Data Mining Assignment - 1

Lending club - Data Exploration - Part A

Name	UIN
Dhananjay Singh	668437546
Srinanda Kurapati	663244158
Sunny Patel	676645654

1. Describe the business model for online lending platforms like Lending Club. Consider the stakeholders and their roles, and what advantages Lending Club offers. What is the attraction for investors? How does the platform make money? (Not more than 1.5 pages, single spaced, 11 pt font. Please cite your sources).

Now-a-days there are many online lending platforms such as Lending club, SoFi, Up start, Prosper, Funding Circle and many more . Lending Club is one of the pioneers which are leading peer to peer lending platforms which lends loans to borrowers and takes money from the investors. Lending Club follows the multi-sided business model. A multi sided business model is one where the business identifies multiple customer groups to cater to. In most cases, it is two groups. The business is designed in a manner that connects the groups and also benefits both of them. In the case of Lending Club, the two customer parties are the Borrowers and Investors. Borrowers can be individuals that are applying for personal loans, education or health related loans. Even small-scale businesses can apply for loans in order to expand their business or buy new inventory. Another customer group is the Investors or Lenders that consists of high-net-worth individuals or even investment, insurance companies. They invest in loans by purchasing them in tokens or by means of investment funds. Hence, the stakeholders of Lending Club are its borrowers and investors. Lending Club does not have brick-and-mortar locations and hence transaction costs and interest rates are relatively low. And the stakeholders remunerate from the success of the project.

When borrowers take a loan, they are assigned an Annual Percentage Rate (APR) based on multiple factors and this APR is lesser than the average credit card APR thereby enabling borrowers to not just save money but also almost improve their credit. The Lending Club enables the borrowers to create personal loans between $1000 to $40000. As stated by them, their members save nearly $1000 in finance charges with a lot of them seeing an increase in their credit scores during the course of their personal loan. This set of customers have to only fill out a single application. Due to the company's foundation in technology, data and analytics, everything is processed and assessed online and factors like risk, credit score and interest rates are calculated immediately, thereby saving lots of time and paperwork.

Investors have the liberty to invest in loans by filtering through specific loan attributes. Due to its technology and automation, operational load is reduced. There is data transparency where investors get to see more than hundred data fields per loan. There is no commitment to minimum purchase and it is flexible to the investor. Features like efficient liquidity and same day settlement make it convenient for individuals/companies to make investments. Borrowers have stringent criteria to be able to take a loan, for instance an average borrower must have a 699 FICO score, 16.3 years of credit history, an 18.12% debt-to-income ratio, and a personal income of $75,055. Hence investors can set up a portfolio by issuing low interest loans and gain profits in the form of interest payments.

Lending club applies a set of fees for transactions. A service fee of 1% from borrower payments made during the due dates/grace periods. Up to 40% on all amounts collected on a delinquent loan (net of legal fees and expenses) to the extent any litigation has been initiated against the borrower, or 30% on all amounts collected on a delinquent loan in all cases not involving litigation. These fee rates are subjective and are generally a percentage of the money obtained back.

Sources:

- ***https://www.cleverism.com/company/lending-club/***
- ***https://www.moneyunder30.com/lending-club-investing***
- ***https://www.lendingclub.com/investing/marketplace-lending***

2.a)

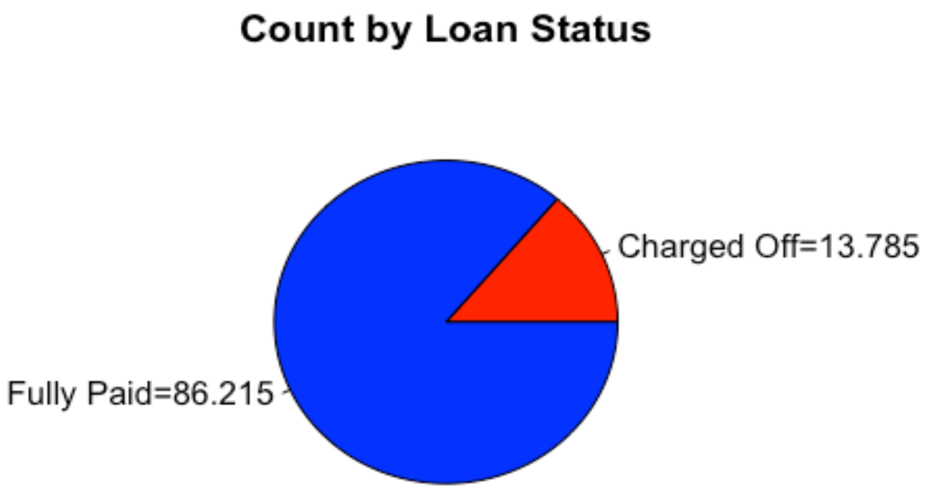
(i) What is the proportion of defaults ('charged off' vs 'fully paid' loans) in the data? How does default rate vary with loan grade? Does it vary with sub-grade? And is this what you would expect, and why?

We have used tables to show the count of both categories of loan status. We have also created a pie chart to display the Percent of each loan category in our dataset. Finally we have calculated the proportion of defaults, that is, "Charged Off Vs. Fully Paid".

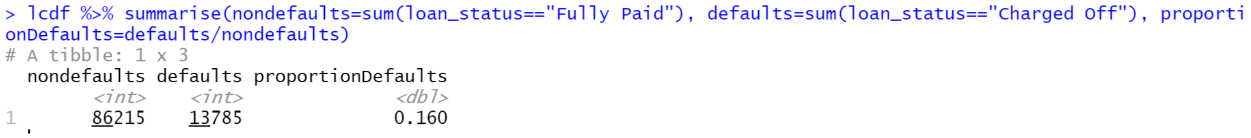
Default rate increases as the grade goes from A-G. Default rate increases as the risk of sub-grade increases. There is an exception with sub grades F1 and G3 wherein the default rate goes down compared to surrounding sub-grades. These are the outliers in our data.

This increase in default rate with increase in loan grade from A-G was expected because the loan grade A, being the safest loan, has the lowest interest rate. And as we go up from sub-grades A1-G5, the loan risk increases. This increase in default rates makes sense with an increase in riskiness of loan.

What is the proportion of defaults (‘charged off’ vs ‘fully paid’ loans) in the data?



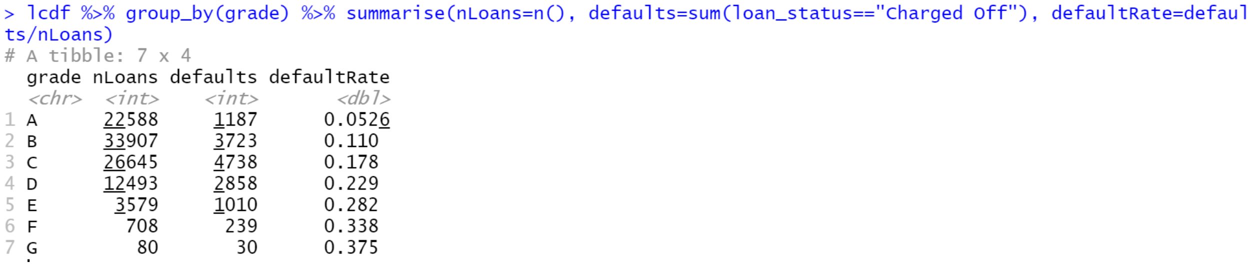
The proportion of defaults is the ratio of defaults to the non-defaults, which in this case is 13785/86215 = 0.15989



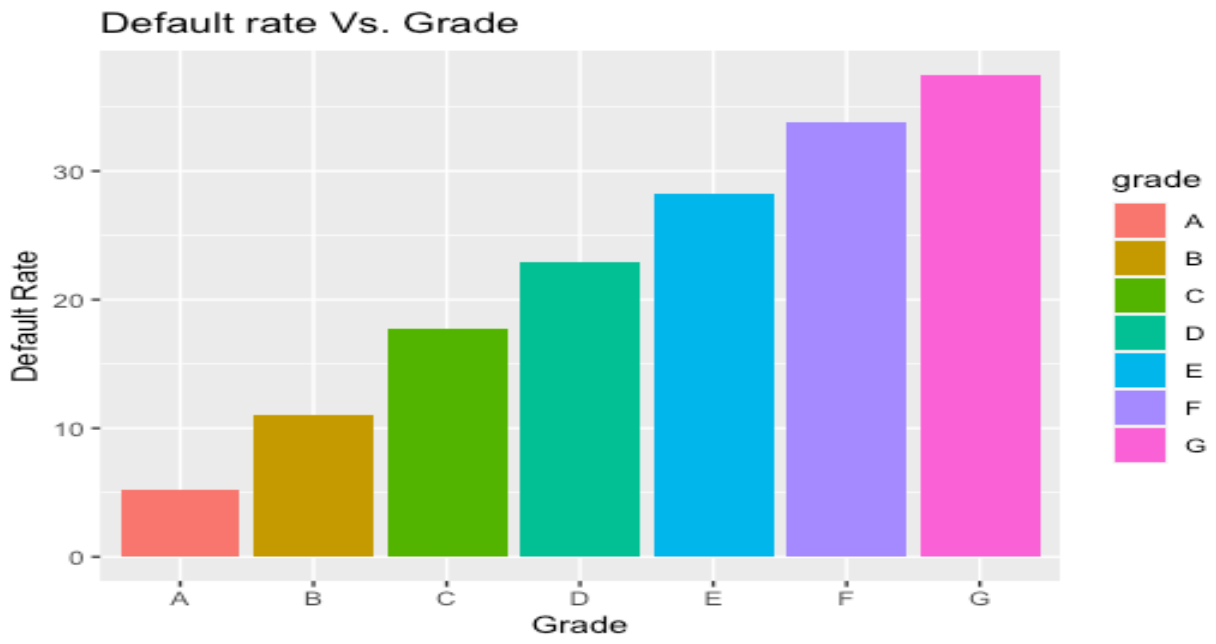
```
> lcdf %>% summarise(nondefaults=sum(loan_status=="Fully Paid"), defaults=sum(loan_status=="Charged Off"), proporti
onDefaults=defaults/nondefaults)
# A tibble: 1 x 3
  nondefaults defaults proportionDefaults
        <int>    <int>              <dbl>
1       86215    13785              0.160
```

How does the default rate vary with loan grade?

The default rate is the least for grade A loans. It goes on steadily increasing as we move across grades with the highest default rate for grade G loans.

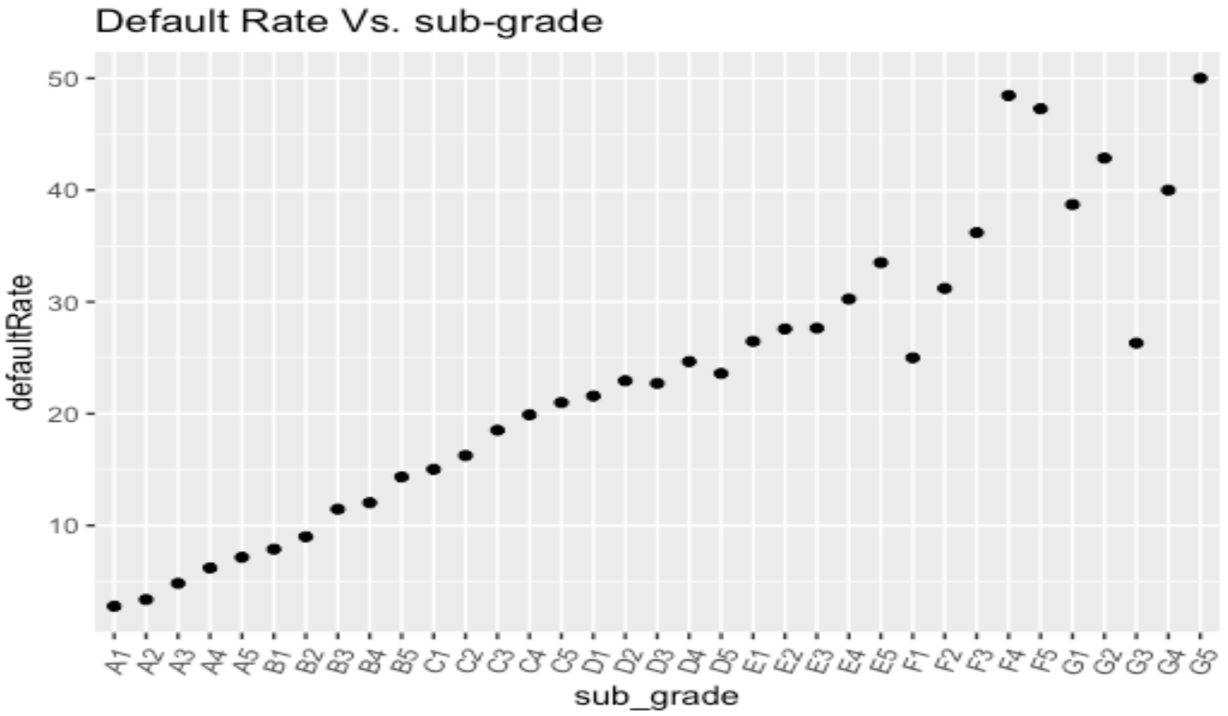


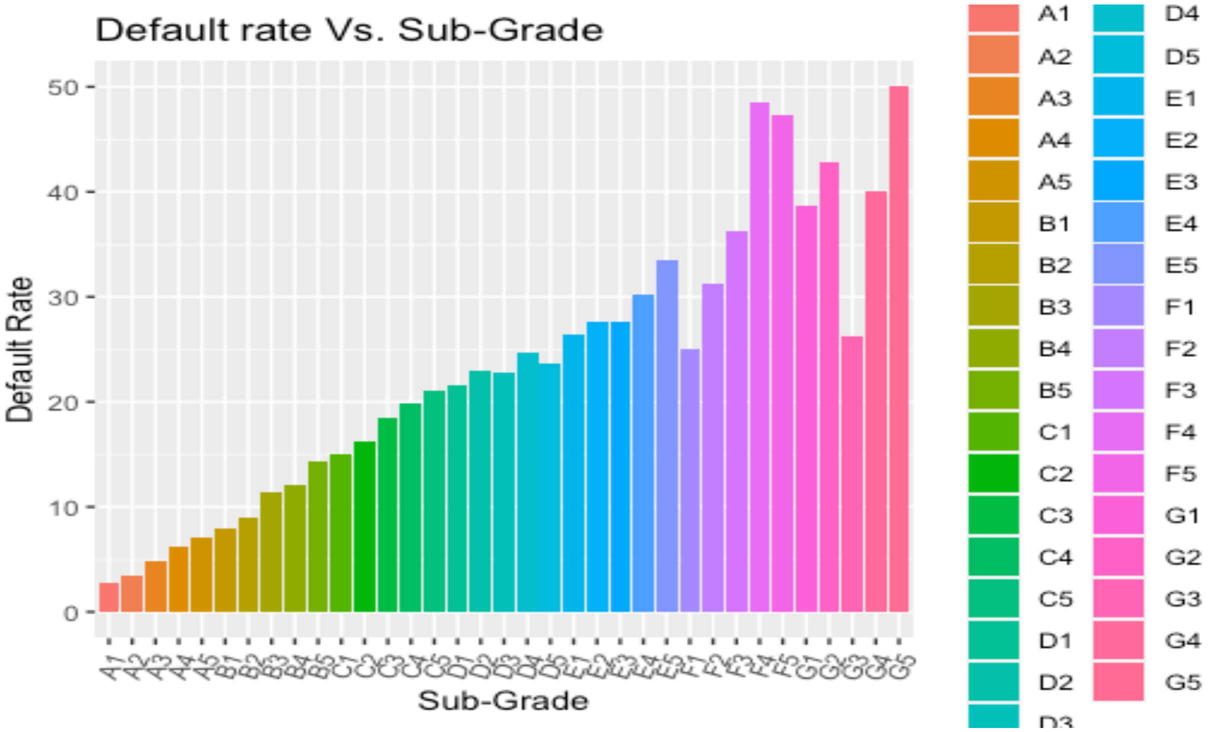
```
> lcdf %>% group_by(grade) %>% summarise(nLoans=n(), defaults=sum(loan_status=="Charged Off"), defaultRate=defaul
ts/nLoans)
# A tibble: 7 x 4
  grade nLoans defaults defaultRate
  <chr>  <int>    <int>       <dbl>
1 A      22588     1187      0.0526
2 B      33907     3723      0.110
3 C      26645     4738      0.178
4 D      12493     2858      0.229
5 E       3579     1010      0.282
6 F        708      239      0.338
7 G         80       30      0.375
```



Does it vary with sub-grade?

