Lending Club

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).

Introduction to lending platform:

Peer-to-peer (P2P) lending platforms connect those who are looking for a loan (borrower) to the people who are looking for investment opportunities ([investor]lender). Some popular examples of these P2P platforms are - *Lending Club (LC), Prosper, Peerform, Upstart, and many more*. These platforms set the rate of interest and the terms and conditions (at times with the input from the investor) of the loan. These platforms use machine learning and data science techniques along with financial parameters like credit ratings, credit and delinquency history and human insights to rank the borrower into pre-defined grades (based on parameters like loan amount, risk involved and more) which determines their rate of interest and the terms of loan. Such platforms have now grown from peer to peer and are now upscaled to institutions and hedge funds investing in companies or individuals looking for a loan.

Reason to opt for P2P lending platform:

Some key reasons to attract both borrower(s) and investor(s):

- Lower interest rate in compared to traditional methods (like banks)
- Lower processing fee
- No collateral required (an option missing from the traditional methods).
- Standardized application process
- Faster loan approval process
- Vast range of loan amounts offered.
- Higher return rates for investors.
- Lower market risk for the investors.
- Unlike the traditional methods, P2P platforms do not require any infrastructure or significant workforce. This brings down the operating costs translating to competitive rate of interest (borrower) and higher earnings (investors).

Working of P2P platform:

Process of Lending and borrowing money on a P2P platform:

- Fill an online application form
- Lending platform assigns a risk grade based on your purpose of borrowing the money, credit score and other such details. This grade is closely bound to the interest rate.
- Investors review your loan request: To garner the investors attention, the borrower has to specify the details of the business model and a plan detailing the expenditure of the borrowed money.
- Then, the investor makes a bid for the proposed idea/request and (if) the borrower is contended with the rate of interest and the terms and conditions of the loan, then the borrower can go forth with the loan process.

Drawbacks of P2P platform:

- Strict government restraints and regulations.
- Lower awareness of such platforms amongst the prospective borrowers and investors, stunts the growth of the industry.
- No insurance or government protection in case a borrower defaults. (risk to the investor)
- Not available at every location..

Lending Club:

Lending Club (LC) is an online P2P platform which connects a borrower (party that is looking to borrow money) to an investor (another party which is looking to invest money). The major selling point for such a platform is that the interest rates are lower for the borrower and the return rate is higher for the lender than any other traditional system. LC facilitates providing an unsecured loan to the borrower within the range of 1000$ - 40000$. LC earns money by charging a processing fee to the borrower and charging a transaction fee to the lender on each repayment of the loan amount by the borrower. Such a fee is usually meager compared to the overall cost implied by the traditional systems.

All those seeking to borrow money using the Lending Club platform, first they upload certain details like the 'purpose' for borrowing the money, how 'plan to invest' the money, how they 'plan to repay' the loan etc. LC then runs some background checks on the borrower like their credit rating etc and based on the analysis, they categorize the borrower into grades. The grading system at Lending Club is as follows: A,B,C,D,E,F & G followed by sub-class in each grade category. 'A' being the safest and 'G' being the riskiest (investment). These grades decide various attributes in the loan system. Like, the 'interest rate' charged to a person with a 'safer grade' rating is <u>lower</u> than the person with a 'riskier grade' rating. If the borrower is content with the charged interest rate and the terms & conditions, he/she can move ahead with the process. On the other hand, the investors can browse on the website/platform looking for the right investing opportunities. They can then connect with the borrower of their choice and the process is complete.

Resource:

- investopedia.com
- alliedmarketresearch.com

Q2. Data exploration
(a) some questions to consider:
(i) What is the proportion of defaults ('charged off' vs 'fully paid' loans) in the data? How does the default rate vary with loan grade? Does it vary with sub-grade? And is this what you would expect, and why?

The proportion of the loan_status split is as follows:

loan_status <chr>	n <int>
Charged Off	13785
Fully Paid	86215

2 rows

Variation in default rate by grade:

