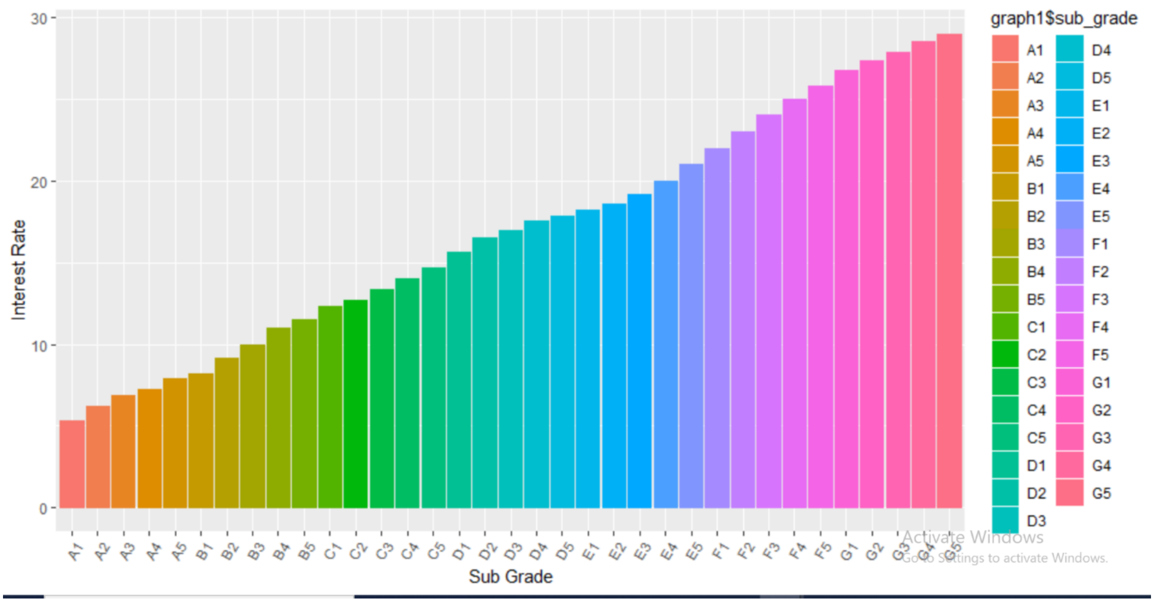
Loan amount does vary from grade to grade. It is highest for grade A and lowest for grade F. The range of loan amount is max – 12000 and min – 8000. So, the difference between grades is not that significant.

Interpretation:



Interest rate increases from A to G. This is what we expected, as the risk of investing increases from A to G the interest rate increases.

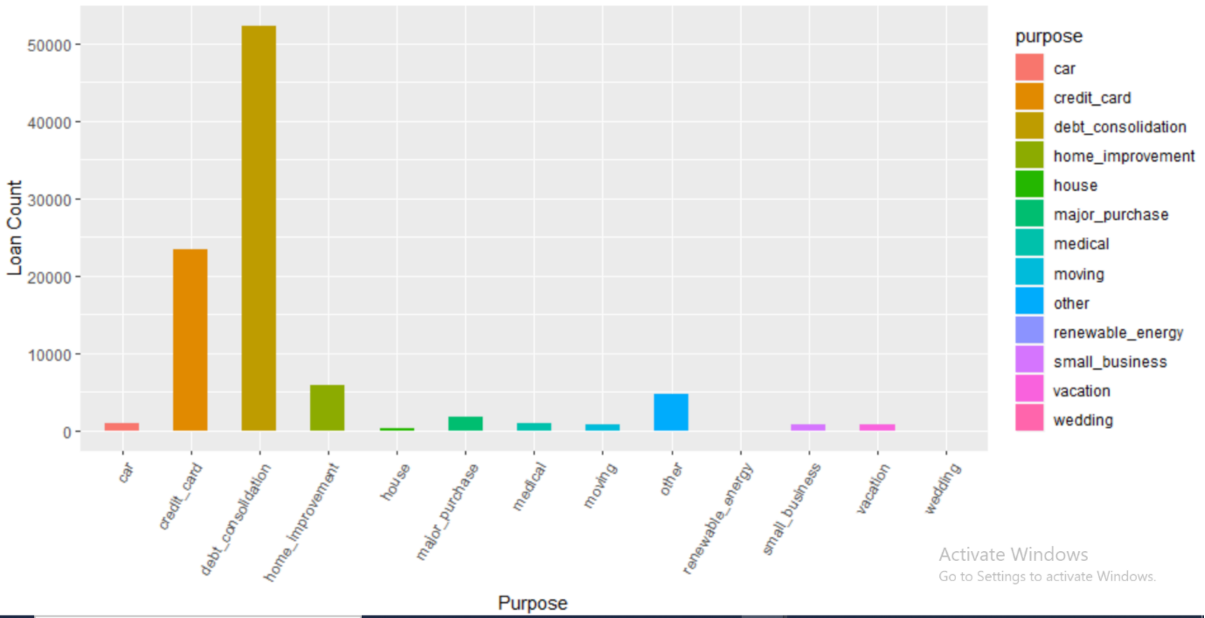
Risk – low credit scores, low annual incomes, other debt obligations and various other parameters that define the grade of the borrower



Interpretation:

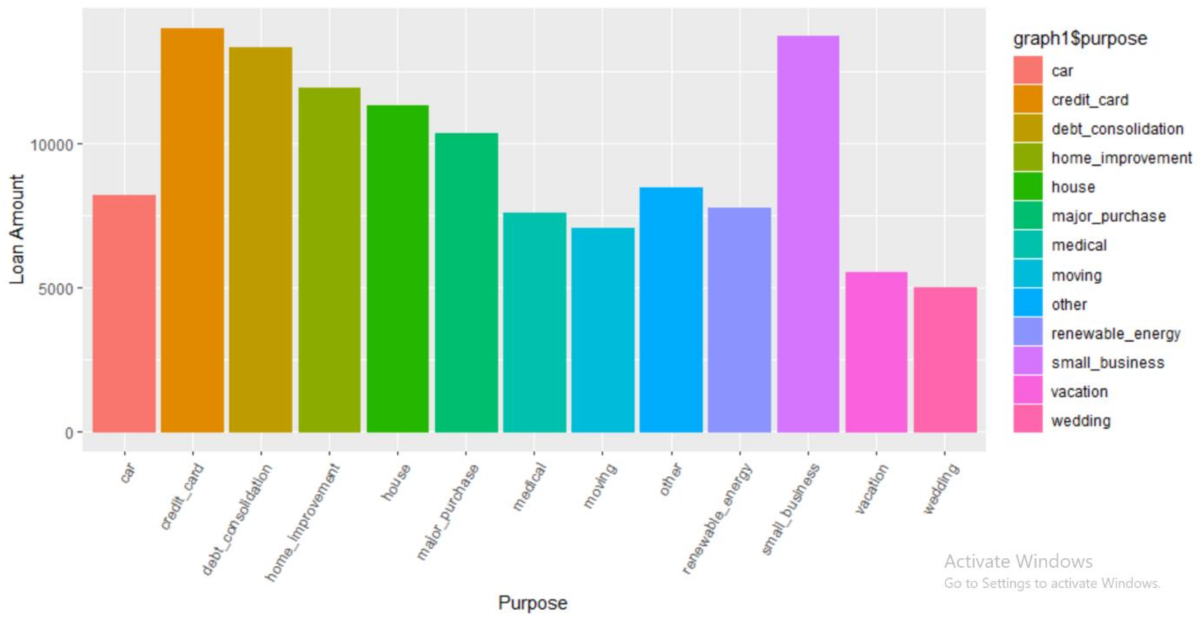
Interest rate increases from 1 to 5 in sub grades of all the grades. This is what we expected as the risk of investment increases from 1 to 5 in sub grades.

iii) What are people borrowing money for (purpose)? Examine how many loans, average amounts, etc. by purpose? And within grade? Do defaults vary by purpose?