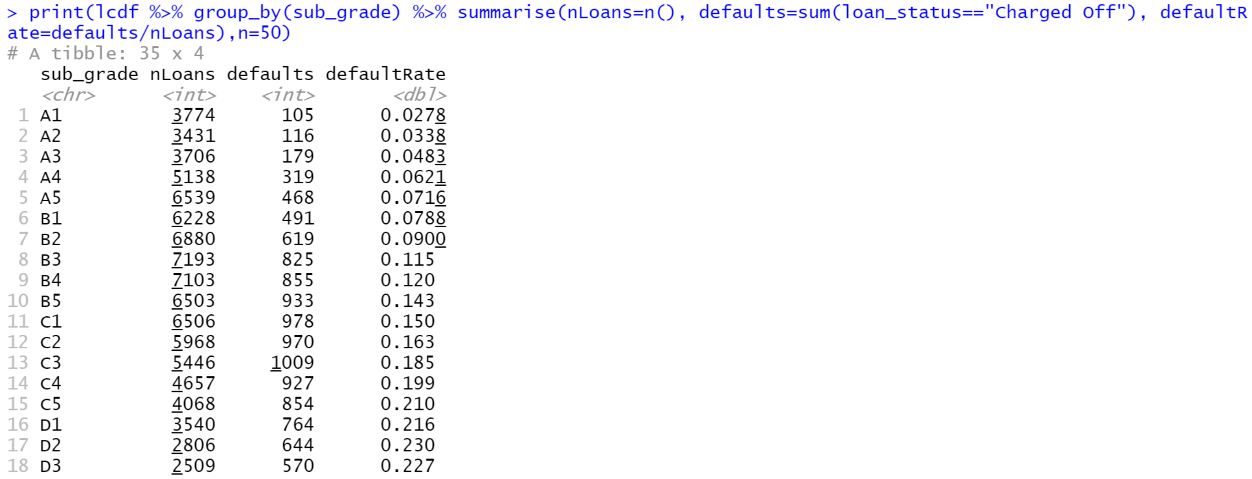
Within subgrades, the lowest default rate is at level 1 and highest at the highest level for that subgrade until loan C, post that we see slight variations. Overall, the lowest default rate is for A1 loan and the highest is for a G5 loan



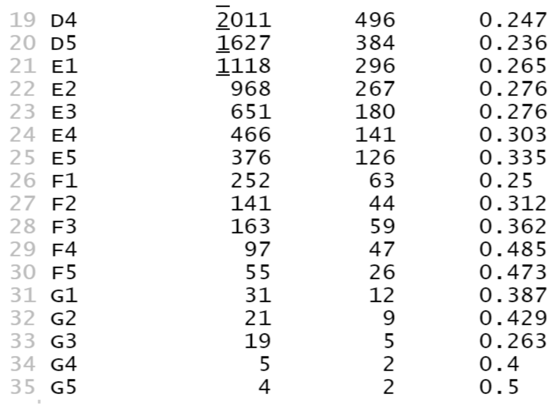


And is this what you would expect, and why?

This trend is what we expect because as per our observations, loan A interest rates are the lowest and with every subsequent category, the interest rates increase proving that they are less favorable. This goes hand in hand with the default rates. Probably due to high interest rates, the number of defaults also increase.



```
> print(lcdf %>% group_by(sub_grade) %>% summarise(nLoans=n(), defaults=sum(loan_status=="Charged Off"), defaultR
ate=defaults/nLoans),n=50)
# A tibble: 35 x 4
   sub_grade nLoans defaults defaultRate
   <chr>      <int>    <int>       <dbl>
 1 A1          3774      105      0.0278
 2 A2          3431      116      0.0338
 3 A3          3706      179      0.0483
 4 A4          5138      319      0.0621
 5 A5          6539      468      0.0716
 6 B1          6228      491      0.0788
 7 B2          6880      619      0.0900
 8 B3          7193      825      0.115
 9 B4          7103      855      0.120
10 B5          6503      933      0.143
11 C1          6506      978      0.150
12 C2          5968      970      0.163
13 C3          5446     1009      0.185
14 C4          4657      927      0.199
15 C5          4068      854      0.210
16 D1          3540      764      0.216
17 D2          2806      644      0.230
18 D3          2509      570      0.227
```



```
19 D4          2011      496      0.247
20 D5          1627      384      0.236
21 E1          1118      296      0.265
22 E2           968      267      0.276
23 E3           651      180      0.276
24 E4           466      141      0.303
25 E5           376      126      0.335
26 F1           252       63      0.25
27 F2           141       44      0.312
28 F3           163       59      0.362
29 F4            97       47      0.485
30 F5            55       26      0.473
31 G1            31       12      0.387
32 G2            21        9      0.429
33 G3            19        5      0.263
34 G4             5        2      0.4
35 G5             4        2      0.5
```

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? Look at the average, standard-deviation, min and max of interest rate by grade and subgrade. Is this what you expect, and why?

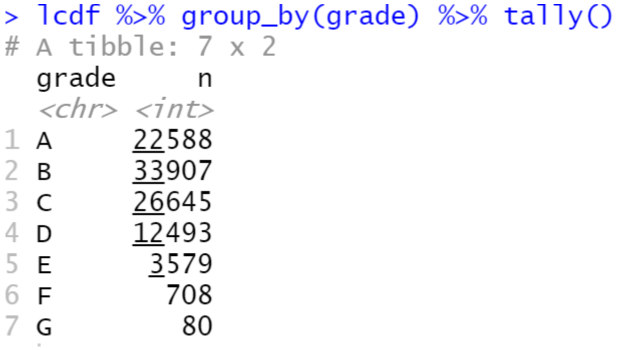
We have created a table displaying average, standard deviation, min, and max of interest rates and also displayed the same using box plots. This table also shows the number of loans in each grade.

Interest Rates increase with an increase in grade from A-G. This is expected because as the loan gets riskier the interest rates are bound to get higher.

Interest rates increase as the sub-grades go from A1-G5. This is expected with A1 being the safest loan and G5 being the riskiest loan. The min interest value of sub-grades B1-B5 < min Interest rates of sub-grade A4. Same pattern is visible for sub-grade C2 where min interest of C2 < min interest of C1.

How many loans are there in each grade?

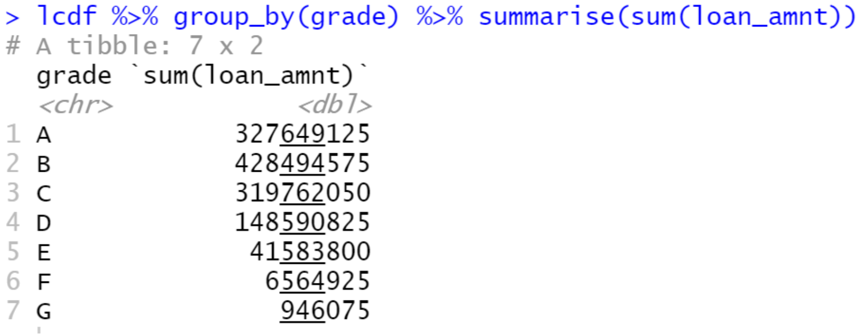
The number of loans according to grade are:



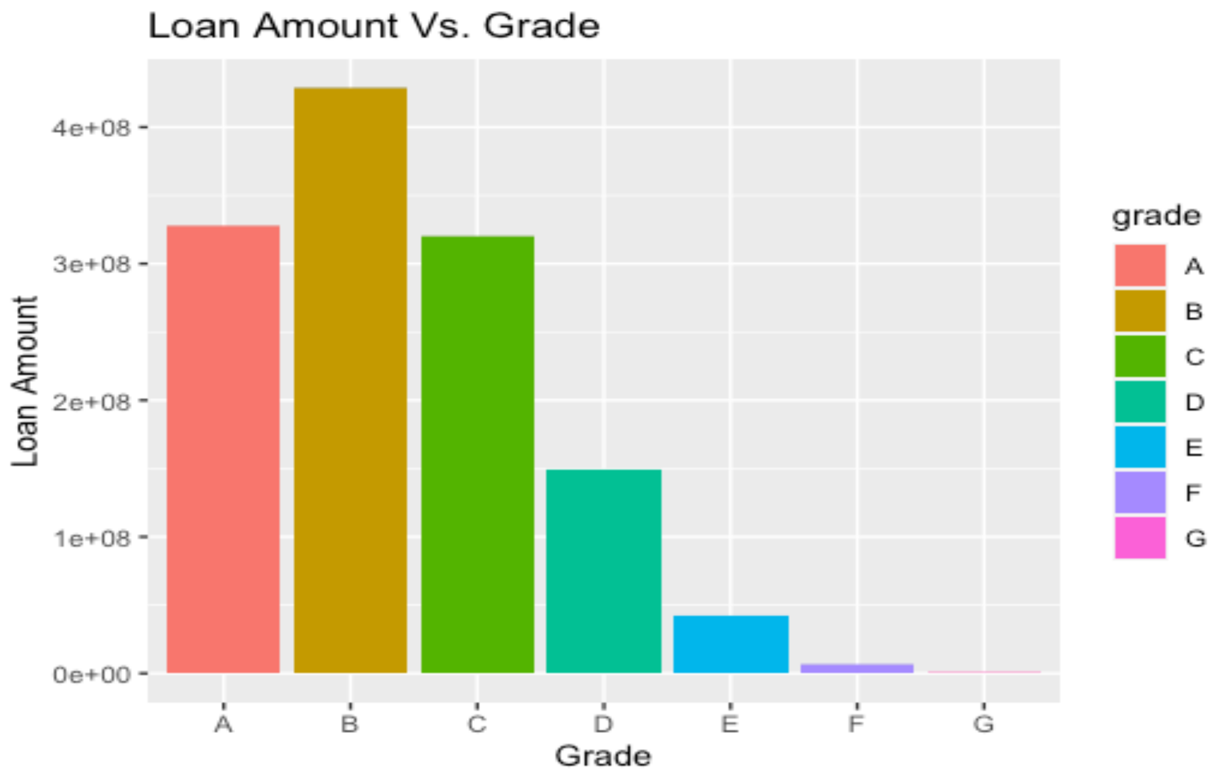
```
> lcdf %>% group_by(grade) %>% tally()
# A tibble: 7 x 2
  grade     n
  <chr> <int>
1 A     22588
2 B     33907
3 C     26645
4 D     12493
5 E      3579
6 F       708
7 G        80
```

And do loan amounts vary by grade?

We have calculated the mean of loan amounts for each grade. Looking at the following result, we can say that the loan amount varies by every grade. The highest loan amount is for grade B loans. Grade B and C loans almost at the same level. The lowest loan amount is for grade G loan.

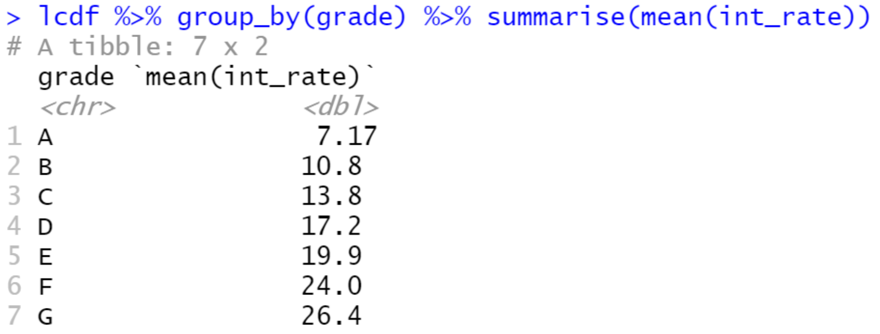


```
> lcdf %>% group_by(grade) %>% summarise(sum(loan_amnt))
# A tibble: 7 x 2
  grade `sum(loan_amnt)`
  <chr>            <dbl>
1 A            327649125
2 B            428494575
3 C            319762050
4 D            148590825
5 E             41583800
6 F              6564925
7 G               946075
```