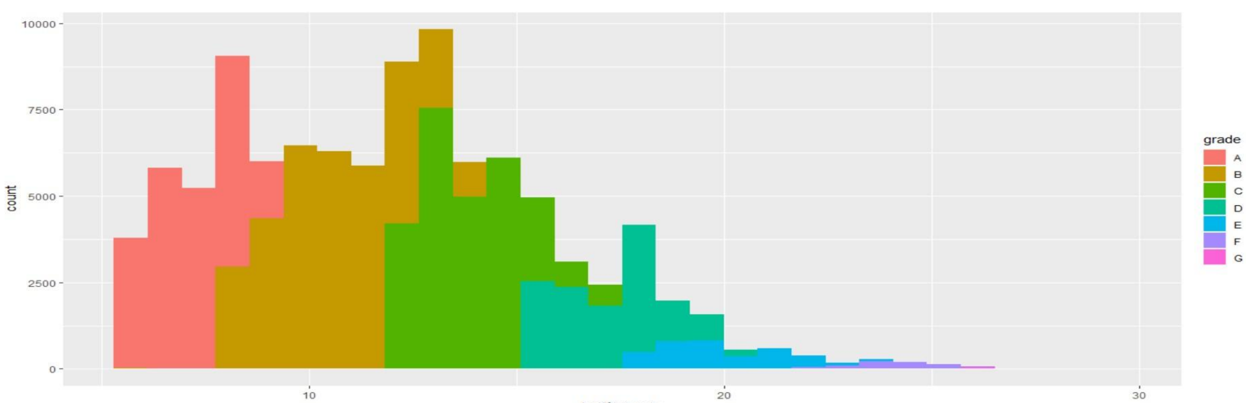
grade <chr>	nLoans <int>	defaults <int>	defaultRate <dbl>
A	22588	1187	0.05255003
B	33907	3723	0.10980034
C	26645	4738	0.17781948
D	12493	2858	0.22876811
E	3579	1010	0.28220173
F	708	239	0.33757062
G	80	30	0.37500000

Variation in default rate by sub-grade:

sub_grade <chr>	nLoans <int>	defaults <int>	defaultRate <dbl>
A1	3774	105	0.02782194
A2	3431	116	0.03380939
A3	3706	179	0.04830005
A4	5138	319	0.06208641
A5	6539	468	0.07157058
B1	6228	491	0.07883751
B2	6880	619	0.08997093
B3	7193	825	0.11469484
B4	7103	855	0.12037167
B5	6503	933	0.14347224
C1	6506	978	0.15032278
C2	5968	970	0.16253351
C3	5446	1009	0.18527360
C4	4657	927	0.19905519
C5	4068	854	0.20993117
D1	3540	764	0.21581921
D2	2806	644	0.22950820
D3	2509	570	0.22718214

sub_grade	nLoans	defaults	defaultRate
<chr>	<int>	<int>	<dbl>
D4	2011	496	0.24664346
D5	1627	384	0.23601721
E1	1118	296	0.26475850
E2	968	267	0.27582645
E3	651	180	0.27649770
E4	466	141	0.30257511
E5	376	126	0.33510638
F1	252	63	0.25000000
F2	141	44	0.31205674
F3	163	59	0.36196319
F4	97	47	0.48453608
F5	55	26	0.47272727
G1	31	12	0.38709677
G2	21	9	0.42857143
G3	19	5	0.26315789
G4	5	2	0.40000000
G5	4	2	0.50000000

As we can see from the graph below, the interest is higher for riskier loans and it is lower for safer loans. We also observe that the count of the number of loans granted to safer loans is higher as compared to riskier loans. This is something we expected to see since the borrowers would not be willing to pay a higher rate of interest if their history indicated that they have always returned the loan amount with interest and also that they have a sound reason for borrowing the money.



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?