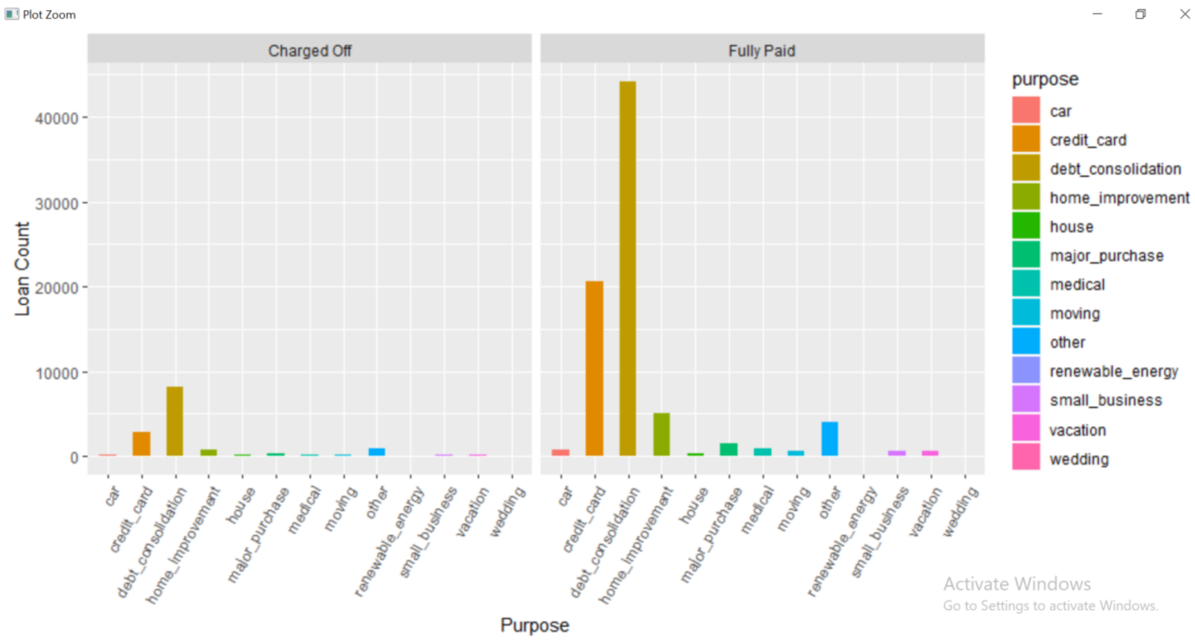
Interpretation:

As we can clearly see that the major reason people borrow loans is for **debt consolidation** and **credit card.**



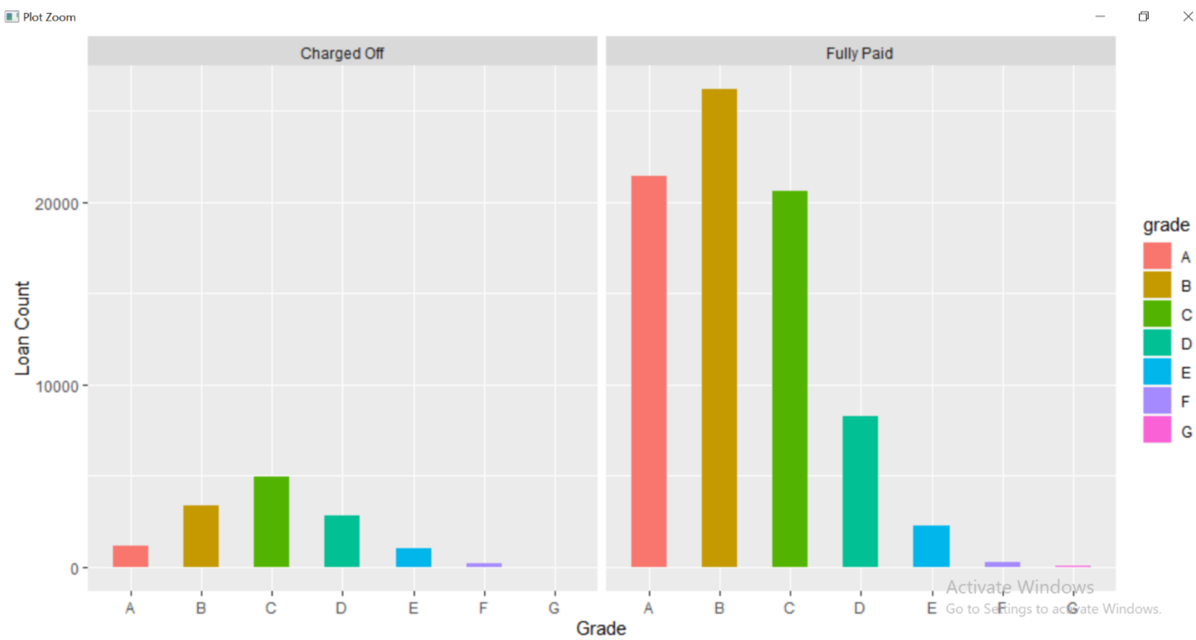
Interpretation:

Purpose wise Credit card, small business and debt consolidation are among the top 3 reasons that borrow highest loan amount. Wedding and vacation are the purposes with lowest loan amount. The max amount – 18000 and min amount – 5000.



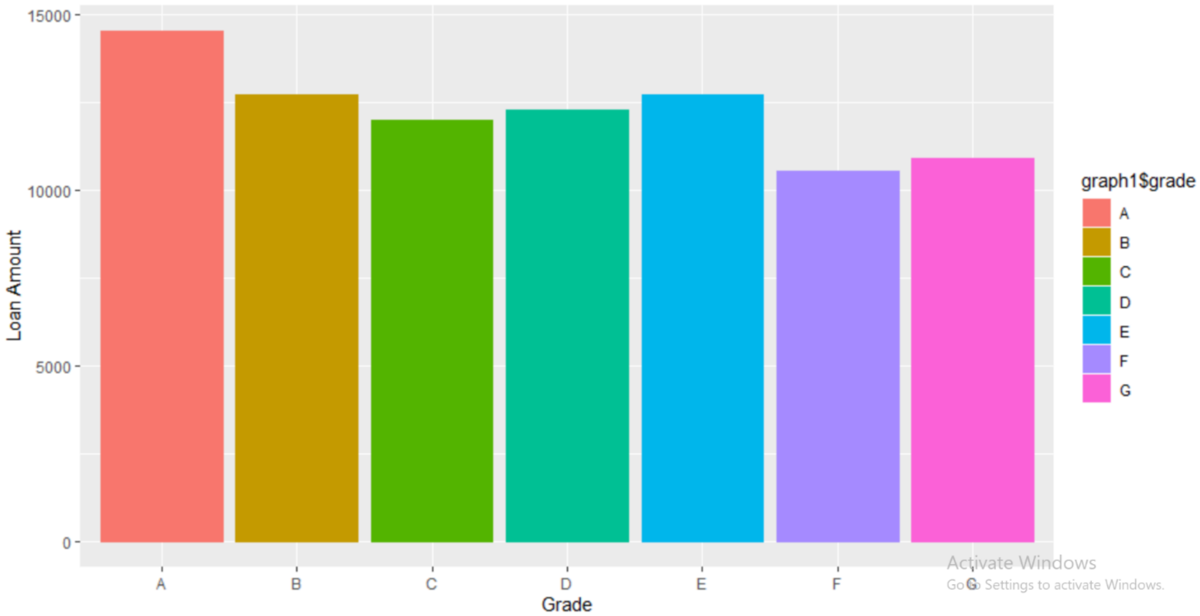
Interpretation:

Loan status wise charged off and fully paid is highest for debt consolidation and credit card. The reason can be, there are the reasons for which the number of loan borrowings are highest in number. Defaults doesn't vary by purpose, the proportion of loans borrowed for a particular purpose and defaults is similar.



Interpretation:

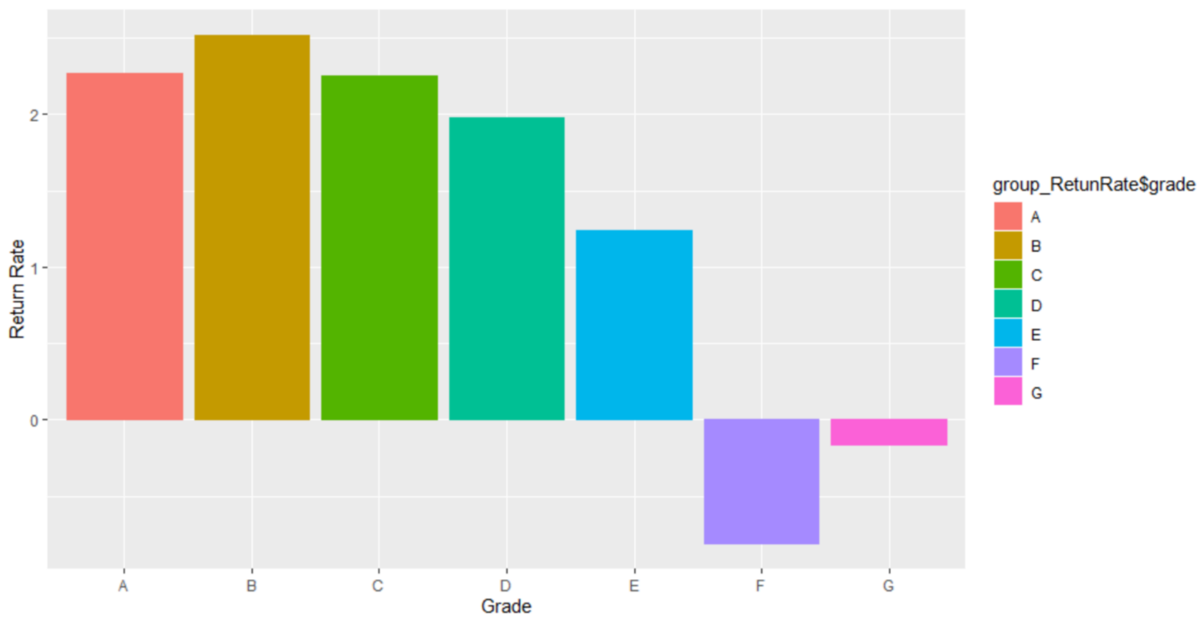
Grade wise the loan count is highest in grade C for Charged off and lowest for grade G. Grade wise the loan count is highest in grade B for fully paid and lowest for G. So, it does change by loan status.