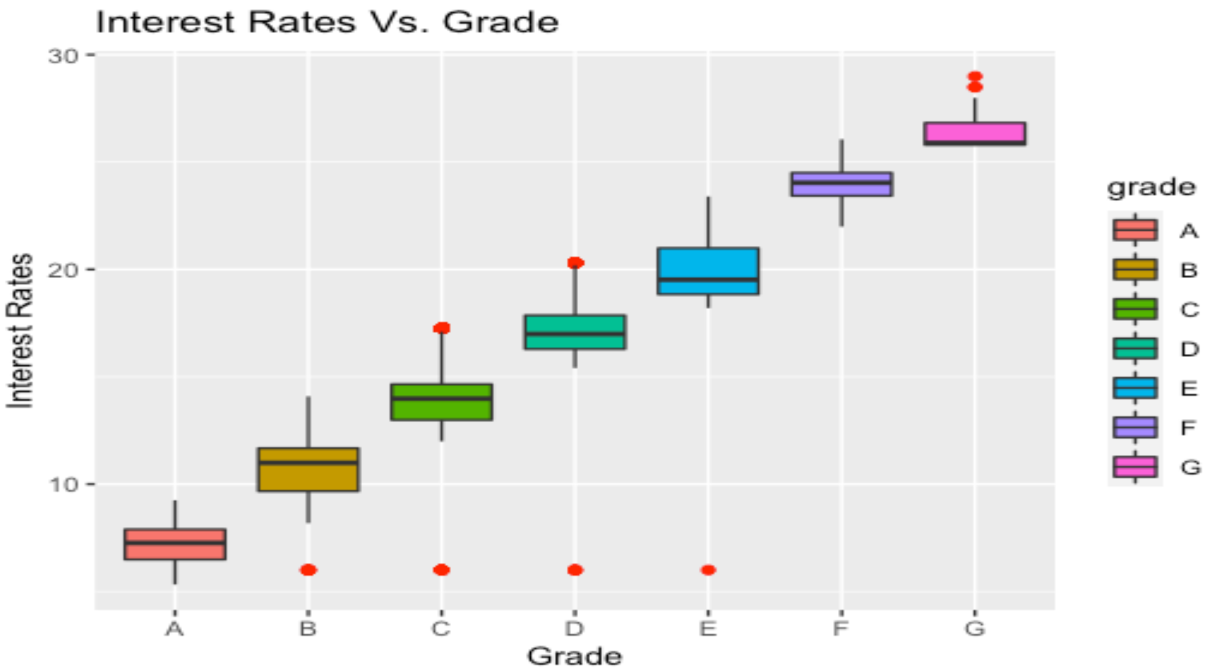


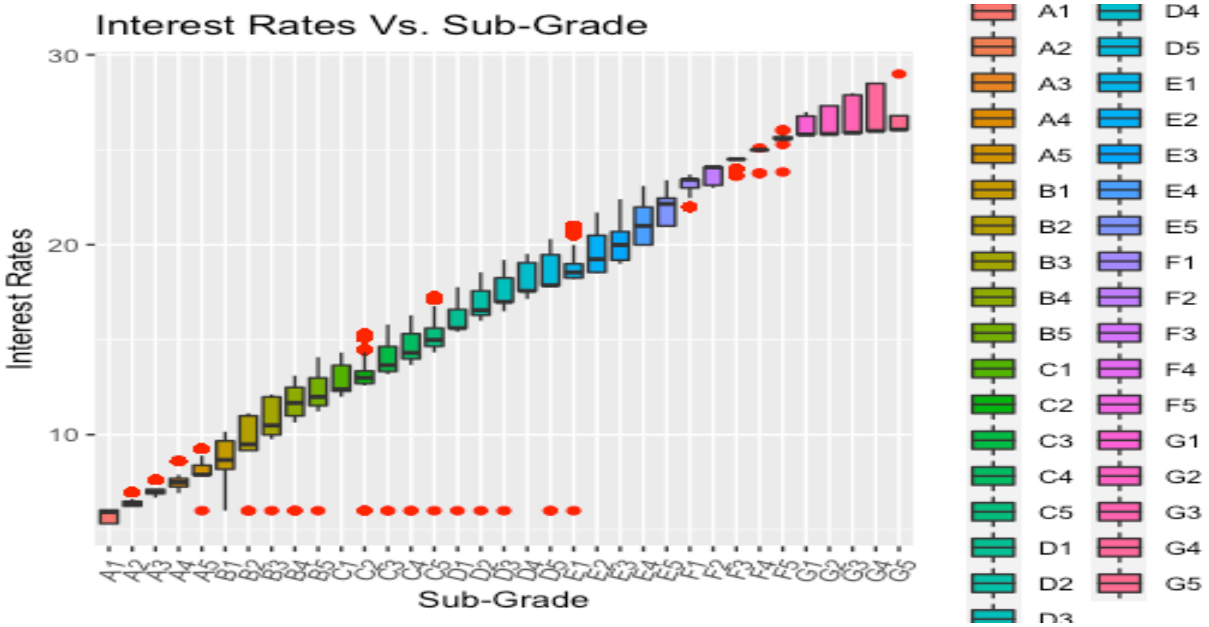
Does interest rate for loans vary with grade, subgrade?

The first snippet corresponds to mean interest rate per grade and the second corresponds to mean interest rate per subgrade. In both cases we see that for every row the interest rates are different. The lowest interest rate is for grade A loans. The highest interest rate is for grade G loan.

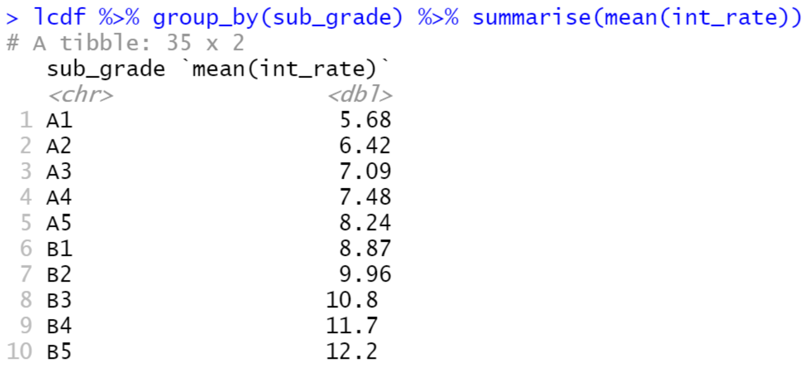


```
> lcdf %>% group_by(grade) %>% summarise(mean(int_rate))
# A tibble: 7 x 2
  grade `mean(int_rate)`
  <chr>            <dbl>
1 A                 7.17
2 B                10.8
3 C                13.8
4 D                17.2
5 E                19.9
6 F                24.0
7 G                26.4
```


Interest Rates Vs. Grade


Interest Rates Vs. Sub-Grade

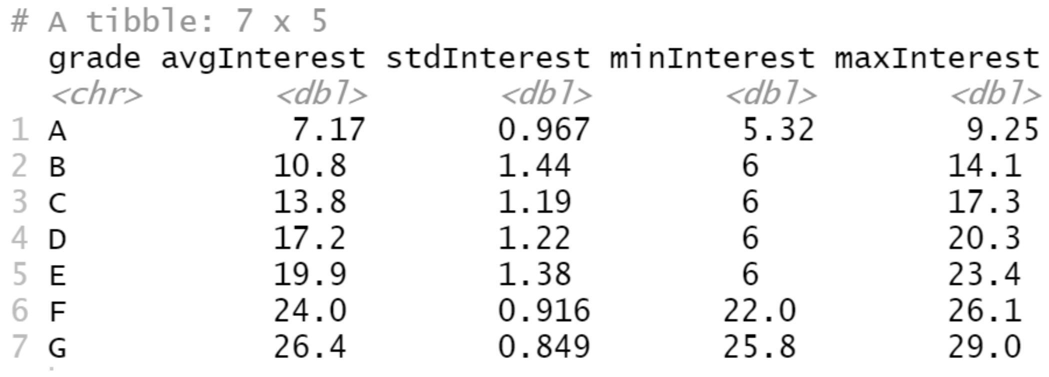
When you analyse by subgrade, A1 has the best interest rate (A low value, which is what borrowers expect). At each subgrade, the level 1 value is the best interest rate.



```
> lcdf %>% group_by(sub_grade) %>% summarise(mean(int_rate))
# A tibble: 35 x 2
   sub_grade `mean(int_rate)`
   <chr>                <dbl>
 1 A1                    5.68
 2 A2                    6.42
 3 A3                    7.09
 4 A4                    7.48
 5 A5                    8.24
 6 B1                    8.87
 7 B2                    9.96
 8 B3                   10.8 
 9 B4                   11.7 
10 B5                   12.2 
```

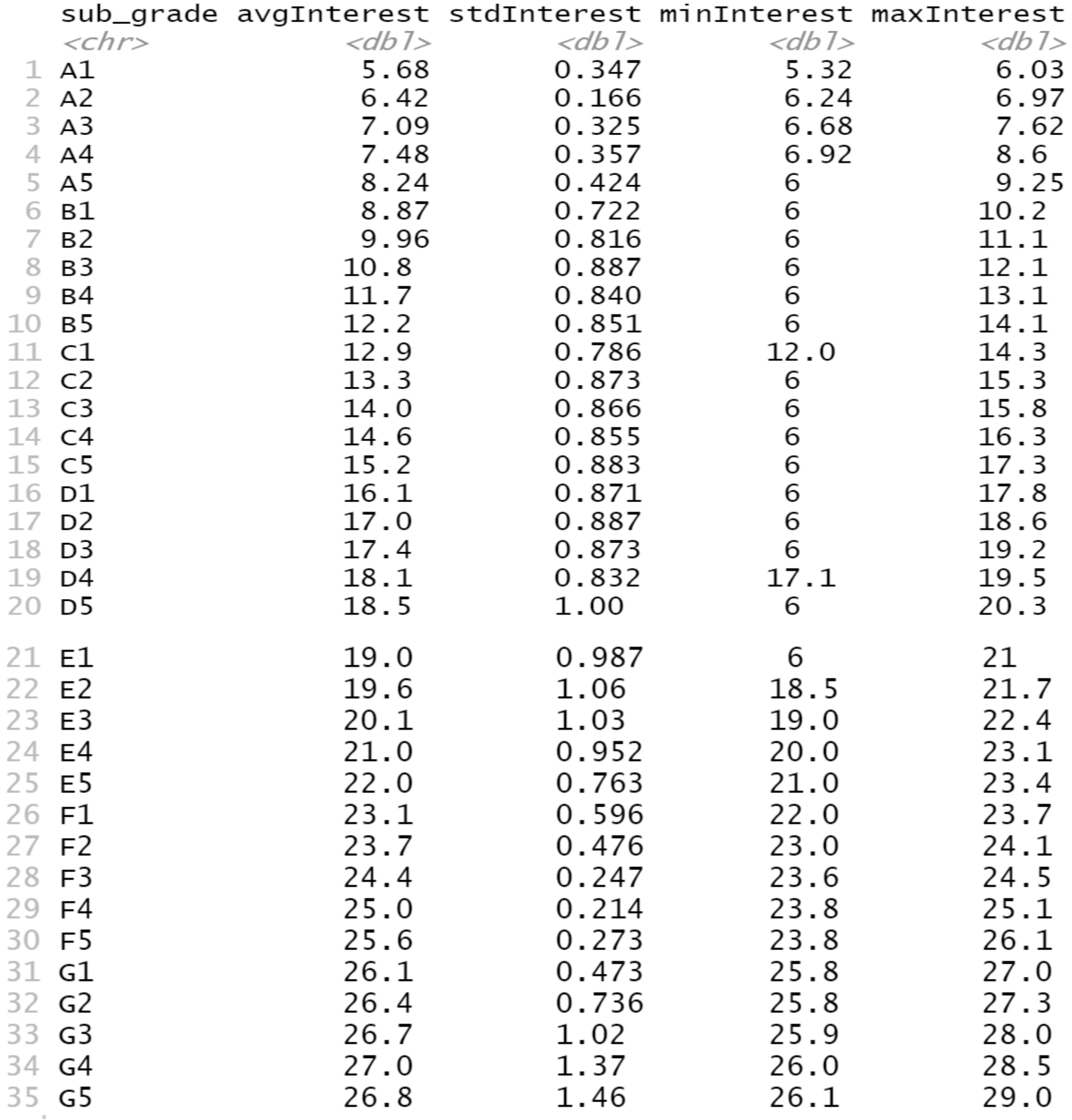
Look at the average, standard-deviation, min and max of interest rate by grade and subgrade. Is this what you expect, and why?

When grouped by grade, the average, standard deviation, min and max interest rates are:



```
# A tibble: 7 x 5
  grade avgInterest stdInterest minInterest maxInterest
  <chr>       <dbl>       <dbl>       <dbl>       <dbl>
1 A            7.17       0.967        5.32        9.25
2 B           10.8        1.44         6          14.1 
3 C           13.8        1.19         6          17.3 
4 D           17.2        1.22         6          20.3 
5 E           19.9        1.38         6          23.4 
6 F           24.0        0.916       22.0        26.1 
7 G           26.4        0.849       25.8        29.0 
```

When grouped by sub_grade, the average, standard deviation, min and max interest rates are:



```
   sub_grade avgInterest stdInterest minInterest maxInterest
   <chr>           <dbl>       <dbl>       <dbl>       <dbl>
 1 A1               5.68       0.347        5.32        6.03
 2 A2               6.42       0.166        6.24        6.97
 3 A3               7.09       0.325        6.68        7.62
 4 A4               7.48       0.357        6.92        8.6 
 5 A5               8.24       0.424        6           9.25
 6 B1               8.87       0.722        6          10.2 
 7 B2               9.96       0.816        6          11.1 
 8 B3              10.8        0.887        6          12.1 
 9 B4              11.7        0.840        6          13.1 
10 B5              12.2        0.851        6          14.1 
11 C1              12.9        0.786       12.0        14.3 
12 C2              13.3        0.873        6          15.3 
13 C3              14.0        0.866        6          15.8 
14 C4              14.6        0.855        6          16.3 
15 C5              15.2        0.883        6          17.3 
16 D1              16.1        0.871        6          17.8 
17 D2              17.0        0.887        6          18.6 
18 D3              17.4        0.873        6          19.2 
19 D4              18.1        0.832       17.1        19.5 
20 D5              18.5        1.00         6          20.3 

21 E1              19.0        0.987        6          21   
22 E2              19.6        1.06        18.5        21.7 
23 E3              20.1        1.03        19.0        22.4 
24 E4              21.0        0.952       20.0        23.1 
25 E5              22.0        0.763       21.0        23.4 
26 F1              23.1        0.596       22.0        23.7 
27 F2              23.7        0.476       23.0        24.1 
28 F3              24.4        0.247       23.6        24.5 
29 F4              25.0        0.214       23.8        25.1 
30 F5              25.6        0.273       23.8        26.1 
31 G1              26.1        0.473       25.8        27.0 
32 G2              26.4        0.736       25.8        27.3 
33 G3              26.7        1.02        25.9        28.0 
34 G4              27.0        1.37        26.0        28.5 
35 G5              26.8        1.46        26.1        29.0 
```

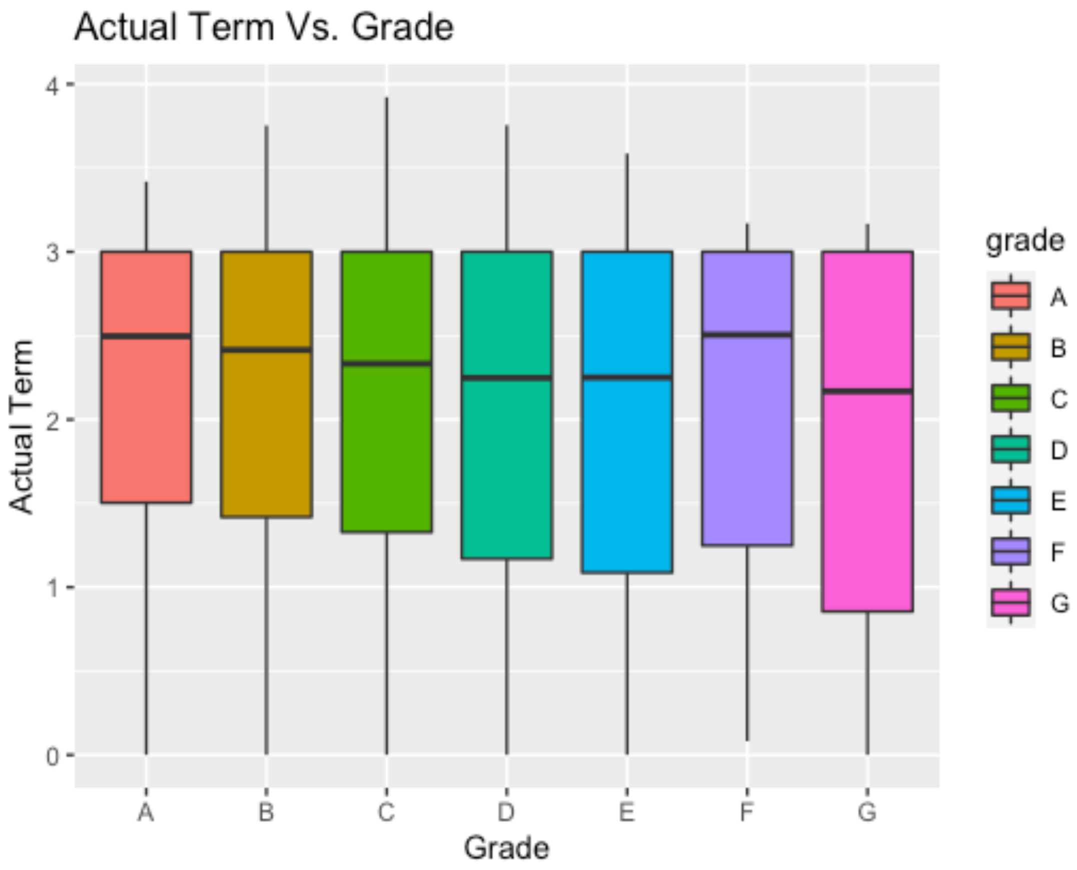
(iii) For loans which are fully paid back, how does the time-to-full-payoff vary? For this, calculate the 'actual term' (issue-date to last-payment-date) for all loans. How does this actual-term vary by loan grade (a box-plot can help visualize this).