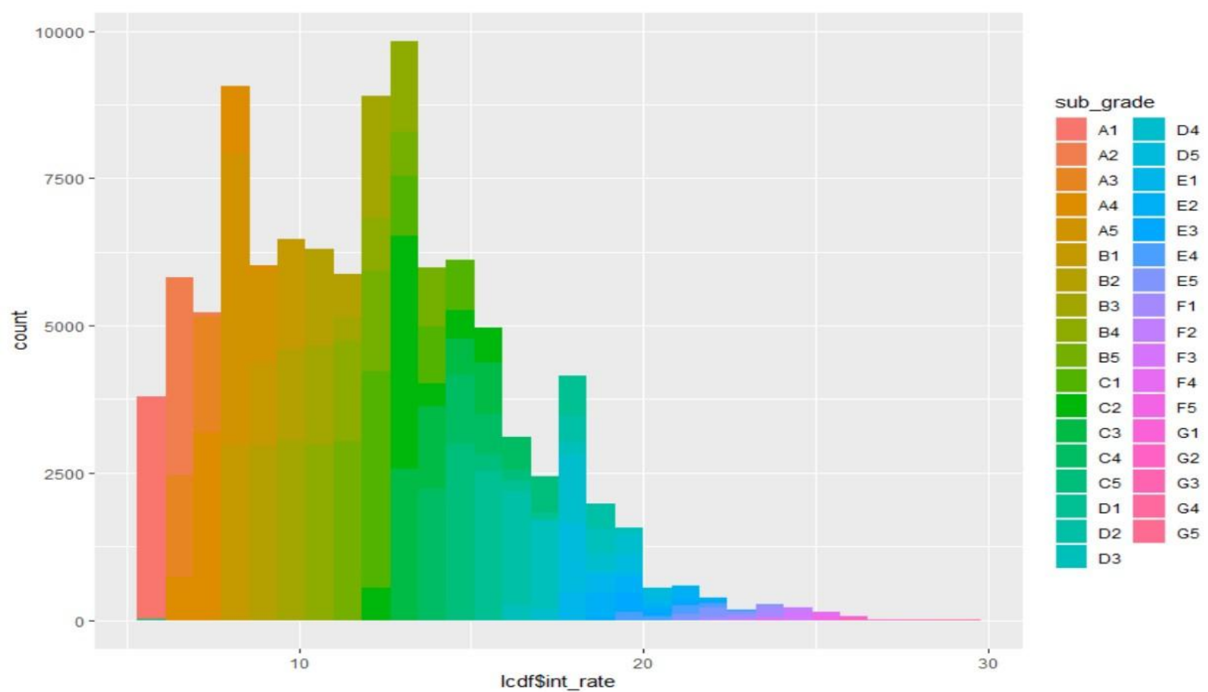
The details of the number of loan amounts and average loan amount for each grade is shown in images below:

grade <chr>	nLoans <int>
A	22588
B	33907
C	26645
D	12493
E	3579
F	708
G	80

grade <chr>	mean(loan_amnt) <dbl>
A	14505.451
B	12637.348
C	12000.828
D	11893.927
E	11618.832
F	9272.493
G	11825.938

As we can see from the table above, we do not see any particular trend in loan amount sanctioned to the borrower.

The interest rate increases from safer loan grade to riskier loan grade as we can see below:



The summary of the mean, standard deviation, min and max of loans by loan grades and sub-grades please refer to the images below:

grade <chr>	sub_grade <chr>	mean(int_rate) <dbl>
A	A1	5.680069
A	A2	6.415494
A	A3	7.094107
A	A4	7.475851
A	A5	8.241788
B	B1	8.870010
B	B2	9.959382
B	B3	10.845931
B	B4	11.731457
B	B5	12.227378
C	C1	12.861531
C	C2	13.308202
C	C3	13.975283
C	C4	14.568033
C	C5	15.221362
D	D1	16.098910
D	D2	16.956411
D	D3	17.445309
D	D4	18.074525
D	D5	18.484259
E	E1	18.972987
E	E2	19.578853

grade <chr>	sub_grade <chr>	mean(int_rate) <dbl>
E	E3	20.143318
E	E4	20.993391
E	E5	21.970027
F	F1	23.124762
F	F2	23.742624
F	F3	24.385337
F	F4	24.952990
F	F5	25.595455
G	G1	26.120000
G	G2	26.393810
G	G3	26.733684
G	G4	26.990000
G	G5	26.792500

grade	sub_grade	max(int_rate)
<chr>	<chr>	<dbl>
A	A1	6.03
A	A2	6.97
A	A3	7.62
A	A4	8.60
A	A5	9.25
B	B1	10.16
B	B2	11.14
B	B3	12.12
B	B4	13.11
B	B5	14.09
C	C1	14.33
C	C2	15.31
C	C3	15.80
C	C4	16.29
C	C5	17.27
D	D1	17.77
D	D2	18.55
D	D3	19.20
D	D4	19.52
D	D5	20.31
E	E1	21.00
E	E2	21.70
E	E3	22.40
E	E4	23.10
E	E5	23.40
F	F1	23.70
F	F2	24.08
F	F3	24.50
F	F4	25.09
F	F5	26.06
G	G1	26.99
G	G2	27.31
G	G3	27.99
G	G4	28.49
G	G5	28.99