Interpretation:

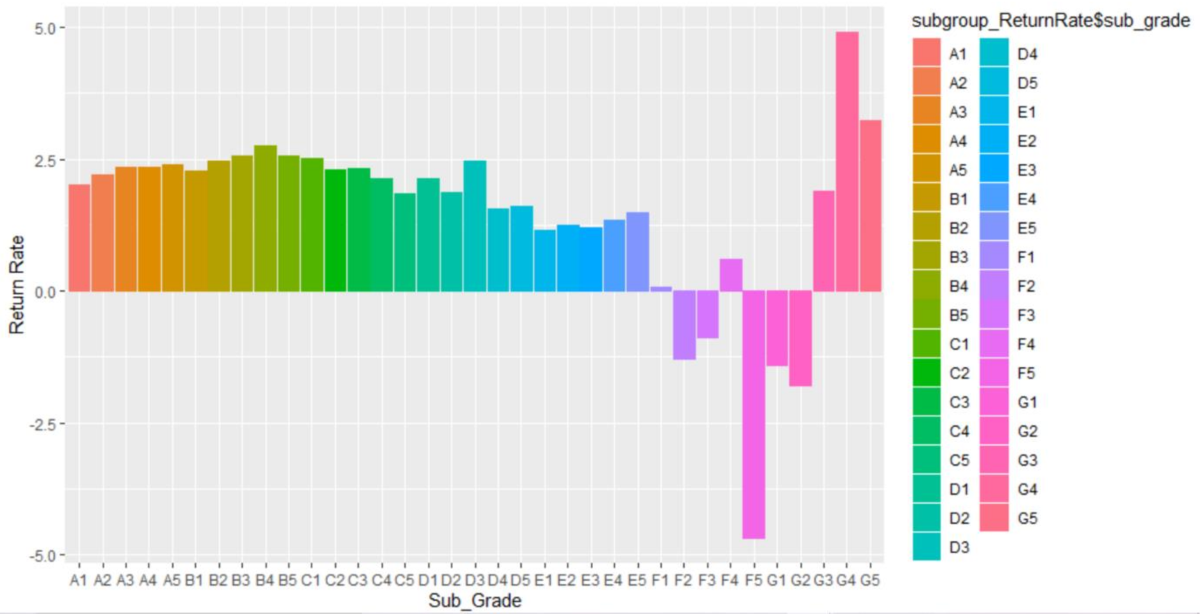
Loan amount does vary from grade to grade. It is highest for grade A and lowest for grade F. the maximum amount is around 14500 and lowest amount is around 11000.

(iv) Calculate the annual return. Show how you calculate the percentage annual return. Compare the average return values with the average interest rate on loans – do you notice any differences, and how do you explain this? How do returns vary by grade, and by sub-grade. 3 If you wanted to invest in loans based on this data exploration, which loans would you invest in?



Interpretation:

Annual Return rate is highest for grade B and lowest for grade F. average return rate is 2% and above for grades A, B, C and D. It is in negatives for F and G.

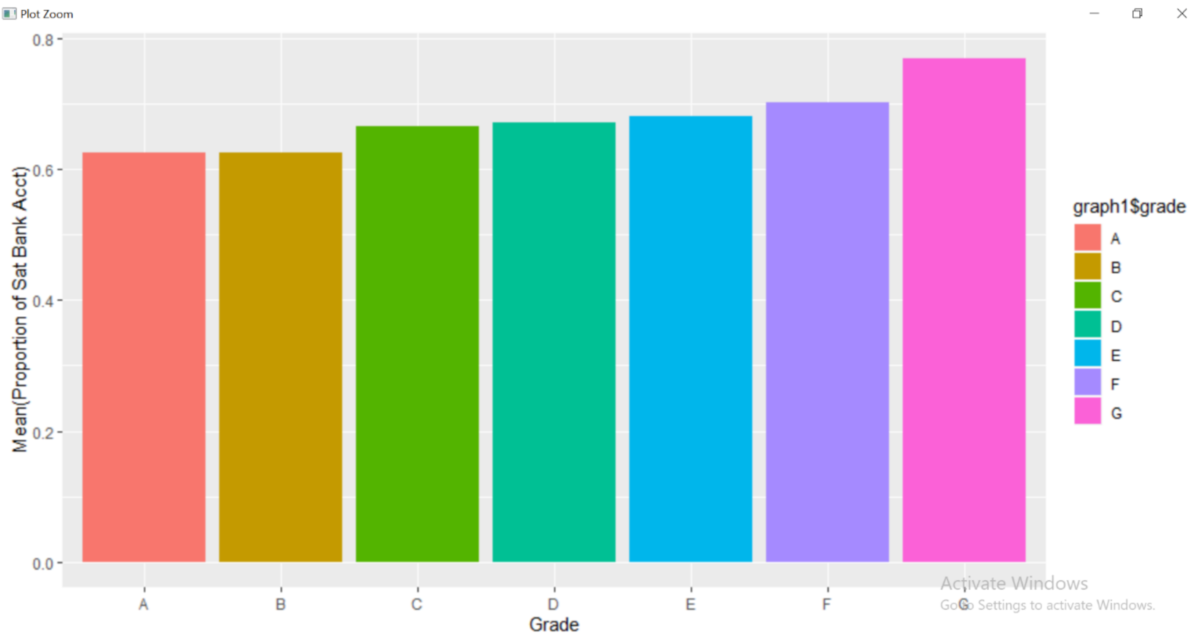


Interpretation:

Average return rate is highest for G4 and lowest for F5 in sub grades. It is in negatives for entire F sub grades and G1, G2.

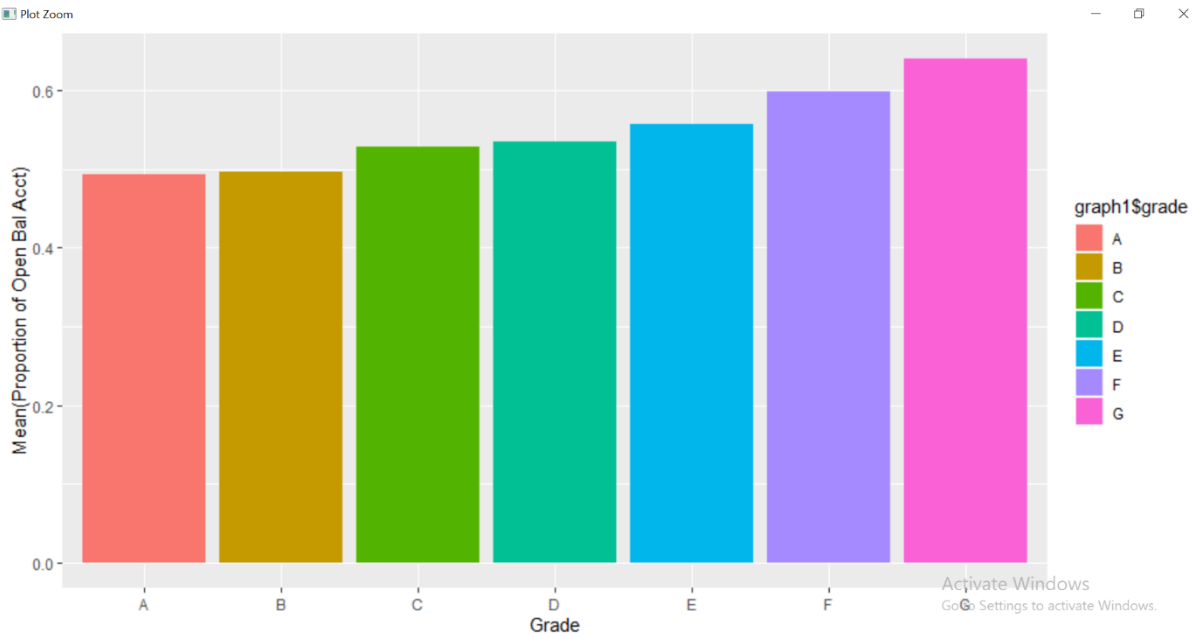
Based on the average return, I will invest in B (grade wise) and G4(sub-grade), they have the highest return rate of 2.5% and 4.9% respectively.