For loans which are fully paid back, how does the time-to-full-payoff vary?

For fully paid loans, the average number of loans are paid back within 2.2 years. This pattern is the same for all grades from A-G.

How does this actual-term vary by loan grade (a box-plot can help visualize this).

We have used boxplot to display Actual term Vs. Grade.



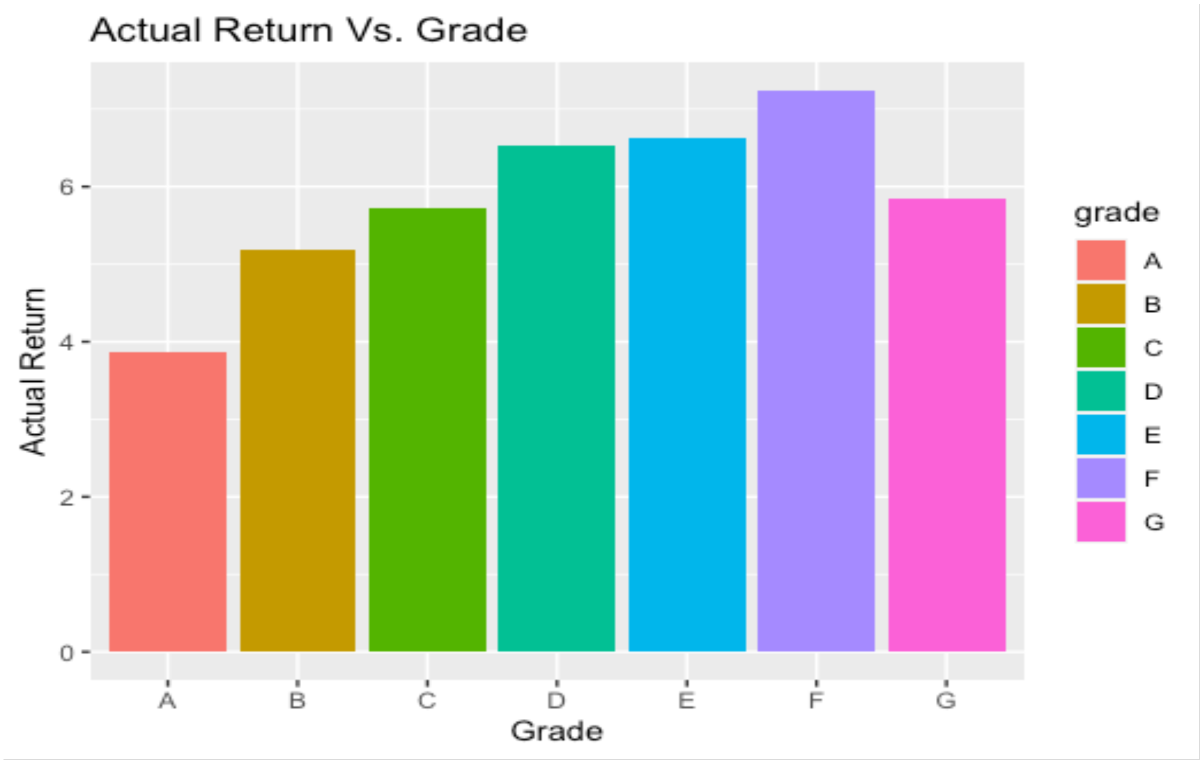
(iv) Calculate the annual return. Show how you calculate the percentage annual return. Is there any return from loans which are 'charged off'? Explain. How does return from charged -off loans vary by loan grade? Compare the average return values with the average interest_rate on loans – do you notice any differences, and how do you explain this? How do returns vary by grade, and by sub-grade. If you wanted to invest in loans based on this data exploration, which loans would you invest in?

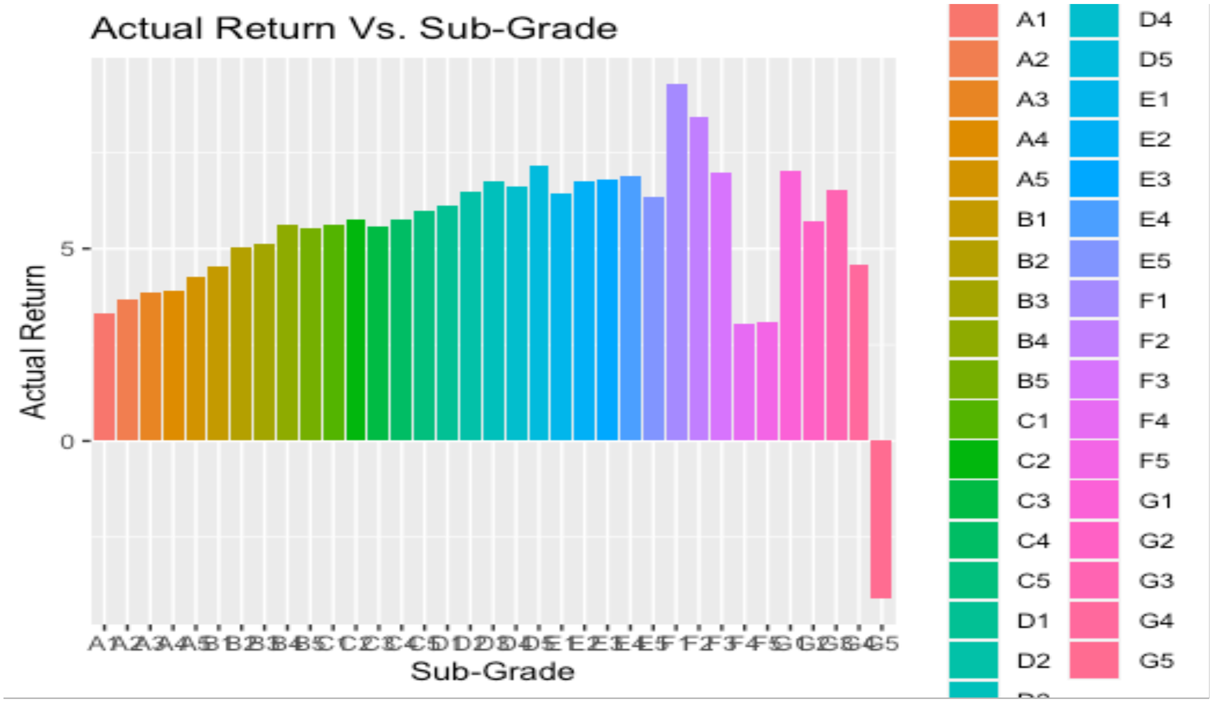
Actual Annual Return increases with loan grades. Highest average return is for grade F. This is possible because the interest rate is high for this grade loan. Actual Annual Return is highest for subgrade F1.

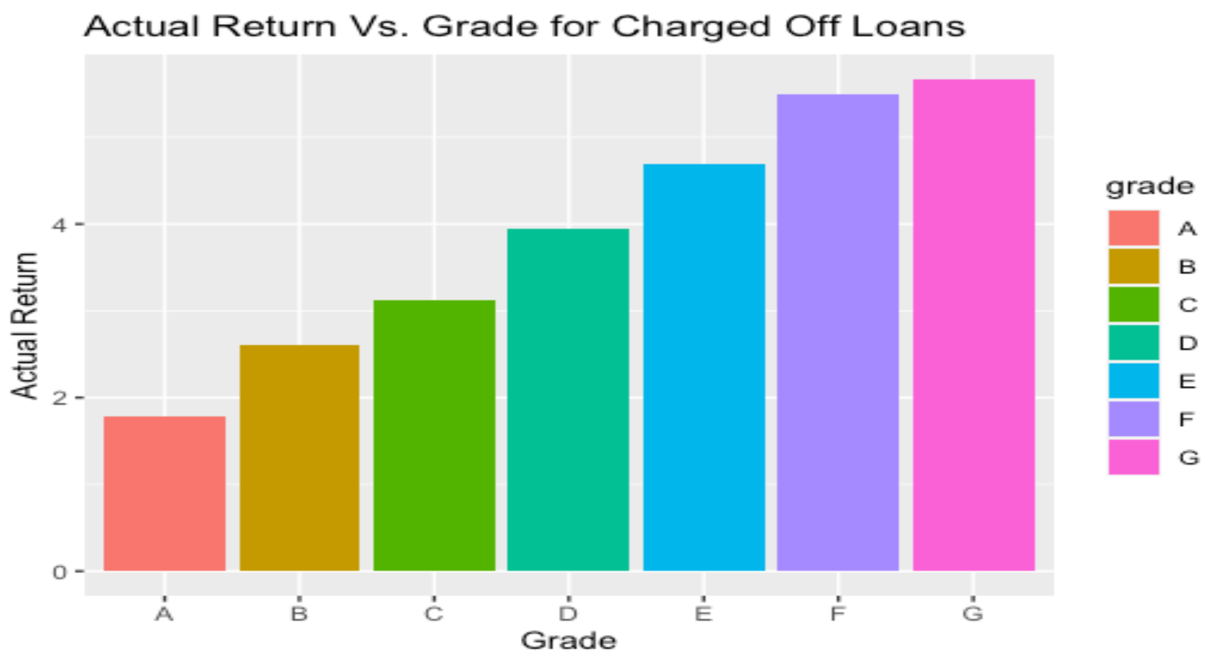
We can see some of the Charged Off loans giving us returns.

I would invest on loan F1 which has the highest return. F1 subgrade loan(outlier) also has high interest rate along with low default rate(25%) compared to near-by sub-grades loans.

Another loan to consider is D5 which has a default rate of 23.6% and has the third highest return.


Actual Return Vs. Grade
Actual Return
0
2
4
6
A
B
C
D
E
F
G
Grade
grade
A
B
C
D
E
F
G

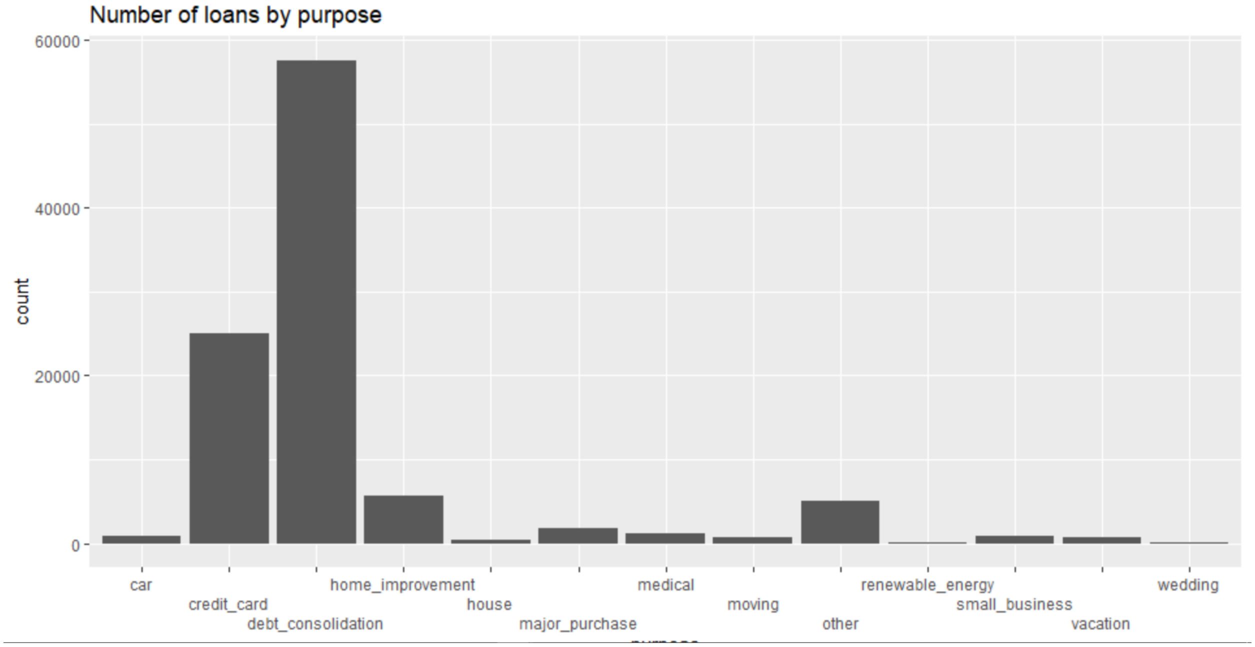

Actual Return Vs. Sub-Grade
Actual Return
0
5
A1A2A3A4A5B1B2B3B4B5C1C2C3C4C5D1D2D3D4D5E1E2E3E4E5F1F2F3F4F5G1G2G3G4G5
Sub-Grade
A1
A2
A3
A4
A5
B1
B2
B3
B4
B5
C1
C2
C3
C4
C5
D1
D2
D4
D5
E1
E2
E3
E4
E5
F1
F2
F3
F4
F5
G1
G2
G3
G4
G5



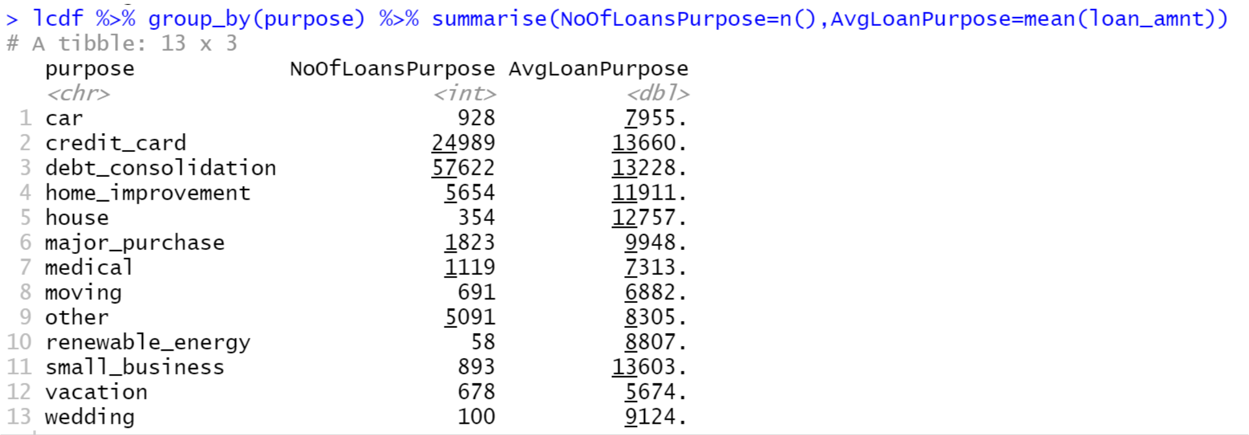
(v)What are people borrowing money for (purpose)? Examine how many loans, average amounts, etc. by purpose? Do loan amounts vary by purpose? Do defaults vary by purpose? Does loan-grade assigned by Lending Club vary by purpose?