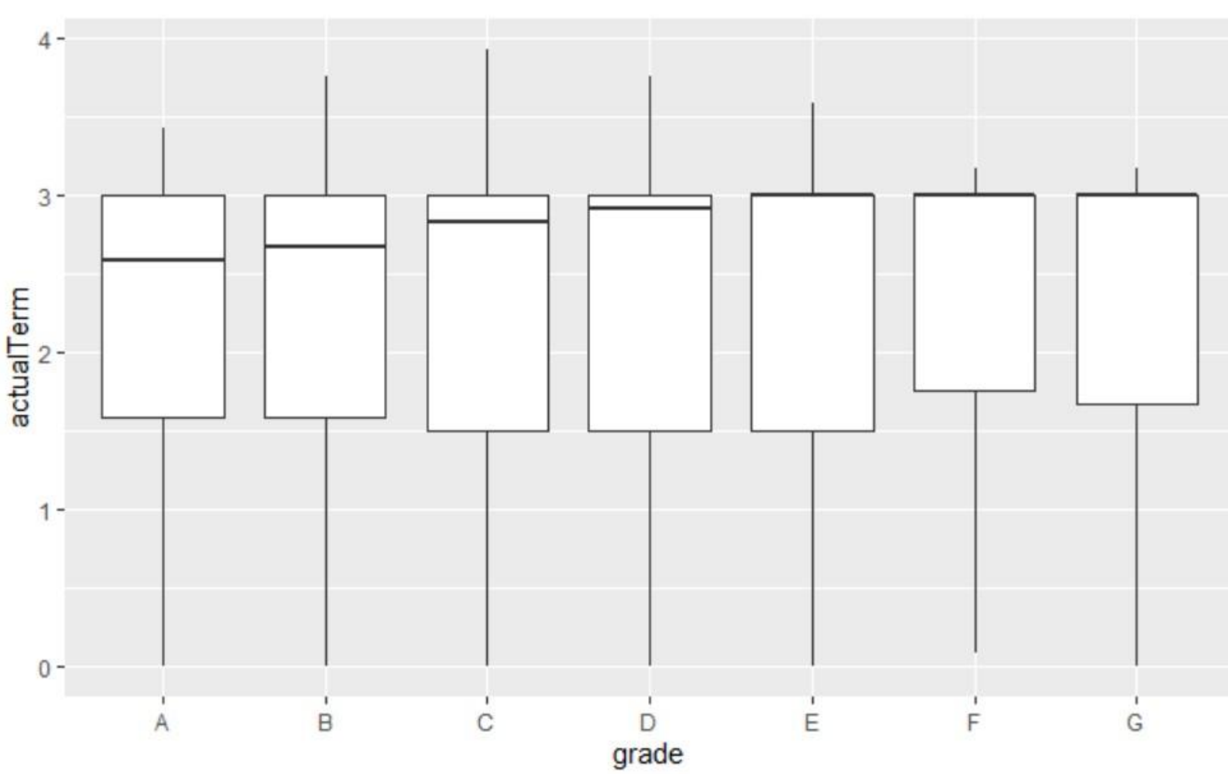
grade	sub_grade	sd(int_rate)
<chr>	<chr>	<dbl>
A	A1	0.3474851
A	A2	0.1662589
A	A3	0.3247008
A	A4	0.3573953
A	A5	0.4244667
B	B1	0.7217524
B	B2	0.8155856
B	B3	0.8873289
B	B4	0.8397941
B	B5	0.8512147
C	C1	0.7861758
C	C2	0.8732851
C	C3	0.8656083
C	C4	0.8547142
C	C5	0.8834418
D	D1	0.8706865
D	D2	0.8866280
D	D3	0.8734737
D	D4	0.8318050
D	D5	1.0020948
E	E1	0.9872700
E	E2	1.0589062
E	E3	1.0321440
E	E4	0.9523777
E	E5	0.7628328
F	F1	0.5962301
F	F2	0.4761702
F	F3	0.2471374
F	F4	0.2144721
F	F5	0.2729049
G	G1	0.4729271
G	G2	0.7364678
G	G3	1.0167061
G	G4	1.3693064
G	G5	1.4650000

grade <chr>	sub_grade <chr>	min(int_rate) <dbl>
A	A1	5.32
A	A2	6.24
A	A3	6.68
A	A4	6.92
A	A5	6.00
B	B1	6.00
B	B2	6.00
B	B3	6.00
B	B4	6.00
B	B5	6.00
C	C1	11.99
C	C2	6.00
C	C3	6.00
C	C4	6.00
C	C5	6.00
D	D1	6.00
D	D2	6.00
D	D3	6.00
D	D4	17.14
D	D5	6.00
E	E1	6.00
E	E2	18.49
E	E3	18.99
E	E4	19.99
E	E5	20.99
F	F1	21.99
F	F2	22.99
F	F3	23.63
F	F4	23.76
F	F5	23.83
G	G1	25.80
G	G2	25.83
G	G3	25.89
G	G4	25.99
G	G5	26.06

The number of investors investing decreases as the loan gets riskier as no investor would want to risk losing their money however the average amount loaned to the borrower remains almost the same throughout all loan grades. The other images indicate that the mean, max, min and standard deviation of interest rate increases as the loan grade protrudes towards the riskier side as it should have been.