(v) Generate some new derived attributes which you think may be useful for predicting default. and explain what these are.



Interpretation:

Number of bankcard accounts and Number of satisfactory bankcard accounts are used to create an attribute proportion of satisfactory bankcard accounts. It is created by number of satisfactory bankcard accounts/number of bankcard accounts.



Interpretation:

The total number of credit lines currently in the borrower's credit file and number of open credit lines in the borrower's credit file are used to create proportion of open credit lines. It is created by number of open credit lines/number of credit lines.

3. **Consider the potential for data leakage. You do not want to include variables in your model which may not be available when applying the model; that is, some data may not be available for new loans before they are funded. Leakage may also arise from variables in the data which may have been updated during the loan period (ie., after the loan is funded). For example, it has been noted that the FICO scores on loan applicants are updated periodically, and the data can carry thus FICO scores from after the loan issue date. So, even though FICO score can be useful, the values in the data may not be usable. Identify and explain which variables you will exclude from the model.**

1. Remove all the attributes which have MORE than 60% NA values.
2. Find out important variables by running random forest model on the entire dataset (reimport data in a new table and remove all NA to run random forest, as random forest cannot be run on any NA data). After obtaining this list, we remov the variables apart from the important variable.

Variables Removed
funded_amnt
term
funded_amnt_inv
emp_title
emp_length
home_ownership
issue_d
pymnt_plan
title
zip_code
addr_state
earliest_cr_line
inq_last_6mths
out_prncp
out_prncp_inv
total_pymnt
total_pymnt_inv
total_rec_prncp

total_rec_int
total_rec_late_fee
recoveries
collection_recovery_fee
last_pymnt_d
last_pymnt_amnt
last_credit_pull_d
last_fico_range_high
last_fico_range_low
collections_12_mths_ex_med
policy_code
application_type
bc_util
hardship_flag

Some of the variables retained which were not in the variable importance list were revol_util, initial_list_status, mo_sin_old_rev_tl_op, mo_sin_rcnt_tl, mths_since_recent_bc, mths_since_recent_inq, pub_rec_bankruptcies, tax_liens.

4. Develop decision tree models to predict default.

(a) Split the data into training and validation sets. What proportions do you consider, why?

The data is split into training and test set in the ratio of 70:30 (70-trainset and 30-testset) and 60:40. More training data is a good thing because it makes the classification model better. We have ensured that our test set meets the following criteria-

1. Is large enough to yield statistically meaningful results.
2. Is representative of the data set as a whole. In other words, don't pick a test set with different characteristics than the training set.

(b) Train decision tree models (use both rpart, c50) [If something looks too good, it may be due to leakage – make sure you address this] What parameters do you experiment with, and what performance do you obtain (on training and validation sets)? Clearly tabulate your results and briefly describe your findings. How do you evaluate performance – which measure do you consider, and why?

Decision Tree

Type	Split of Training and Test Data	CP Value	Minimum Split	Threshold	Accuracy		Sensitivity		AUC	Comments
					Training	Test	Training	Test		
Information	60/40	0.00029491	40	0.5	0.87	0.87	0.19	0.17	0.7066	
		0.00029491	40	0.3	0.877	0.87	0.20	0.17		
		0.00029491	40	0.2	0.7765	0.7738	0.477	0.45634		
		0.00029491	40	0.3	0.75	0.6064	0.75	0.69	0.686312	oversampling
		0.0002949	40	0.5	0.7541	0.7026	0.75	0.5787		oversampling. BEST TREE FOR STANDARD THRESHOLD AND OVERSAMPLING
		0.0002949	40	0.4	0.75	0.6931	0.75	0.59		oversampling
		0.0005	40	0.5	0.8755	0.8764	0.1596	0.15488	0.5774	
		0.0005	40	0.3	0.8755	0.876	0.1596	0.154		
		0.0005	40	0.4	0.8755	0.876	0.1596	0.154		
		0.0005	40	0.4	0.8755	0.61	0.1596	0.71	0.7205	oversampling
		0.0005	30	0.4	0.68	0.6145	0.68	0.71	0.72	oversampling. BEST TREE FOR THRESHOLD = 0.4 AND OVERSAMPLING
		0.0005	30	0.4	0.875	0.874	0.159	0.154	0.64	
		0.0001	30	0.5	0.89	0.85	0.40	0.24	0.701	BEST TREE FOR STANDARD THRESHOLD AND WITHOUT OVERSAMPLING
		0.0001	30	0.2	0.89	0.8179	0.40	0.339	0.64	BEST TREE FOR THRESHOLD = 0.2 AND WITHOUT OVERSAMPLING
		0.0001	30	0.3	0.8945	0.84	0.4	0.26		
		0.00043(for pruning)	30	0.3	0.875	0.8764	0.159	0.1548	0.577	pruning
Information	70/30	0.0005	30	0.3	0.6853	0.6072	0.5881	0.7322	0.7283	oversampling
		0.0005	30	0.5	0.7	0.683	0.7	0.641		oversampling
		0.0005	30	0.5	0.875	0.877	0.157	0.1582	0.579	
		0.0005	30	0.3	0.8751	0.8776	0.157	0.158		

The best decision tree from rpart is for standard threshold – 0.5 and without oversampling is for CP = 0.0001 (highlighted in pink). The positive class for decision tree is ‘Charged Off’.

C5.0 - The positive class for decision tree is ‘Charged Off’. The train dataset for the best model for C5.0(highlighted in yellow) has accuracy of 72.32, sensitivity is 61.4.

Sl.No.	Split	Threshold	Trials	CF	Accuracy for TestSet	Sensitivity for TestSet	AUC
1	70/30	0.2	30	0.4	69.9	60.02	72.65
2	70/30	0.2	30	0.5	69.35	60.22	60.22
3	70/30	0.3	20	0.25	71.54	60.27	74.34
4	70/30	0.3	50	0.3	72.09	61.11	72.94
5	70/30	0.2	40	0.4	69.03	62.35	73.18
6	70/30	0.2	40	0.5	68.54	62.92	62.92
7	70/30	0.2	50	0.4	68.17	64.01	73.35
8	70/30	0.2	50	0.5	67.41	64.13	64.13
9	70/30	0.2	10	0.25	64.36	69.31	73.42
10	70/30	0.2	10	0.3	64.36	69.31	73.42

The following parameters have been changed and experimented with-

Rpart Decision Tree

1. Training and Test Dataset Split (60/40 and 70/30)
2. Different CP and nsplit values combination
3. Different Thresholds (0.2,0.3, 0.4, 0.5).
4. Oversampling – Since the data is imbalanced (Charged Off records have less records compared to Fully Paid), the sensitivity to predict “Charged Off” was below 0.5. To curb the affect of imbalanced dataset, we oversampled the trainset.

C5.0

1. Trial – 10, 20, 30, 40 and 50.
2. CF – 0.25, 0.3, 0.4, 0.5

3. Training and Test Set Split 70/30.
4. Threshold – 0.2, 0.3

The following parameters will help us evaluate as to which is the best prediction model:

1. Accuracy
2. Sensitivity
3. ROC (AUC) - It tells how much model is capable of distinguishing between classes. Higher the AUC, better the model is at predicting 0s as 0s and 1s as 1s.