What are people borrowing money for (purpose)?

Based on the result below, we can see that people have borrowed many for a variety of reasons, the most common reason being for debit_consolidation followed by credit card payments.



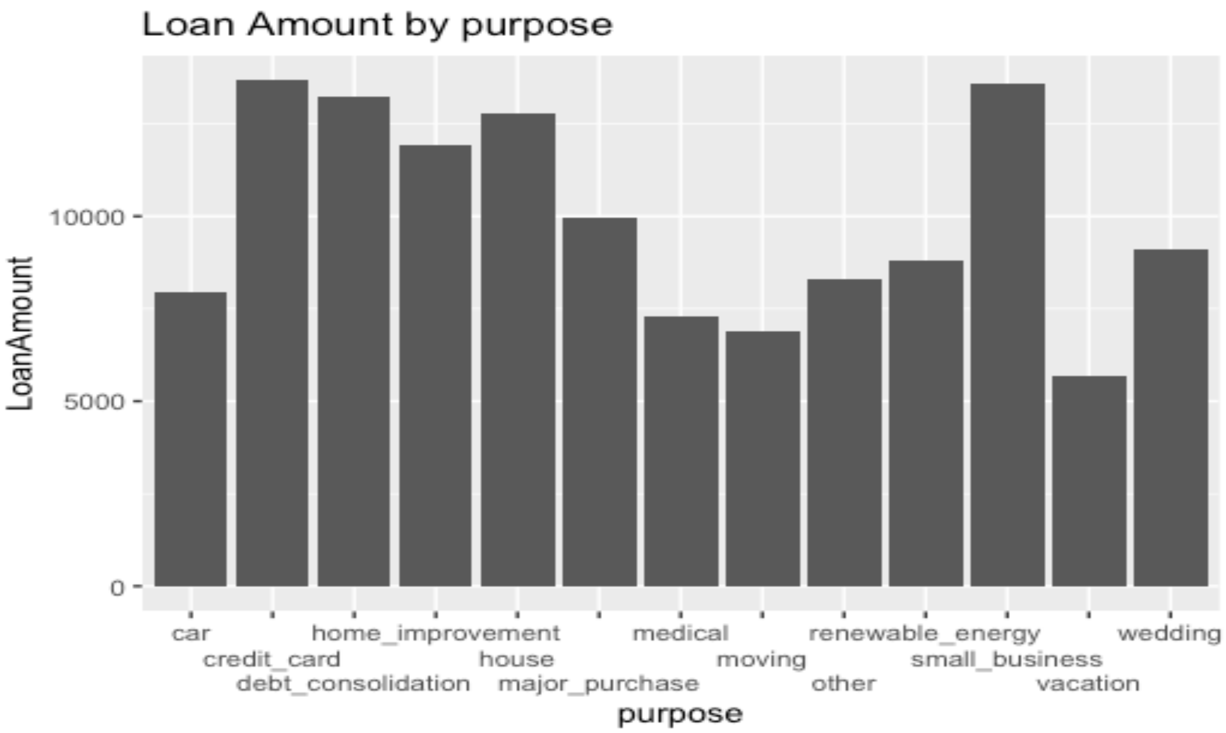
Examine how many loans, average amounts, etc. by purpose?



```
> lcdf %>% group_by(purpose) %>% summarise(NoOfLoansPurpose=n(),AvgLoanPurpose=mean(loan_amnt))
# A tibble: 13 x 3
   purpose            NoOfLoansPurpose AvgLoanPurpose
   <chr>                         <int>          <dbl>
 1 car                             928          7955.
 2 credit_card                   24989         13660.
 3 debt_consolidation            57622         13228.
 4 home_improvement               5654         11911.
 5 house                           354         12757.
 6 major_purchase                 1823          9948.
 7 medical                        1119          7313.
 8 moving                          691          6882.
 9 other                          5091          8305.
10 renewable_energy                 58          8807.
11 small_business                  893         13603.
12 vacation                        678          5674.
13 wedding                         100          9124.
```

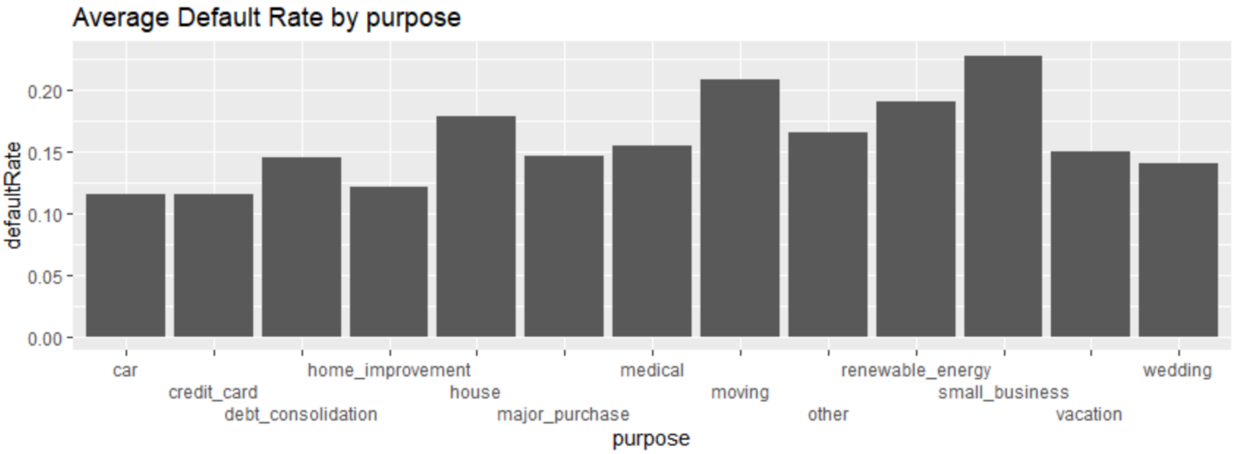
Do loan amounts vary by purpose?

Average loan amounts vary by purpose as follows. The values are high for credit card payments and small business loans followed by debt consolidation.



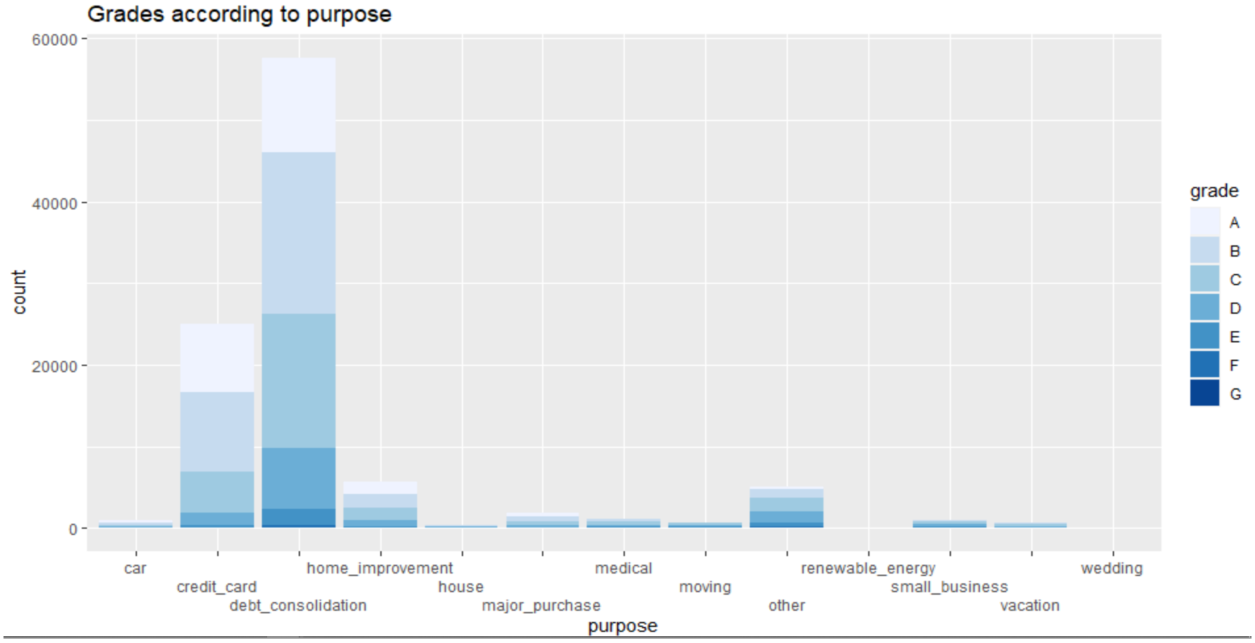
Do defaults vary by purpose?

Small businesses have the highest default rate followed by loans taken for moving. The least default rates are for car and credit card loans.



Does loan-grade assigned by Lending Club vary by purpose?

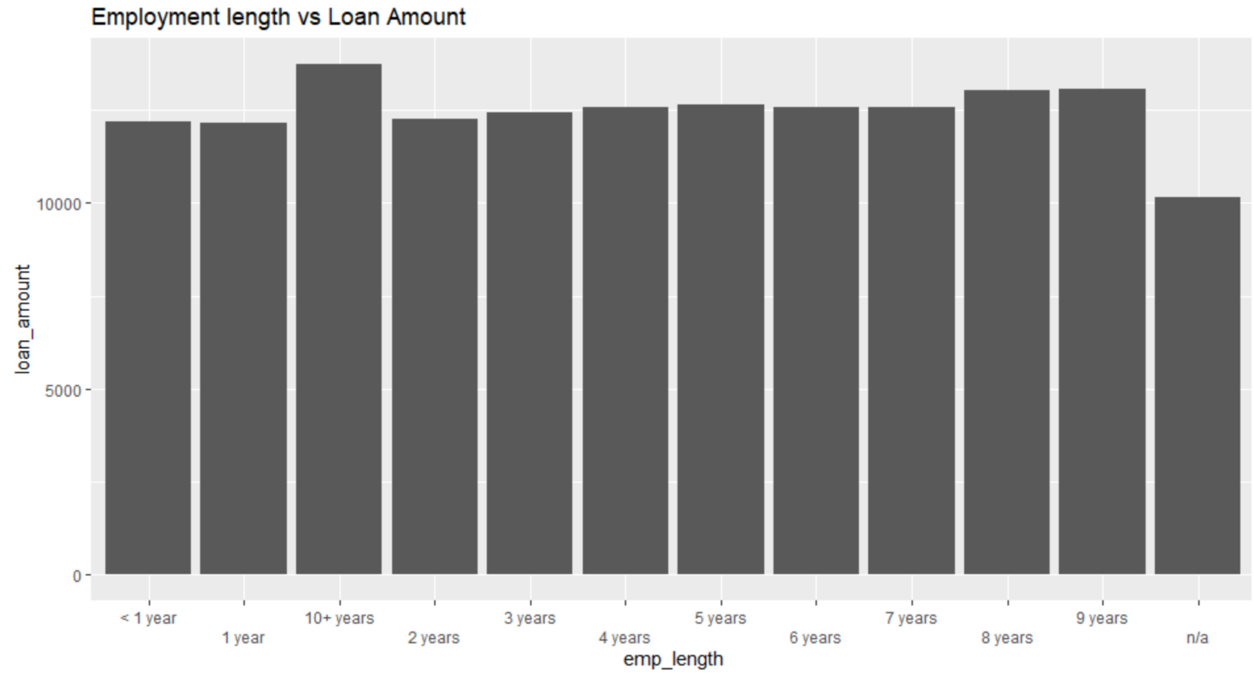
As one can observe, the maximum proportion of grade A loans are based on home improvement, followed credit card.



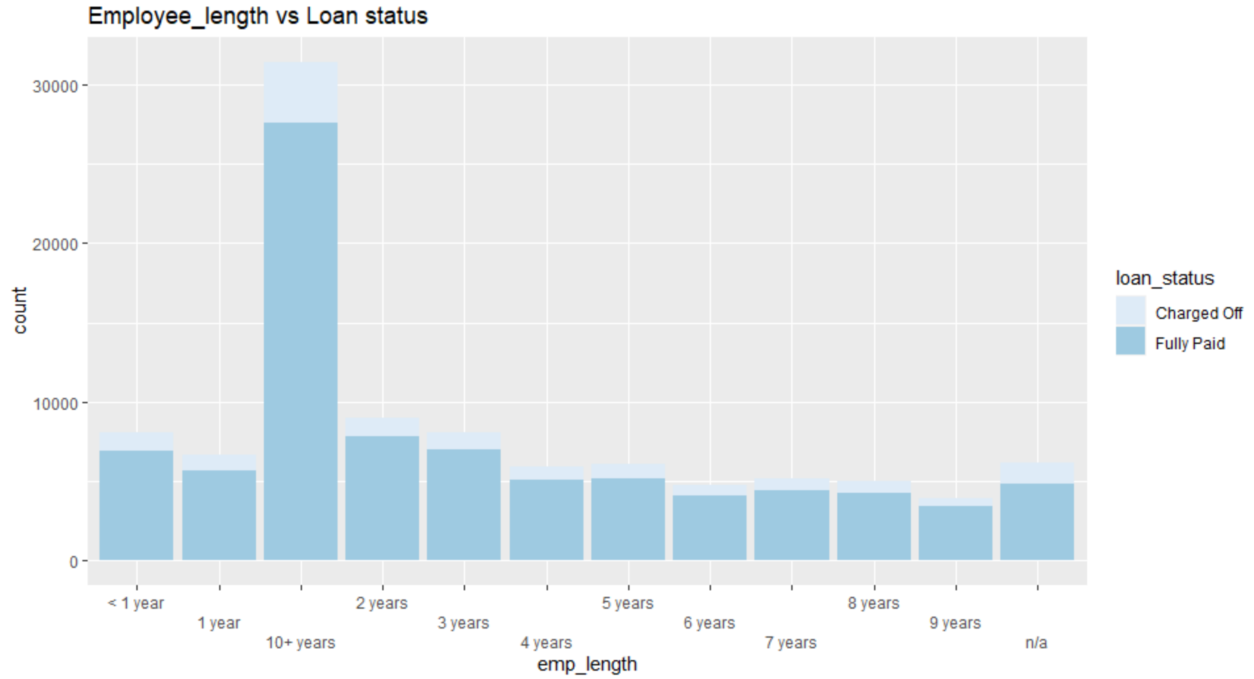
(vi) Consider some borrower characteristics like employment-length, annual-income, fico-scores (low, high). How do these relate to loan attributes like, for example, loan_amout, loan_status, grade, purpose, actual return, etc.

Loan grade gets worse from A-G as annual income decreases. Mean annual income is higher for Fully Paid loans.