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).

The term for all loans given in the dataset is 3 years. As we can see from the boxplot below, on an average borrowers belonging to safer loan grade tend to repay the loan amount before the end of the 3 year term and as the loan grade gets riskier the, borrowers time taken to repay the loan treads closer to the entire duration of 3 years. This is as expected since the loans belonging to the riskier grade category have higher interest rates and thus would take more time to repay the money.
It can be rightly drawn from observing the box plot that as the loan grade increases to higher risk the actual term is pushed more towards finalized term period (3 years). Majority of those (higher risk loans) also contribute to no return towards the end of the term.



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?

The annual return table and the table displaying returns from charged off loans is visible in the rmd file shared. The line of code used to calculate the annual return is as follows:

```
# Annual return in $
lcdf$annualRet <- ((lcdf$total_pymnt-lcdf$funded_amnt))*(12/36)
lcdf$annualRet
```

The line of code used to calculate the % annual return is as follows:

```
# % annual return
lcdf$annRet <- ((lcdf$total_pymnt -lcdf$funded_amnt)/lcdf$funded_amnt)*(12/36)*100
lcdf$annRet
```

The line of code used to calculate the returns from the charged off loans is as follows:

```
# Payments from charged off loans
lcdf %>% filter(loan_status=='Charged Off') %>% summarise(funded_amnt,total_pymnt)
```

The image below shows the average payment from the loans by grades and other such details:

grade <chr>	nLoans <int>	defaults <int>	defaultRate <dbl>	avgInterest <dbl>	stdInterest <dbl>	avgLoanAMt <dbl>	avgPmnt <dbl>
A	22588	1187	0.05255003	7.173848	0.9669664	14505.451	15579.42
B	33907	3723	0.10980034	10.753559	1.4431575	12637.348	13778.88
C	26645	4738	0.17781948	13.847765	1.1859154	12000.828	13011.01
D	12493	2858	0.22876811	17.190576	1.2220189	11893.927	12870.82
E	3579	1010	0.28220173	19.927656	1.3755560	11618.832	12374.37
F	708	239	0.33757062	23.980438	0.9163869	9272.493	10050.14
G	80	30	0.37500000	26.425625	0.8490767	11825.938	12645.26

We can notice that as the average interest rate increases, the difference between average loan amount and the average payment decreases.
The annual return decreases with increase in risk based on grades and sub-grade but there is no strong pattern observed. But the relation among the attributes annual return and grade + subgrade is inversely proportional.

Based upon results obtained so far, it would be advisable to invest in loan category 'A' since it is low risk and the default rate is extremely low as compared to other loan grades.

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?

The purpose of borrowing money is mentioned below with the number of loans for each categorical purpose:

purpose <chr>	n <int>
car	928
credit_card	24989
debt_consolidation	57622
home_improvement	5654
house	354
major_purchase	1823
medical	1119
moving	691
other	5091
renewable_energy	58
small_business	893
vacation	678
wedding	100

The average funded amount based on purpose and the number of defaults are mentioned below:

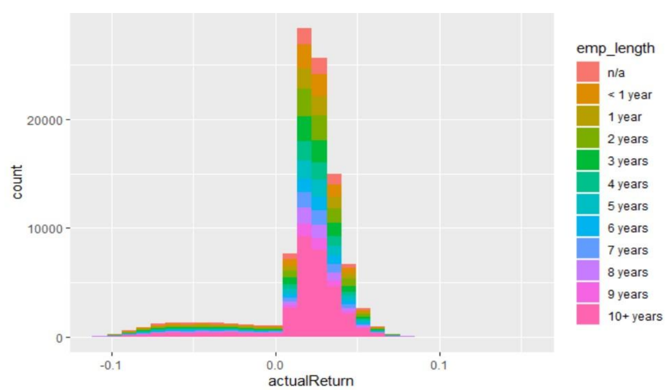
purpose <chr>	avg_funded_amt <dbl>	no_of_defualts <int>
car	7955.038	107
credit_card	13660.144	2865
debt_consolidation	13227.955	8319
home_improvement	11911.059	682
house	12756.568	63
major_purchase	9948.286	266
medical	7313.248	172
moving	6882.308	144
other	8304.920	838
renewable_energy	8806.897	11
small_business	13603.415	203
vacation	5674.410	101
wedding	9123.750	14