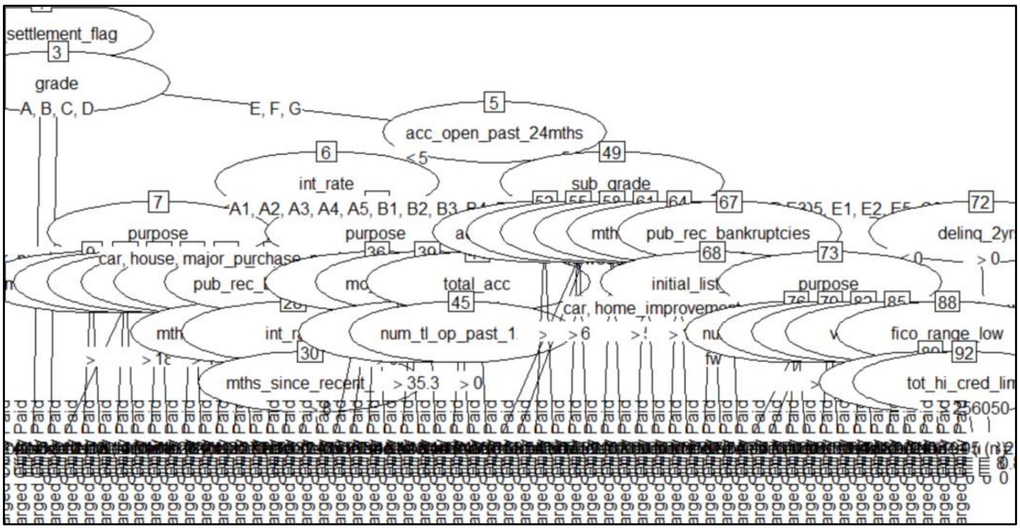
The profit on good customer loan is not equal to the loss on one bad customer loan. The loss on one bad loan might eat up the profit on 100 good customers. In this case one bad customer is not equal to one good customer. If p is probability of default then we would like to set our threshold in such a way that we don't miss any of the bad customers.
We set the threshold in such a way that Sensitivity is high
We can compromise on specificity here. If we wrongly reject a good customer, our loss is very less compared to giving a loan to a bad customer.

We don't really worry about the good customers here; they are not harmful hence we can have less Specificity.

(c) Identify the best tree model. Why do you consider it best? Describe this model – in terms of complexity (size). Examine variable importance. Briefly describe how variable importance is obtained in your best model.

After comparing the models from rpart and C5.0, the best tree model selected is from C5.0.

The C5.0 tree model overall gives a better accuracy (72.09%), sensitivity (61.11%) and AUC value (74.94%). The model with threshold 0.3 is chosen (Sl.No.4 from the table above) even though 0.2 threshold has better values. Hence, we are selecting the model C5.0 model with Threshold = 0.3, Trials – 50, CF – 0.3 and Training/Test Data split – 70/30.



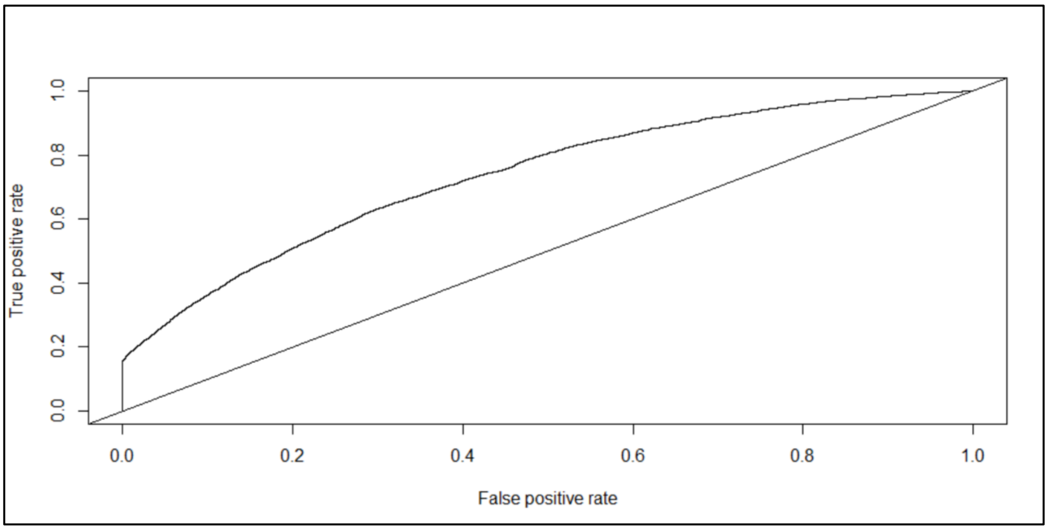
<u>Variable Importance</u>

```
> C5imp(ctree)
                          Overall
tot_hi_cred_lim            100.00
debt_settlement_flag       100.00
mo_sin_old_il_acct          99.46
num_bc_tl                   99.23
int_rate                    99.22
num_rev_accts               99.13
num_op_rev_tl               99.08
mths_since_recent_bc        98.82
delinq_2yrs                 98.77
mo_sin_old_rev_tl_op        98.53
num_tl_op_past_12m          98.51
acc_open_past_24mths        98.47
total_bc_limit              98.47
```

num_actv_rev_tl	98.33
pct_tl_nvr_dlq	98.33
avg_cur_bal	98.20
chargeoff_within_12_mths	98.13
tot_cur_bal	98.12
grade	98.11
installment	98.07
num_il_tl	98.05
open_acc	98.02
purpose	98.00
mo_sin_rcnt_tl	98.00
sub_grade	97.99
bc_open_to_buy	97.96
num_actv_bc_tl	97.95
num_bc_sats	97.93
total_acc	97.80
pub_rec	97.76
fico_range_low	97.76
total_rev_hi_lim	97.71
num_accts_ever_120_pd	97.68
dti	97.64
num_tl_30dpd	97.63
total_il_high_credit_limit	97.59
mths_since_recent_inq	97.57
total_bal_ex_mort	97.46
revol_bal	97.46
pub_rec_bankruptcies	97.45
loan_amnt	97.44
tot_coll_amt	96.98
num_tl_90g_dpd_24m	96.95
num_tl_120dpd_2m	96.80
annual_inc	96.77
mths_since_last_delinq	96.18
mort_acc	95.18
revol_util	95.02
num_rev_tl_bal_gt_0	95.00
mo_sin_rcnt_rev_tl_op	94.91
tax_liens	93.55
delinq_amnt	92.23
verification_status	90.80
acc_now_delinq	90.01
num_sats	89.64
percent_bc_gt_75	82.77
initial_list_status	51.16
fico_range_high	0.00

ROC Curve-

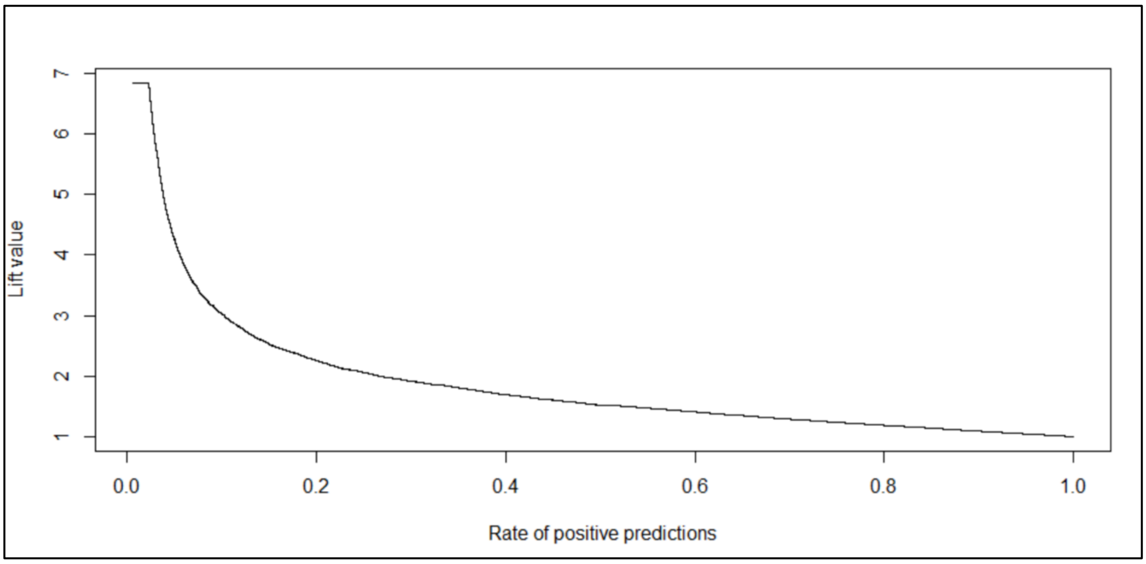
The AUC (Area under the Curve) is 74.94%. A model is considered good if the AUC is between 0.6 and 1.



Lift Graph –

Lift is a measure of the effectiveness of a predictive model calculated as the ratio between the results obtained with and without the predictive model. The greater the area between the lift curve and the baseline, the better the model.

The lift curve output for best model of (C5.0) shows that the model is good.