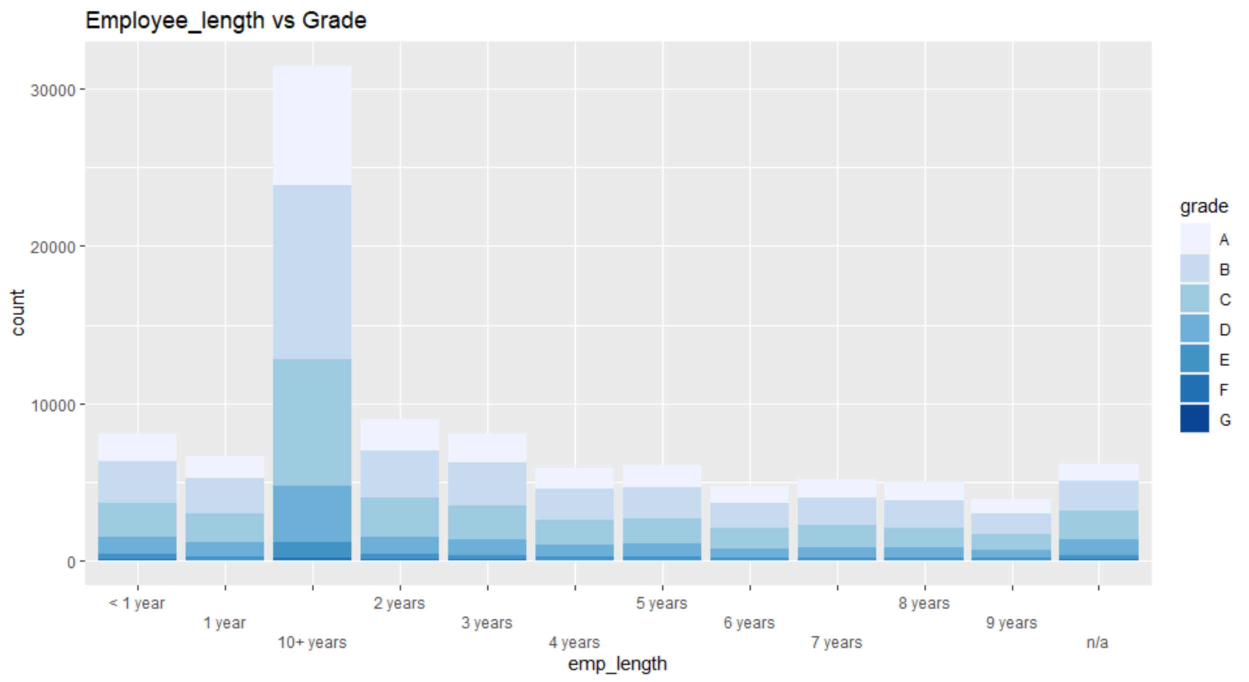
Employee Length vs Loan Amount



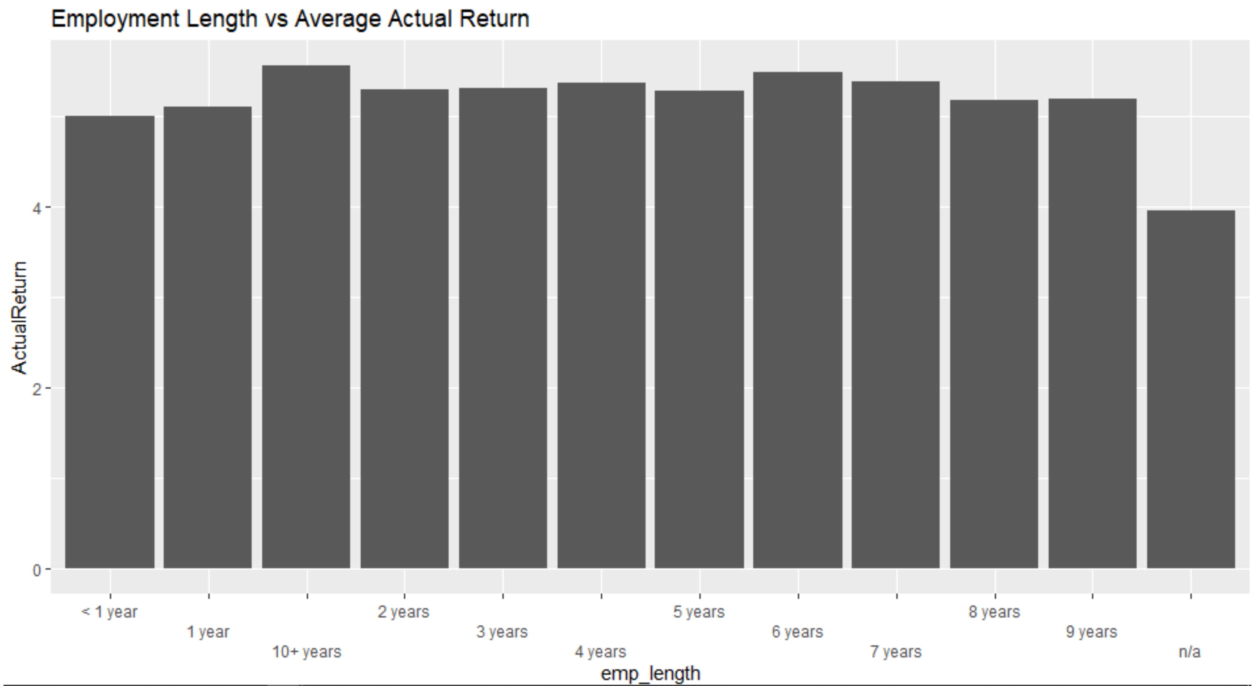
Employee Length vs Loan status



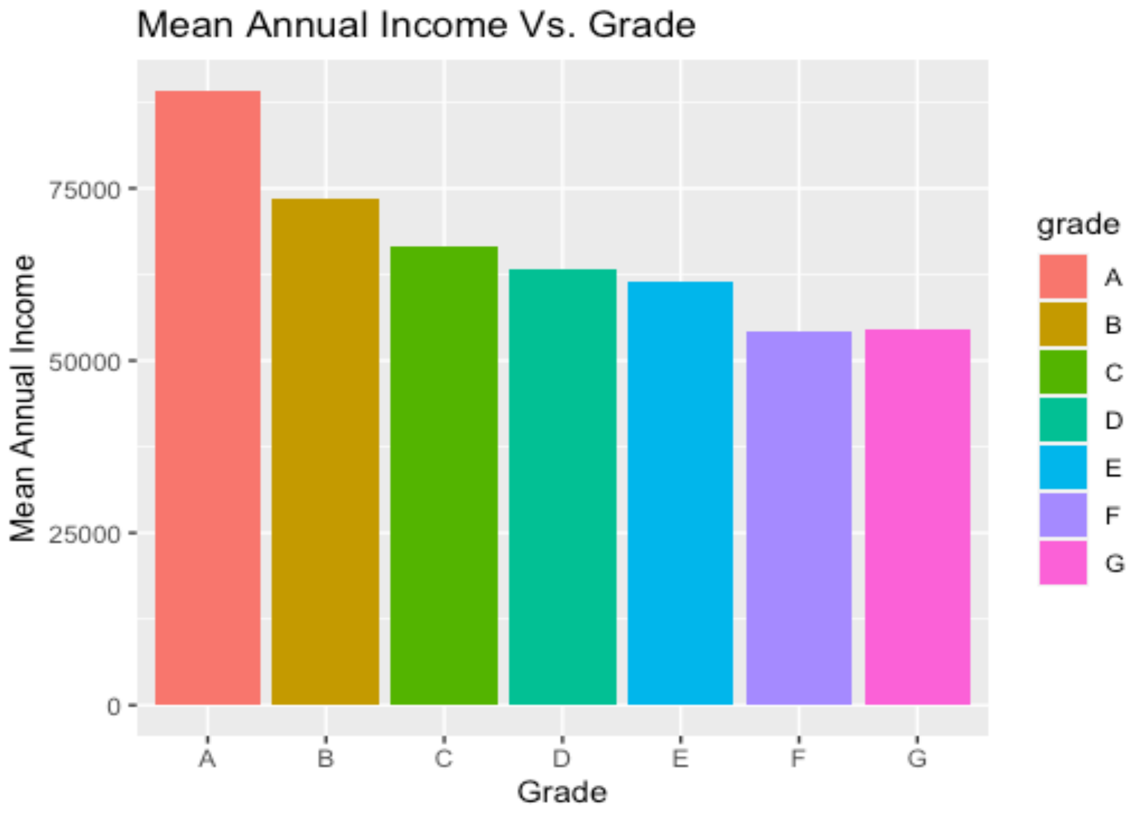
Employee Length vs Grade



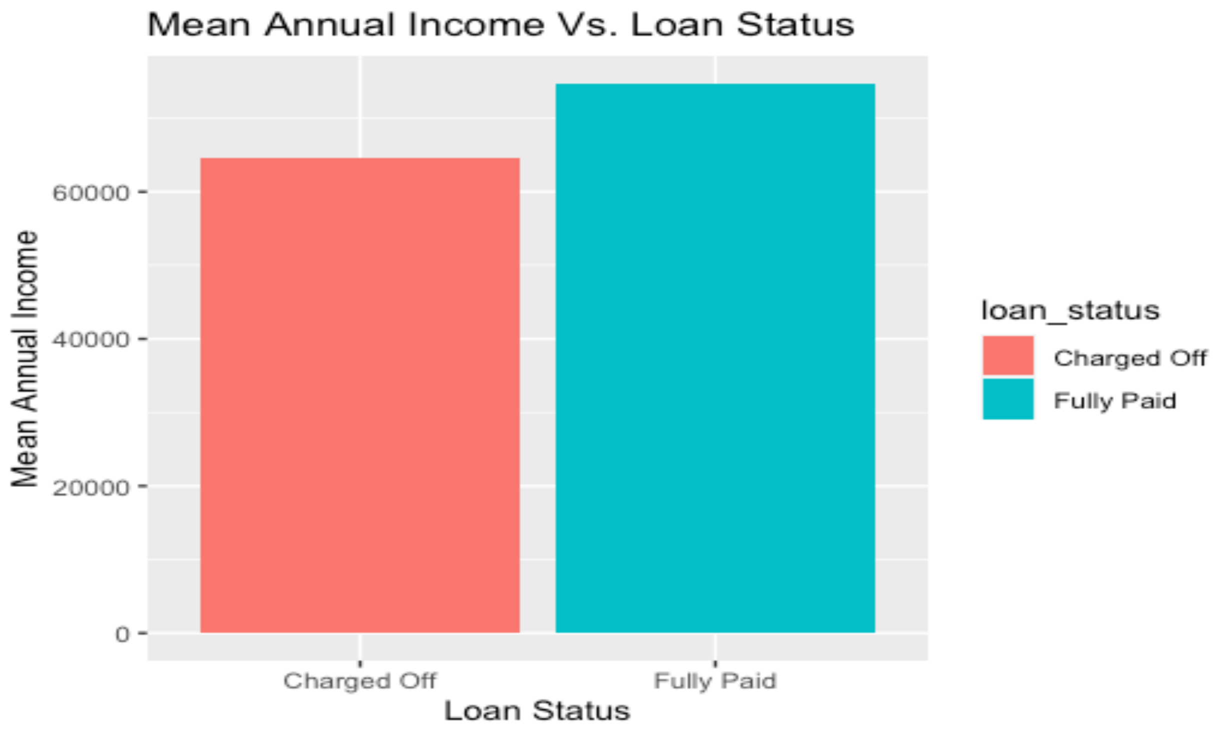
Employee Length vs Actual Return



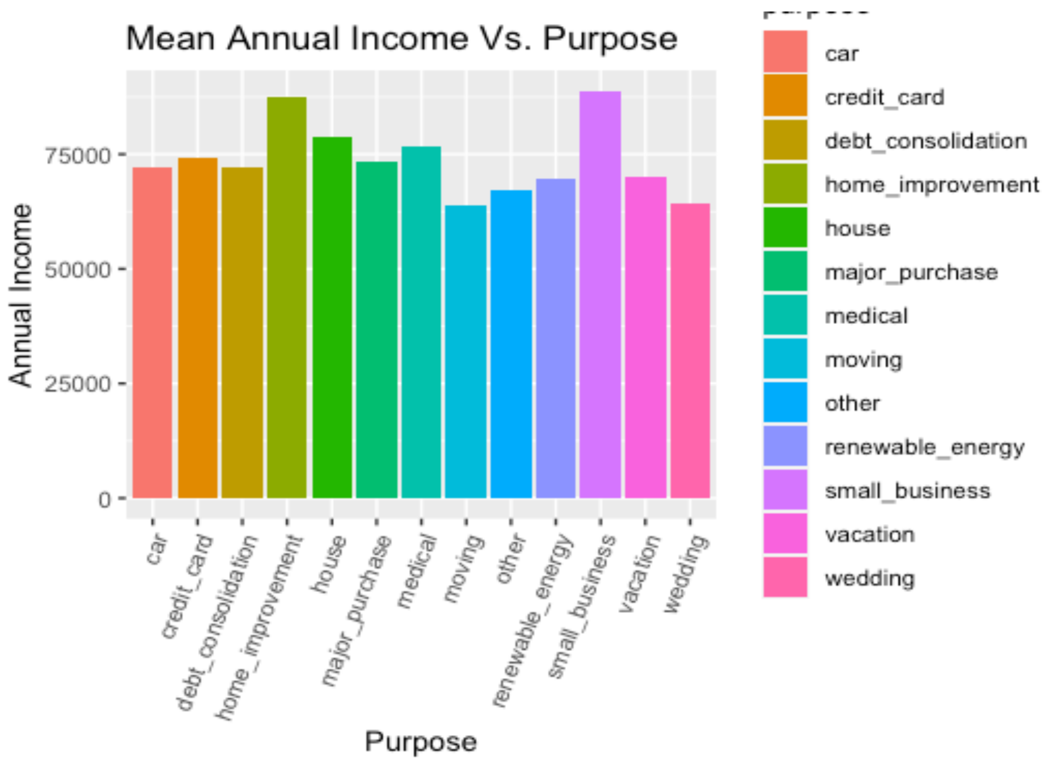
Mean annual income vs grade



Mean annual income vs loan status



Mean annual income vs purpose



(vii) Generate some (at least 3) new derived attributes which you think may be useful for predicting default., and explain what these are. For these, do an analysis as in the questions above (as reasonable based on the derived variables).

We have the following derived attributes

1 - Proportion of satisfactory bankcard accounts [prop_stat_bankcard_acc]

This ratio shows how good of a relationship you have with Lending Club by paying off the loans. Borrowers with a high proportion of satisfactory bankcard accounts are more likely to pay off their loan.

2 - Length of borrower's history with lending Club [bor_hist_len_lc]

Greater the borrower's history, less risky is the loan for that borrower based on the fact that he has been with Lending Club for a long time.

3 - Ratio of openAccounts to totalAccounts [openacc_totacc_rat]

This shows how many open accounts the borrower has in proportion to total accounts. A high ratio can signify borrowers inability to pay off loans.

(b) Summarize your conclusions and main themes from your analyses

The default rate is the least for grade A loans. It goes on steadily increasing as we move across grades with the highest default rate for grade G loans. Within subgrades, the lowest default rate is at level 1 and highest at the highest level for that subgrade until loan C, post that we see slight variations. Overall, the lowest default rate is for A1 loans and the highest is for a G5 loan. This trend is what we expect because as per our observations, loan A interest rates are the lowest and with every subsequent category, the interest rates increase proving that they are less favorable. This goes hand in hand with the default rates. Probably due to high interest rates, the number of defaults also increased. We have calculated the mean of loan amounts for each grade. Looking at the following result, we can say that the loan amount varies by every grade. The highest loan amount is for grade B loans. Grade B and C loans are almost at the same level. The lowest loan amount is for grade G loan. We have calculated the mean of loan amounts for each grade. Looking at the following result, we can say that the loan amount varies by every grade. The highest loan amount is for grade B loans. Grade B and C loans are almost at the same level. The lowest loan amount is for grade G loan. The first snippet corresponds to mean interest rate per grade and the second corresponds to mean interest rate per subgrade. In both cases we see that for every row the interest rates are different. The lowest interest rate is for grade A loans. The highest interest rate is for grade G loans. Small businesses have the highest default rate followed by loans taken for moving. The least default rates are for car and credit card loans.