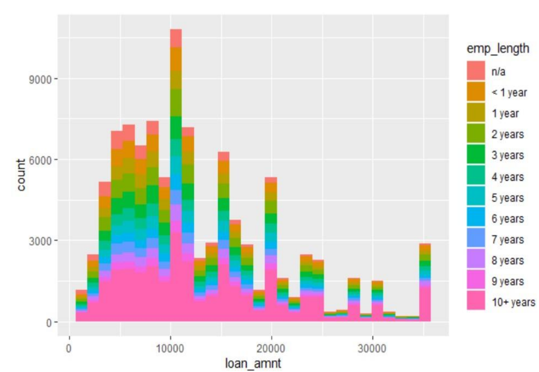
There is no clear demarcation to indicate difference in loan amount by purpose. However, we can see that there are a higher number of defaults for purposes like debt_consolidation and credit_card payment.

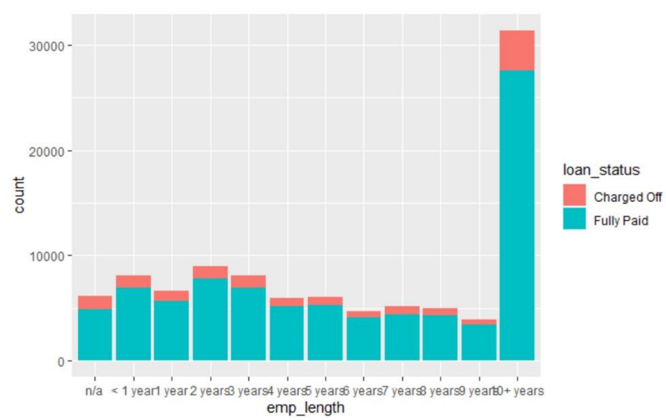
The loan grade assigned by Lending Club does vary by purpose as seen from the table shared in the rmd file. For example, loans taken for purchasing a car were typically classified as a category ‘A’ or a category ‘B’ loan, on the other hand loans given for vacation purposes were typically classified as a ‘D’ category loan.

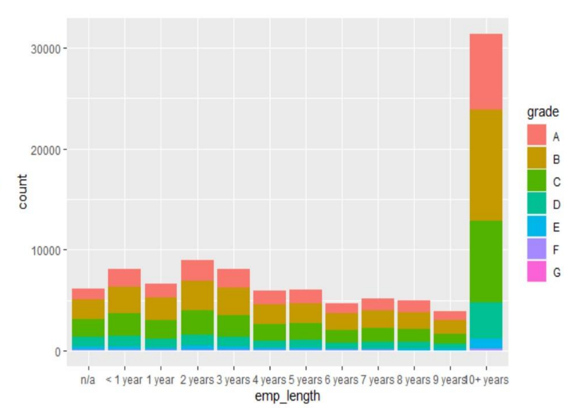
vi) Consider some borrower characteristics like employment-length, annual-income, fico-scores (low, high). How do these relate to loan attribute like, for example, loan_amout, loan_status, grade, purpose, actual return, etc.

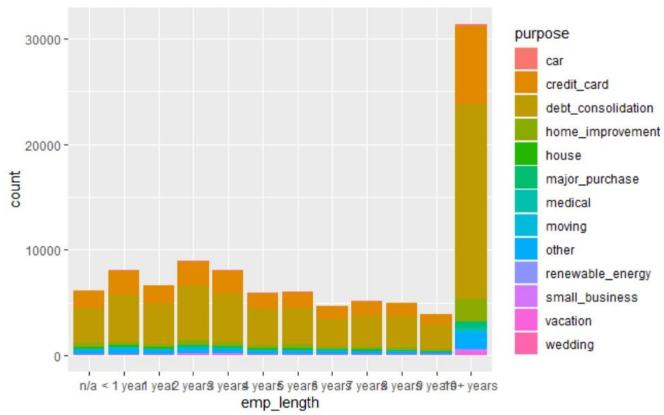
Shown below are some of the many relationships from the rmd file between the borrower characteristics and the loan attributes:











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 analyses as in the questions above (as reasonable based on the derived variables).