5. **Develop a random forest model. What parameters do you experiment with, and does this affect performance? Describe the best model in terms of number of trees, performance, variable importance. Compare the random forest and best decision tree model from Q 4 above. Do you find the importance of variables to be different? Which model would you prefer, and why. For evaluation of models, you should include confusion matrix related measures, as well as ROC analyses and lifts. Explain which performance measures you consider, and why.**

Split of Training and Test Data	Ntree	Threshold	Accuracy	Sensitivity	Specificity	Pos Pred Value	Neg Pred Value	AUC
70/30	10	0.7	87.49	14.1	99.97	99.13	87.24	**67.9**
70/30	10	0.5	86.44	19.82	97.77	60.27	87.75	AUC remains the same as it is built on model
70/30	10	0.3	83.05	29.03	92.24	38.9	88.42	
70/30	10	0.2	76.17	43.14	81.78	28.72	89.42	
70/30	**10**	**0.1**	**62.28**	**62.97**	62.16	22.06	90.79	
70/30	20	0.7	87.7	15.44	99.99	99.84	87.42	**70.23**
70/30	20	0.5	87.39	16.98	99.36	82.05	87.55	AUC remains the same as it is built on model
70/30	20	0.3	83.75	29.5	92.98	41.68	88.57	
70/30	20	0.2	74.8	47.1	79.51	28.12	89.83	
70/30	**20**	**0.1**	**56.53**	**72.97**	53.74	21.16	92.12	
70/30	50	0.7	87.76	15.79	100	100	87.46	**72.35**
70/30	50	0.5	87.72	16.16	99.89	96.31	87.51	AUC remains the same as it is built on model
70/30	50	0.3	84.75	27.77	94.44	45.94	88.48	
70/30	**50**	**0.2**	**74.73**	**52.25**	78.55	29.3	90.62	
70/30	**50**	**0.1**	**52.17**	**80.5**	47.35	20.64	93.45	
70/30	100	0.7	87.76	15.82	100	100	87.47	**73.29**
70/30	100	0.5	87.74	15.99	99.94	98.02	87.48	AUC remains the same as it is built on model
70/30	100	0.3	85.03	27.47	94.81	47.41	88.48	
70/30	**100**	**0.2**	**74.33**	**53.74**	77.82	29.2	90.82	
70/30	**100**	**0.1**	**50.3**	**83.84**	44.6	20.47	94.19	
70/30	200	0.7	87.76	15.82	100	100	87.47	**73.53**
70/30	200	0.5	87.76	15.94	99.97	99.23	87.48	AUC remains the same as it is built on model
70/30	200	0.3	85.45	27.22	95.35	49.9	88.5	
70/30	**200**	**0.2**	**74.06**	**54.2**	77.43	29.01	90.85	
70/30	200	0.1	49.03	85.25	42.87	20.25	94.47	
70/30	500	0.7	87.76	15.82	100	100	87.47	**73.89**
70/30	500	0.5	87.76	15.91	99.98	99.38	87.48	AUC remains the same as it is built on model
70/30	500	0.3	85.62	26.65	95.65	51.06	88.46	
70/30	500	0.2	73.94	55.61	77.05	29.19	91.07	
70/30	500	0.1	48.47	86.01	42.08	20.17	94.65	

The random forest tree is the model with ntree = 500, threshold = 0.2 because it has the highest sensitivity (55.61), highest AUC(73.89) and accuracy (73.94). The positive class for random forest tree is 'Charged Off'.

The parameters that we have experimented with to determine the best random forest model are-

1. Ntree – 10,20,50,100,200
2. Threshold values – 0.5, 0.3, 0.2, 0.1, 0.7
3. Training and Test Dataset Split – 70/30

```
> confusionMatrix(predRF, testset$loan_status)
Confusion Matrix and Statistics

Reference
Prediction    Charged Off Fully Paid
Charged Off        2246       5477
Fully Paid         1794      18270

Accuracy : 0.7383
95% CI : (0.7331, 0.7435)
No Information Rate : 0.8546
P-Value [Acc > NIR] : 1

Kappa : 0.236

Mcnemar's Test P-Value : <2e-16

Sensitivity : 0.55594
Specificity : 0.76936
Pos Pred Value : 0.29082
Neg Pred Value : 0.91059
Prevalence : 0.14539
Detection Rate : 0.08083
Detection Prevalence : 0.27794
Balanced Accuracy : 0.66265

'Positive' Class : Charged Off
```

Variable importance –

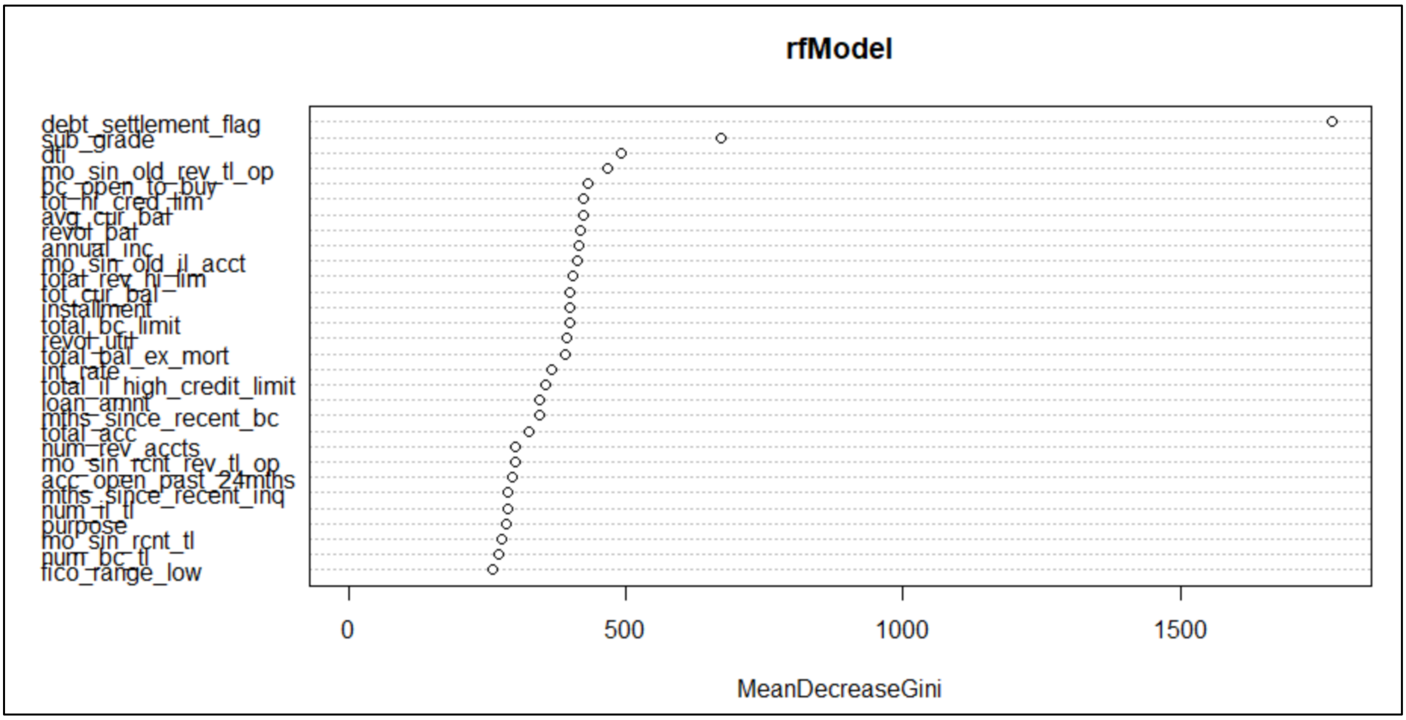
```
> (VI_F=importance(rfModel)) -
```

This importance is a measure of by how much removing a variable decreases accuracy, and vice versa — by how muc h including a variable increases accuracy.
When a tree is built, the decision about which variable to split at each node uses a calculation of the Gini impurity. For each variable, the sum of the Gini decrease across every tree of the forest is accumulated every time that variabl e is chosen to split a node. The sum is divided by the number of trees in the forest to give an average.
Neither measure is perfect but viewing both together allows a comparison of the importance ranking of all variables across both measures.