(c)Are there missing values? What is the proportion of missing values in different variables? Explain how you will handle missing values for different variables. You should consider what the variable is about, and what missing values may arise from – for example, a variable monthsSinceLastDeliquency may have no value for someone who has not yet had a delinquency; what is a sensible value to replace the missing values in this case? Are there some variables you will exclude from your model due to missing values?

We have used following criteria for handling missing values -

- Drop all columns which have 100% missing values
- Drop columns which have more than 70% missing values
- Replace the missing values with either mean, median, or mode depending on the data.

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). Identify and explain which variables you will exclude from the model.

We have identified and removed the following variables from our dataset by adding them in one of three buckets -

1. **Variable not available for new loans** - term, funded_amnt_inv, funded_amnt, out_prncp, out_prncp_inv, total_rec_late_fee, total_pymnt, total_pymnt_inv, total_rec_prncp, total_rec_int, collection_recovery_fee, debt_settlement_flag, hardship_flag, , actualReturn, actualTerm, last_pymnt_d, last_pymnt_amnt, annRet

2. **Variable value updating after the loan is funded** - recoveries, last_credit_pull_d, mths_since_recent_inq, inq_last_6mths

3. **Not Useful for model** - emp_title, pymnt_plan, title, zip_code, addr_state, policy_code, disbursement_method, application_type, issue_d, tot_cur_bal, tot_coll_amt, pub_rec

4. Do a univariate analysis to determine which variables (from amongst those you decide to consider for the next stage prediction task) will be individually useful for predicting the dependent variable (loan_status). For this, you need a measure of relationship between the dependent variable and each of the potential predictor variables. Given loan-status as a binary dependent variable, which measure will you use? From your analyses using this measure, which variables do you think will be useful for predicting loan_status? (Note – if certain variables on their own are highly predictive of the outcome, it is good to ask if this variable has a leakage issue).

We have calculated AUC values to determine which predictor variables best predict the target variable. The reason for choosing AUC curves is that we have a binary classification problem and our class distribution ("Fully Paid" Vs. "Charged Off") is skewed.

We selected variables with AUC values above 0.56 to best help in predicting the target variable loan status. A predictor with AUC value as 0.5 is unable to distinguish between classes. So we have chosen predictor variables with AUC >0.56. Based on our selection criteria the best 3 predictor variables are - int_rate, annual_inc, and dti.

PART B

5.

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

We have converted the character variables as factors with levels. After that we have split the data frame into training and validation sets. We have divided the data frame into 70% training set, and 30% validation set. Our data frame size is very large with 100,000 rows, so it allows us to assign more data to the training set, which will help our model train better.

(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?

(c)Identify the best tree model. Why do you consider it best?

Describe this model – in terms of complexity (size).

Examine variable importance. How does this relate to your univariate analyses in Question

4 above?

Briefly describe how variable importance is obtained (the process used in decision trees).

Variables causing data leakage have already been addressed prior to applying the models on our data frame.

rpart -

We made our decision tree model by loosening the rpart parameters due to the fact that the training set is highly imbalanced with the following proportions - fully paid(0.8629571) and charged off(0.1370429). We also find out that in each of the models the Positive class is always Fully Paid which is due to data imbalance which we have tried to address as follows -

We have tried to handle this class imbalance by creating different models using rpart -

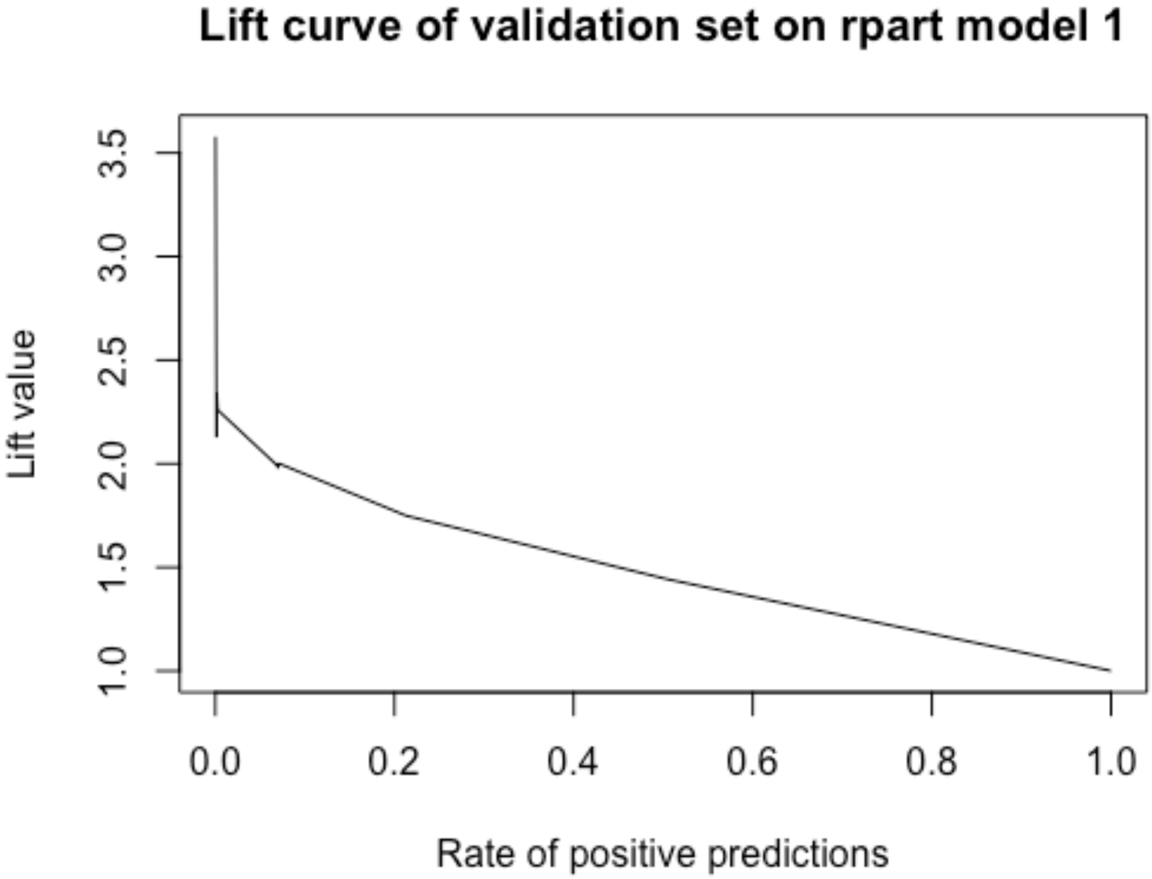
Model 1 - Relaxed cp parameter to 0.0001 and minsplit to 50. Training gave a mean accuracy of 87.22% and testing gave 84.46%. Pruned tree gave an accuracy of 86.38% on training and 85.97% on testing. The testing accuracy increased for pruned tree which gives the better model.

The F1 score on the validation set is 0.92.

TP rate on validation set = 0.99

FP rate on validation set = 0.9961

AUC value for validation set = 0.65



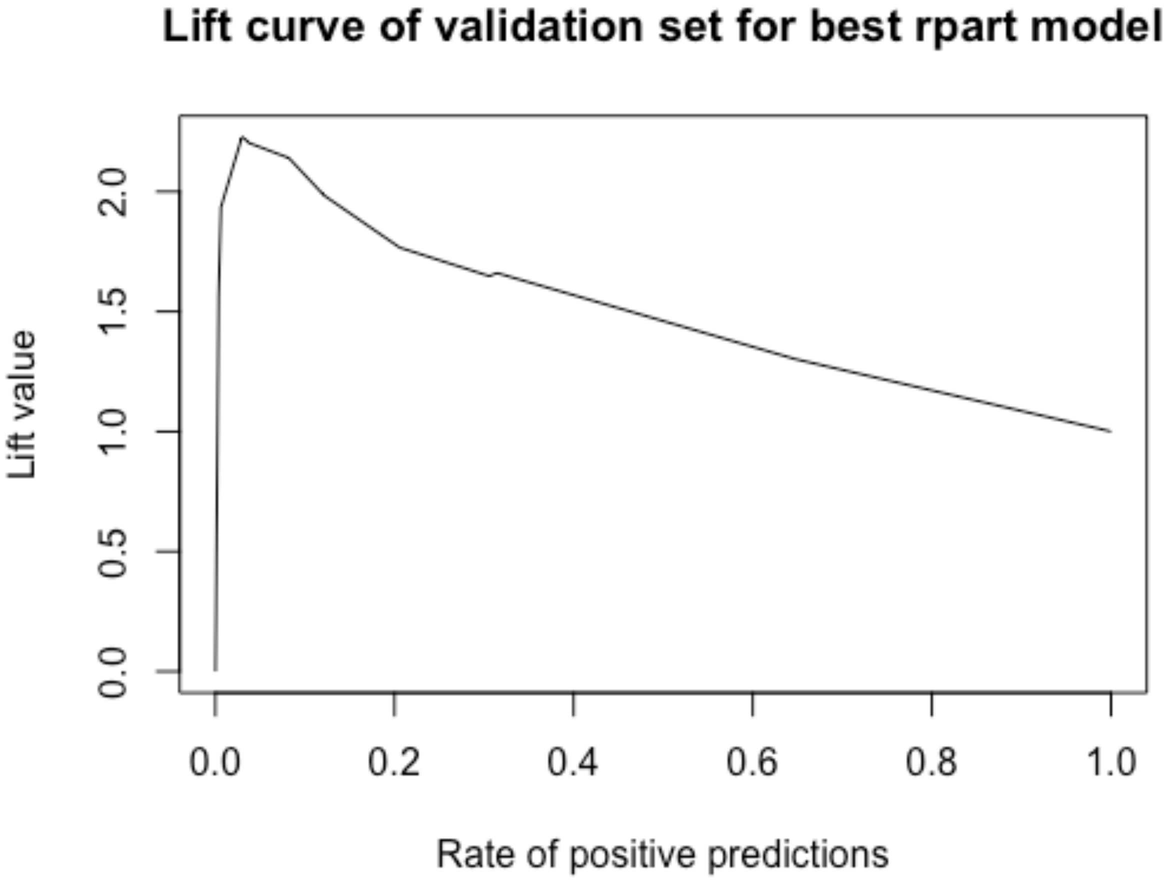
Model 2 - Undersampled the number of fully paid loans in our training set. The new training set would have 2/3 fully paid loans and 1/3 charged off loans. This time the cross validation was decreasing which was expected due to balancing the class reasonably. The pruned tree gave a mean accuracy of 83.16% on the training set, and 82.73% on the validation set.

F1 score on validation set is 0.902.

TP rate on validation set = 0.9332

FP rate on validation set = 0.8244

AUC value for validation set = 0.66



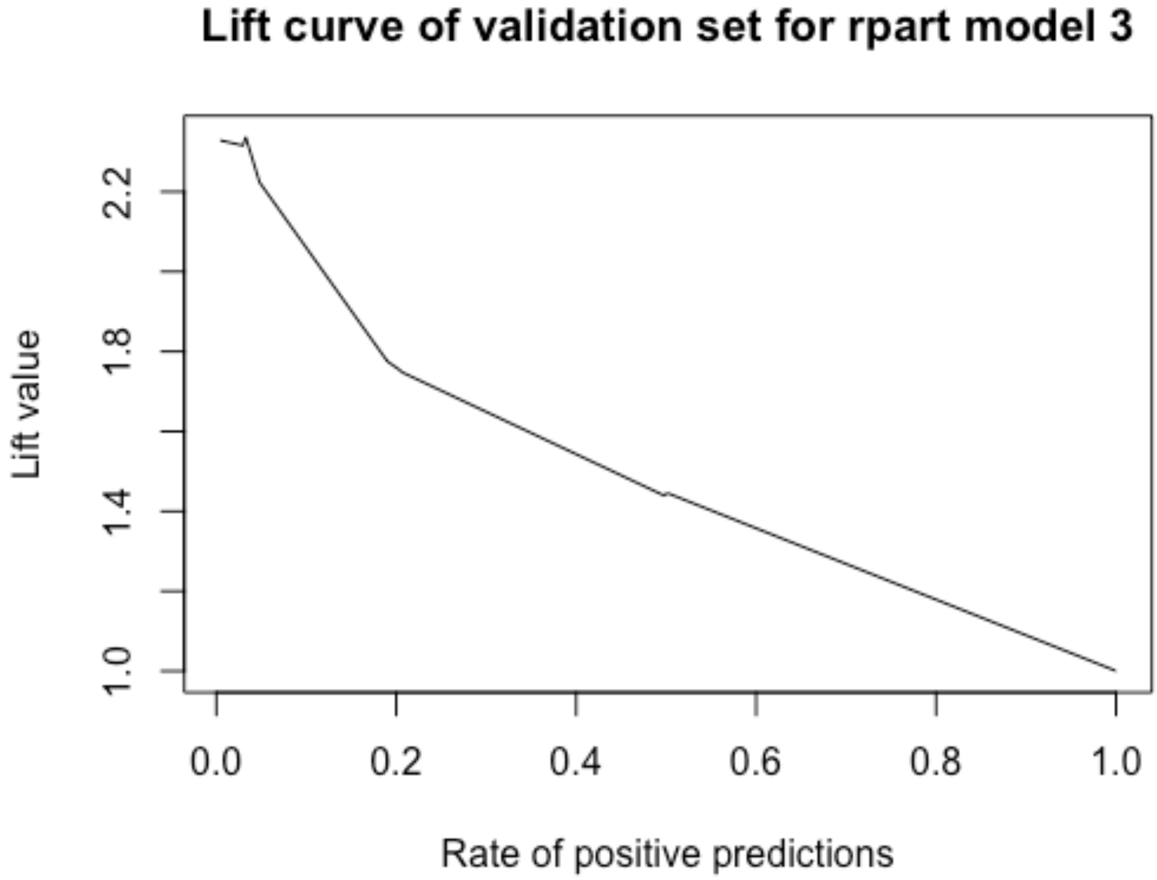
Model 3 - Used the prior parameter to set prior probabilities of fully paid as 0.7 and charged off as 0.3. This was done to adjust the importance of misclassifying each class. The pruned tree gave a mean accuracy of 84.72% on the training set, and 84.20% on the validation set.