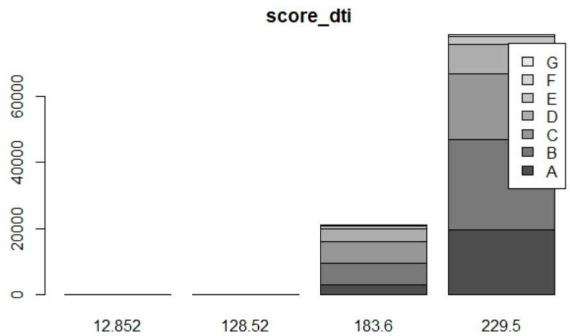
The list of derived variables is as follows with the line number from the rmd file:
Derived variables
Actual Annual Return (validate above for actual term)
Amounts owed by the borrower: lcdf$score_dti (line no. - 274)
Credit history of the borrower: lcdf$score_credhist (line no. - 279)
Credit mix of the borrower: lcdf$score_cred (line no. - 282)
New credit of the borrower: lcdf$score_newcred (line no. - 285)
Delinquency term of the borrower: lcdf$score_delinq (line no. - 288)
Delinquency amount of the borrower: lcdf$score_delinq_amn (line no. - 291)
Fico score: lcdf$score (line no. - 294)

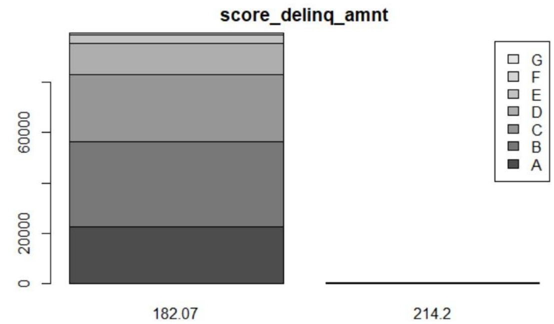
How does amounts owed by a borrower relate to various loan attributes?

- vary by loan status - As the debt to income ratio increases the number of fully paid loans increases, which implies a strong positive relation among the attributes.
- vary by grade - As the debt to income ratio increases the number of approved loans for higher grades decreases, which implies a strong negative relation among the attributes.

How does the borrower relate to various loan attributes?

- vary by loan status - As the delinquency amount increases the number of loans fully paid becomes 0 the charged off loans are slightly above 0. And, for lower delinquency amounts the count for fully paid loans increases, which shows a weak relationship among the variables.
- vary by grade - There is no clear pattern observed and the relation among the variables cannot be determined.





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

Ans. The main themes observed across all variables are that the employment length, grade, sub grade, actual return and average income have strong contributions towards the loan status, borrower profile and many more derived variables. Some of the derived variables like fico score, and attributes around fico score help evaluate the borrower profile and identify potential risk at an early stage. For a detailed view refer to the values and conclusions discussed above.

(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?

Yes, there are quite a lot of missing values in the attributes in the dataset provided in the proportions as mentioned in the rmd file.

We have dropped the variables with over 60% missing values as manipulating and replacing the missing values would not give us a good model later.

To handle the missing values we would use mean, median or mode for the variable depending on what the variable represents. If it were a categorical variable, then we would bundle up similar categories and replace the missing value with the value occurring most often in the bundle.

Variables like monthsSinceLastDeliquency where the empty fields mean that the person has not defaulted and thus there is no value entered in the field, we simply replace it by zero to make sense out of it.

The variables that we should drop from the dataframe are mentioned in the below answer. These variables are either not necessary for the analysis or they do not make sense.

Q3. 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 will you exclude from the model.

Ans. List of attributes that can potentially cause data leakage:

Funded_amount_inv, loan_status, revol_bal, out_prncp, out_prncp_inv, mths_since_last_delinq, revol_util, total_pymnt, total_pymnt_inv, total_rec_prncp, total_rec_prncp, total_rec_int, total_rec_int, total_rec_late_fee, recoveries, collection_recovery_fee, last_pymnt_amt, last_credit_pull_d, collections_12_mnths_ex_med, tot_coll_amnt,total_cur_bal, chargeoff_within_12_mths, delinq_amt

The attributes mentioned above should be excluded since it could potentially affect the accuracy of the model since these attributes were included after the loan was sanctioned. Adding these attributes would definitely increase the accuracy of but it would not be a realistic representation of the data that we have while deciding whether to sanction the loan or not.

List of attributes that we would exclude from the dataset while training the model are as follows:

Issue_d, emp_title, addr_state, zip_code, inq_last_6mths, inq_last_12mths, open_acc, last_pymt_d, policy_code, mths_since_recent_inq

The attributes mentioned above are being excluded from the dataset because these are redundant data which would not have any value addition to the model.

Q4. 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).