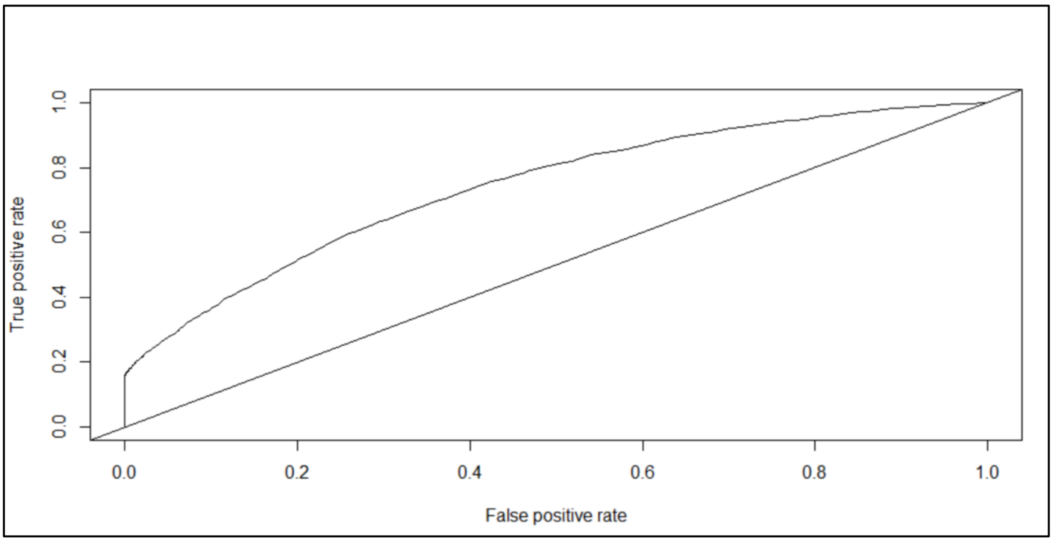
```
                          Charged Off   Fully Paid
loan_amnt                 -33.0497943   40.1871365
int_rate                   22.9191777   10.1583573
installment               -32.3761336   49.2929619
grade                       8.8107726   13.9126911
sub_grade                  47.3245621   -2.5846691
annual_inc                -23.8401918   46.1445355
verification_status         1.0190712    7.7834432
purpose                    -2.6365741   17.6960268
dti                        -2.8274406   44.5640320
delinq_2yrs                -7.1938173   12.1016732
fico_range_low            -11.0596785   36.2035481
fico_range_high           -11.2835799   35.0822381
mths_since_last_delinq    -11.4613574   26.4709760
open_acc                  -29.8962465   45.9333098
pub_rec                    -2.9966465   16.8788591
revol_bal                 -42.2322093   52.3192984
revol_util                -17.9578138   40.4802270
total_acc                 -30.4408970   52.5362214
initial_list_status         0.7974693    4.5674767
acc_now_delinq             -1.1144040   -2.1689878
tot_coll_amt               -2.2443744   14.8778834
tot_cur_bal               -37.4572482   42.0345184
```

```
total_rev_hi_lim           -38.2720546  42.0854386
acc_open_past_24mths        -7.3290411  41.3489547
avg_cur_bal                -36.8341261  47.8437924
bc_open_to_buy             -30.0224971  43.1019304
chargeoff_within_12_mths     1.0913013   6.3988498
delinq_amnt                  2.5319729   0.3262353
mo_sin_old_il_acct         -12.8727712  31.9659110
mo_sin_old_rev_tl_op        -9.1599617  42.8012948
mo_sin_rcnt_rev_tl_op      -22.0625840  39.7291601
mo_sin_rcnt_tl             -15.0495491  38.0221202
mort_acc                   -12.7927550  21.2932539
mths_since_recent_bc        -8.6634182  30.9208527
mths_since_recent_inq       -4.5680520   9.2921244
num_accts_ever_120_pd       -8.1489496  17.7767722
num_actv_bc_tl             -22.5268923  36.3184697
num_actv_rev_tl            -23.3988275  30.6886459
num_bc_sats                -27.5433409  46.3006760
num_bc_tl                  -30.5305136  52.4960649
num_il_tl                  -21.6946204  44.1001101
num_op_rev_tl              -36.7260782  49.0587563
num_rev_accts              -36.2477701  56.7287711
num_rev_tl_bal_gt_0        -24.5891474  29.8554587
num_sats                   -30.2359775  41.9475085
num_tl_120dpd_2m            -5.6842467  15.2084975
num_tl_30dpd                -0.1400103  -0.8376607
num_tl_90g_dpd_24m          -3.7271113  10.6844949
num_tl_op_past_12m         -17.0566479  30.8526387
pct_tl_nvr_dlq             -16.0360933  33.4733415
percent_bc_gt_75           -19.9611533  37.6577209
pub_rec_bankruptcies        -3.6107821  15.5223088
tax_liens                   -0.6889328   5.2969318
tot_hi_cred_lim            -36.4995980  42.0352195
total_bal_ex_mort          -45.8796508  50.8995493
total_bc_limit             -34.6937152  39.5363945
total_il_high_credit_limit -35.2849453  52.5822766
debt_settlement_flag       183.4223159 187.2718681
                           MeanDecreaseAccuracy MeanDecreaseGini
loan_amnt                             37.411662       342.405452
int_rate                              15.674985       364.906689
installment                           47.347336       396.924890
grade                                 19.784179       235.296347
sub_grade                              2.716743       670.184526
annual_inc                            44.123329       413.559476
verification_status                    7.409066       100.409040
purpose                               15.794470       281.794869
dti                                   43.904319       491.434812
delinq_2yrs                           10.900905        79.795596
fico_range_low                        33.383168       258.874530
fico_range_high                       32.739942       258.129527
mths_since_last_delinq                24.151794       237.118916
open_acc                              43.352165       234.622057
pub_rec                               15.898793        76.520176
revol_bal                             48.785584       416.684674
revol_util                            36.370859       393.497108
total_acc                             49.760949       325.245704
initial_list_status                    4.602468        45.683071
acc_now_delinq                        -2.429433         5.285367
tot_coll_amt                          13.090166       145.938234
tot_cur_bal                           40.931242       398.101622
total_rev_hi_lim                      40.052956       403.975660
acc_open_past_24mths                  42.202066       293.523488
avg_cur_bal                           46.641774       421.169706
bc_open_to_buy                        42.367069       431.173075
chargeoff_within_12_mths               6.573562        16.954861
delinq_amnt                            1.122688         7.264271
mo_sin_old_il_acct                    27.074408       411.240649
mo_sin_old_rev_tl_op                  41.106607       465.299748
mo_sin_rcnt_rev_tl_op                 36.236549       298.077487
mo_sin_rcnt_tl                        34.774313       274.431206
mort_acc                              20.614410       146.877839
mths_since_recent_bc                  28.184446       342.356987
mths_since_recent_inq                  6.893157       286.480587
num_accts_ever_120_pd                 15.150654       104.252215
num_actv_bc_tl                        34.419481       193.565299
num_actv_rev_tl                       30.190868       211.198792
num_bc_sats                           43.840765       211.370933
```

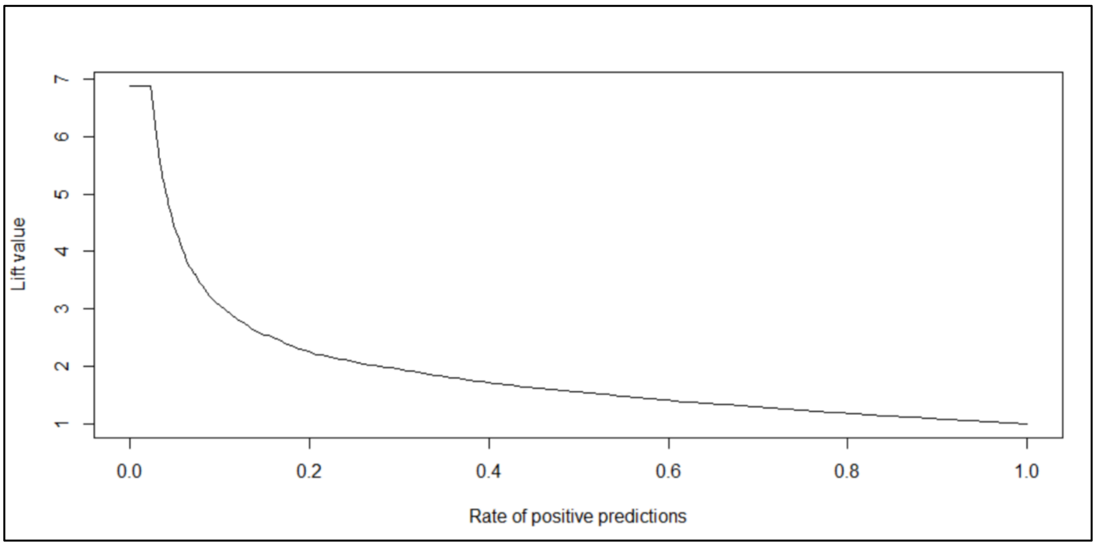
```
num_bc_tl                          47.905778      269.624185
num_il_tl                          40.124128      285.361132
num_op_rev_tl                      46.494382      229.044066
num_rev_accts                      52.288201      298.114130
num_rev_tl_bal_gt_0                28.765045      206.235422
num_sats                           39.105202      233.978331
num_tl_120dpd_2m                   14.862774       26.874524
num_tl_30dpd                       -0.866663        2.359445
num_tl_90g_dpd_24m                  9.685656       43.065667
num_tl_op_past_12m                 30.153534      202.012572
pct_tl_nvr_dlq                     30.306519      243.965495
percent_bc_gt_75                   34.450898      212.643993
pub_rec_bankruptcies               14.438056       55.454746
tax_liens                           4.905738       38.609782
tot_hi_cred_lim                    40.985189      422.993980
total_bal_ex_mort                  48.252076      389.539789
total_bc_limit                     37.836267      396.631144
total_il_high_credit_limit         50.501267      354.526527
debt_settlement_flag              189.772838     1775.047110

> varImpPlot(rfModel,type=2)
```