F1 score on validation set is 0.912

TP rate on validation set = 0.96

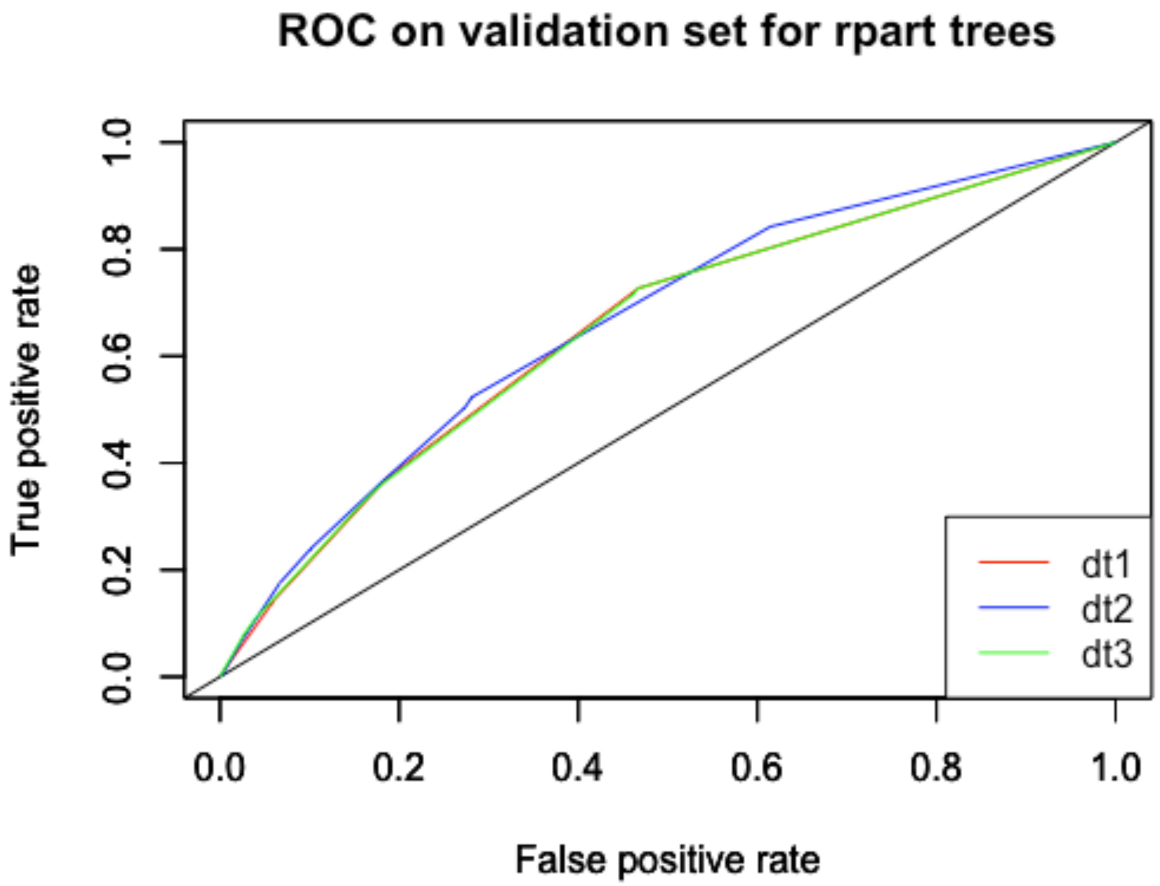
FP rate on validation set = 0.89

AUC value for validation set = 0.65



For each of the models we have found out the best cp value which reduces the cross validation error, and used this cp to prune our tree. This pruned tree is used to find the most optimal tree based on our model.

We have also evaluated performance using confusion matrix, ROC curve, AUC value, and Lift curves on both the training and testing sets. We have also calculated precision, recall, and F1 score to see how well the model is able to predict the positive class. We have found out the TP rate, FP rate of each model, and also plotted the ROC curves of each model on the same plot to compare them simultaneously.

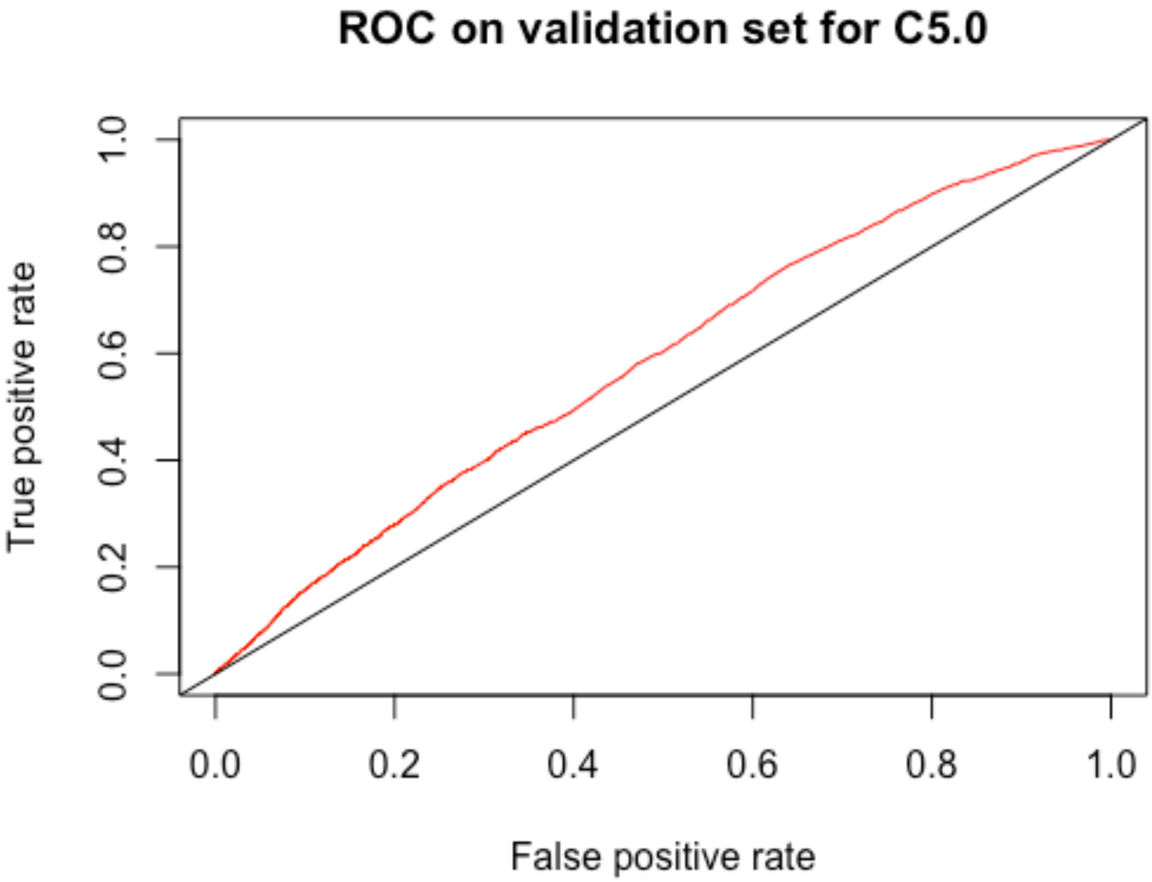


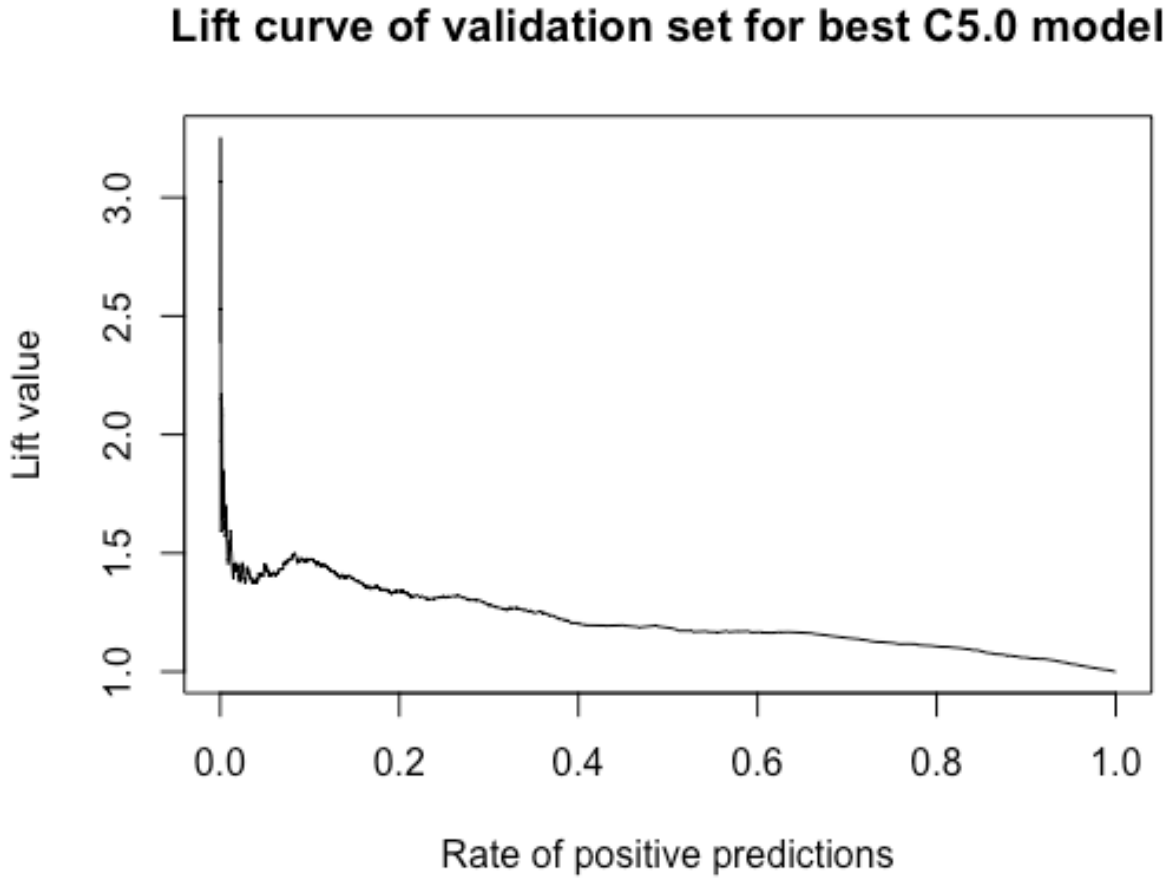
We give importance to Fully Paid loans on the whole since we are interested in finding out the loans that do not default. Based on the AUC curves, F1 measure, FP rate and TP rate, the best rpart model that we achieved was Model - 2. The cross validation error for Model - 2 also decreased and allowed us to select the best cp for pruning. This model also has the least FP rate which is the misclassification of a charged off loan as fully paid loan.

C5.0 -

We used the ROSE library to oversample the Charged Off loans to balance our training dataset.

After that we applied the C5.0 model by experimenting with the parameter mincases. The accuracy went down as the mincases value was increased. We achieved an accuracy of 72.55% with mincases = 10. We have also evaluated performance using ROC, and lift curves.


ROC on validation set for C5.0
True positive rate
0.0
0.2
0.4
0.6
0.8
1.0
0.0
0.2
0.4
0.6
0.8
1.0
False positive rate



(c) Identify the best tree model. Why do you consider it best?

Describe this model – in terms of complexity (size).

Examine variable importance. How does this relate to your univariate analyses in Question

4 above?

Briefly *describe* how variable importance is obtained (the process used in decision trees).

The best tree model is model - 2, in which we have undersampled the training data such that Full Paids loans are 2/3rds of the training set, and Charged Off as 1/3rds. We chose information gain over gini for the splitting of nodes, and chose the cp parameter as 0.001. The cross validation error had the decreasing pattern and we chose the best cp value for pruning our tree with cp = 0.001537579. The Pruned tree gave a mean accuracy of 83.16% on training set, and 82.73% on validation set. F1 score on validation set is 0.912 which shows that model is handling the positive class - "Fully Paid" very nicely.

TP rate on validation set = 0.96.

FP rate on validation set = 0.89.

This shows the cost of misclassifying the "Charged Off" as a "Fully Paid" loan. AUC value for validation set = 0.65, which is reasonably okay in distinguishing between classes given class imbalance. We also compared the AUC curves for all the models on the same plot to decide that Model - 2 is the best among all.

Variable importance in decision tress is calculated by taking the sum of value of split for a variable when it is used for a split, plus where this variable is a surrogate split.

6. Develop a random forest model. (Note the 'ranger' library can give faster computations) 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 performance of random forest and best decision tree model from Q 5 above. Do you find the importance of variables to be different ?

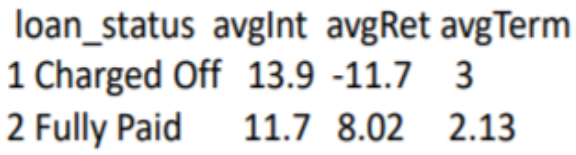
Which model would you prefer, and why ?

We have made random forest models using the ranger library. We have made different models based on different parameters and compared them. Based on our analysis the best random forest model came out to be rf3 with parameters as num.trees=500, mtry=7, min.node.size=30. The accuracy came out to be 86.24% on the validation set. Sensitivity came out to be 0.86, and FP rate came out to be 0.36 which is the lowest that we have got till now.

For our best model we have also found out some of the variables with high importance as follows - int_rate, installment, sub_grade, annual_inc, openAccRatio, borrHistory.

7.(a)

a. Information we have with us:

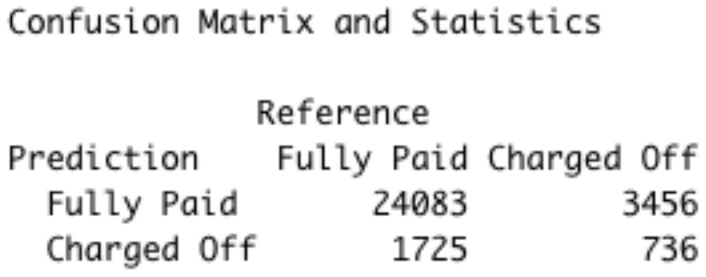


```
  loan_status  avgInt  avgRet avgTerm
1 Charged Off  13.9   -11.7    3
2 Fully Paid   11.7    8.02    2.13
```

Based on our analysis and various models developed, we have observed that the Random Forest model performs best in terms of accuracy.

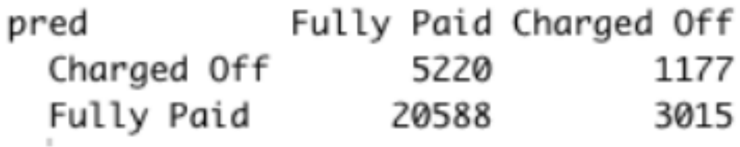
While choosing a model for investments such that we get highest gains and minimal risk/loss, in which case we look for a model with a combination of high True Positive Rate and low False Positive Rate. These are the confusion matrices for all the models:

Decision Tree Model built using RPART



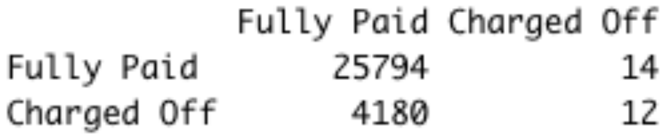
```
Confusion Matrix and Statistics


             Reference
Prediction    Fully Paid Charged Off
  Fully Paid       24083        3456
  Charged Off       1725         736
```

Decision Tree Model using C50



```
pred           Fully Paid Charged Off
  Charged Off        5220        1177
  Fully Paid        20588        3015
```

Random Forest Model



```
            Fully Paid Charged Off
Fully Paid       25794          14
Charged Off       4180          12
```

Analysing the above matrices, we see that the Random Forest Model will give the best returns.

We have done Threshold analysis on the rpart, C5.0, and random forest models as follows -

rpart - We used Threshold value as 0.5 to achieve accuracy of 83.16%.

C5.0 - We used Threshold value as 0.9 to achieve accuracy of 91.62%%

Investing in a loan for 2.13 years and then investing in another loan for the remainder of the 3 years makes sense as 0.87 years alone gives a 2% interest rate which makes it a total interest of 8.02*2.13 + 2*0.87 on a $100 investment

7b) We have calculated cumulative profits in R by arranging in descending order of probability of being paid. This is the approach used by the model to judge loan performance.