ROC Curve

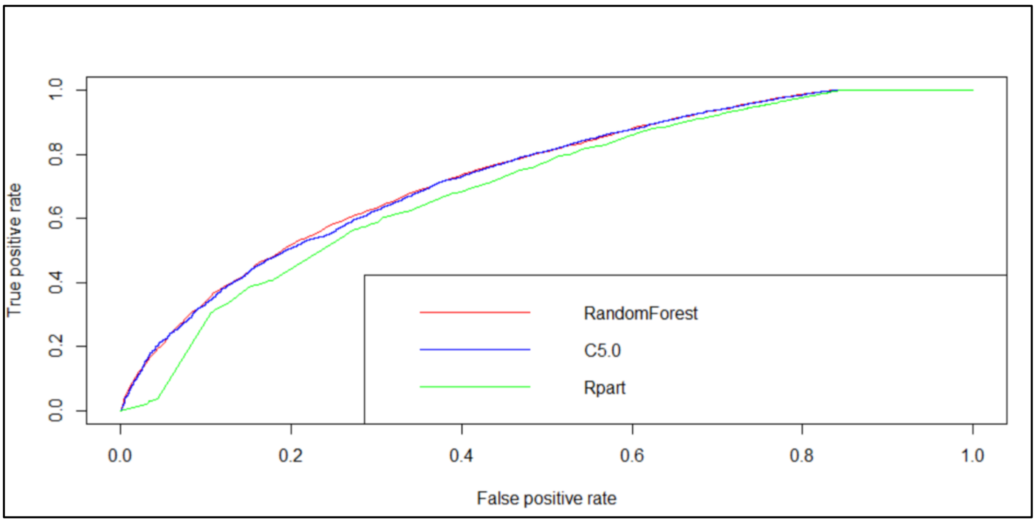


Lyft Curve



6 (a) Compare the performance of your models from Qs 4 and 5 above based on this. Note that the confusion matrix depends on the classification threshold/cutoff you use. Evaluate different thresholds and analyze performance. Which model do you think will be best, and why.

Results for different thresholds for rpart, C5.0 and random forest has been captured in Q4 and Q5. Please refer to the table attached for these questions. The best models from Q4(rpart and C5.0) and Q5(random forest) have been plotted on a consolidated ROC curve below. Based on the AUC plot, AUC Random Forest model (73.49) seems to be (slightly) better than the C5.0 mode (72.94).



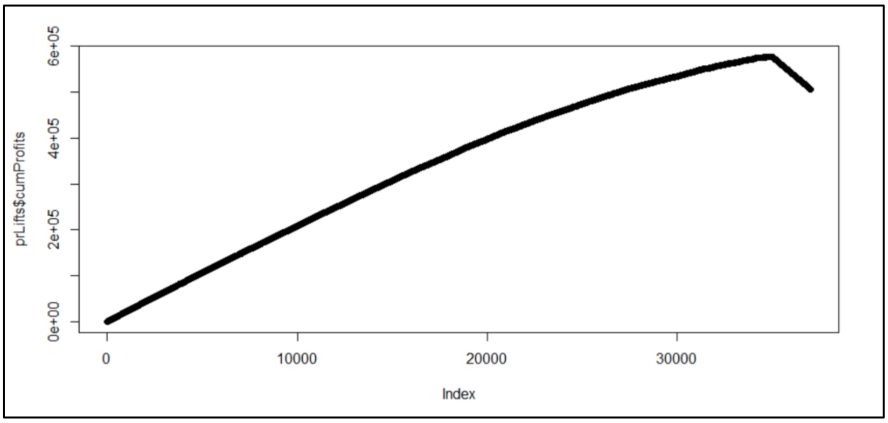
(b) Another approach is to directly consider how the model will be used – you can order the loans in descending order of prob(fully-paid). Then, you can consider starting with the loans which are most likely to be fully-paid and go down this list till the point where overall profits begin to decline (as discussed in class). Conduct an analyses to determine what threshold/cutoff value of prob(fully-paid) you will use and what is the total profit from different models. Also compare the total profits from using a model to that from investing in the safe CDs. Explain your analyses and calculations. Which model do you find to be best and why. And how does this compare with what you found to be best in part (a). some loans ae paid before their maturity.

The calculated values for COSTVAL and PROFITVAL is obtained from return rate. Actual return rate is chosen over interest rate because some loans are paid before their maturity and interest rate does not handle this scenario. The average return rate obtained from the return rates calculated with actual term is-

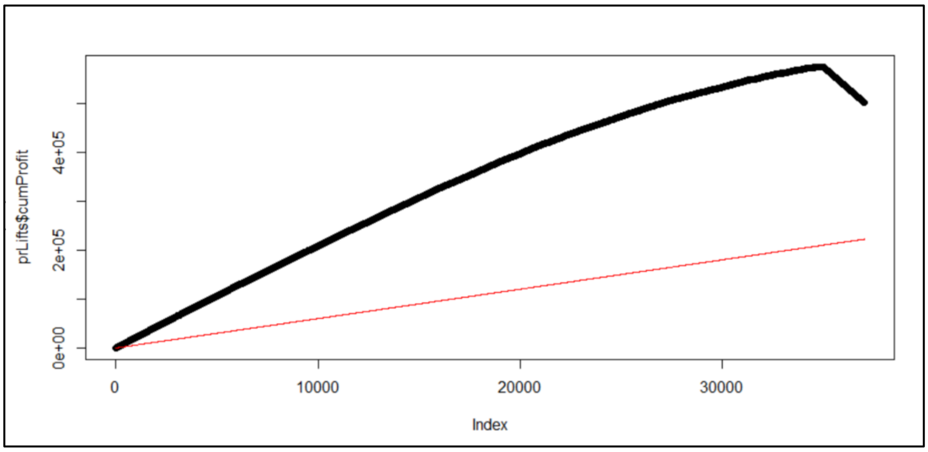
```
> avgReturnFP [1] 7.503414 (For Fully Paid)
```

```
> avgReturnCF [1] -12.21714 (For Charged Off)
```

The above values are multiplied by 3 (as we have to obtain return for the whole term –till maturity) and the values obtained are
COST VALUE -> (-36.65)
PROFIT VALUE -> 22.5
These values are used to obtain the cumulative profits and the plot of the cumulative profits v/s the count of records in test set is shown below-



When comparing the RF model and profits from bank CD, we can see that profit from RFmodel is exponential large. Hence, it is in preferable to not invest in back CD. Please refer to the graph below-



The threshold at which the maximum profit is obtained is 0.576.

```
> print(c(maxProfit = maxProfit, scoreTst = maxProfit_score))
 maxProfit   scoreTst 
577407.000      0.576
```

Appendix-

1. https://campus.datacamp.com/courses/machine-learning-toolbox/preprocessing-your-data?ex=13
2. https://www.kaggle.com/c/digit-recognizer/discussion/38768
3. https://rdrr.io/cran/caret/man/nearZeroVar.html
4. https://topepo.github.io/caret/pre-processing.html#nzv
5. http://information-gain.blogspot.com/2012/07/why-split-data-in-ratio-7030.html
6. http://www2.cs.uregina.ca/~dbd/cs831/notes/lift_chart/lift_chart.html
7. https://developers.google.com/machine-learning/crash-course/training-and-test-sets/splitting-data
8. https://statinfer.com/203-4-2-calculating-sensitivity-and-specificity-in-r/