After performing the univariate analysis, we get the values as follows:

loan_amnt	funded_amnt	term
0.5021399	0.5211402	0.5211402
installment	grade	verification_status
0.6581483	0.5071865	0.5767804
delinq_2yrs	initial_list_status	acc_open_past_24mths
0.5682696	0.5184907	0.5655743
avg_cur_bal	bc_open_to_buy	bc_util
0.5825897	0.5691553	0.5743476
mo_sin_old_il_acct	mo_sin_old_rev_tl_op	mo_sin_rcnt_rev_tl_op
0.5435189	0.5303673	0.5511155
mo_sin_rcnt_tl	mort_acc	mths_since_recent_bc
0.5538335	0.5596704	0.5583196
num_accts_ever_120_pd	num_il_tl	num_op_rev_tl
0.5551020	0.5152625	0.5099021
num_rev_accts	num_rev_tl_bal_gt_0	num_tl_120dpd_2m
0.5176556	0.5078333	0.5077449
percent_bc_gt_75	total_bal_ex_mort	total_bc_limit
0.5123979	0.5735512	0.5169192
total_il_high_credit_limit	hardship_flag	
0.5730079	0.5116315	

We have filtered out the variables which have the auc value higher than 0.5. These are the variables that we will be using for developing the predictive model.

Q5. (a) Split the data into training and validation sets. What proportions do you consider, why?

Proportion that we have considered for our training and testing data is 0.65 and 0.35 respectively as anything less than 65% of the training data would not give us true representation on the entire data. This proportion provides enough training samples even for multiclass classification.

(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?

Rpart-

We tinkered with the minsplit parameter and set it to 30 once and 50 the other time. We observed that the model performed better when the minsplit value was set to 50. We also changed the classification threshold value and found 0.5 to be working the best amongst 0.5, 0.3 and 0.25

- The matrix for the classification threshold value of 0.5 is:

	true	
predTrnCT	Fully Paid	Charged Off
Charged Off	408	708
Fully Paid	55722	8162

- The matrix for the classification threshold value of 0.25 is:

	true	
predTrnCT	Fully Paid	Charged Off
Charged Off	2298	1515
Fully Paid	53832	7355

- The matrix for the classification threshold value of 0.3 is:

	true	
predTrnCT	Fully Paid	Charged Off
Charged Off	1013	1035
Fully Paid	55117	7835

If we compare the above three results, we find the threshold value of 0.5 to work the best.

Variables actually used in tree construction using rpart:

annual_inc	avg_cur_bal	dti	emp_length	grade	home_ownership
installment	int_rate	loan_amnt	pub_rec	purpose	sub_grade

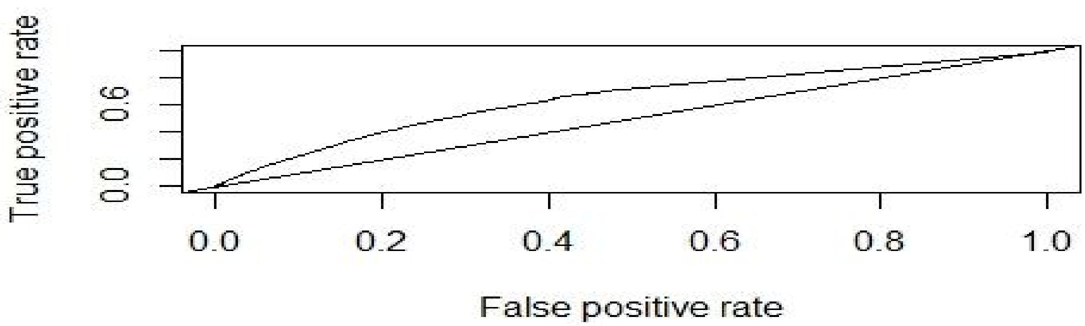
The variable importance table is given in the rmd file.

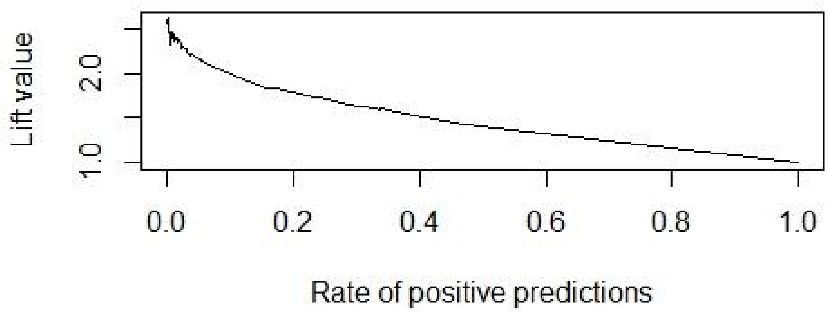
Train statistics:
Following is the confusion matrix for the tree created above:

	true	
pred	Charged Off	Fully Paid
Charged Off	997	1107
Fully Paid	8595	59300

The accuracy of the decision tree is 86.82%

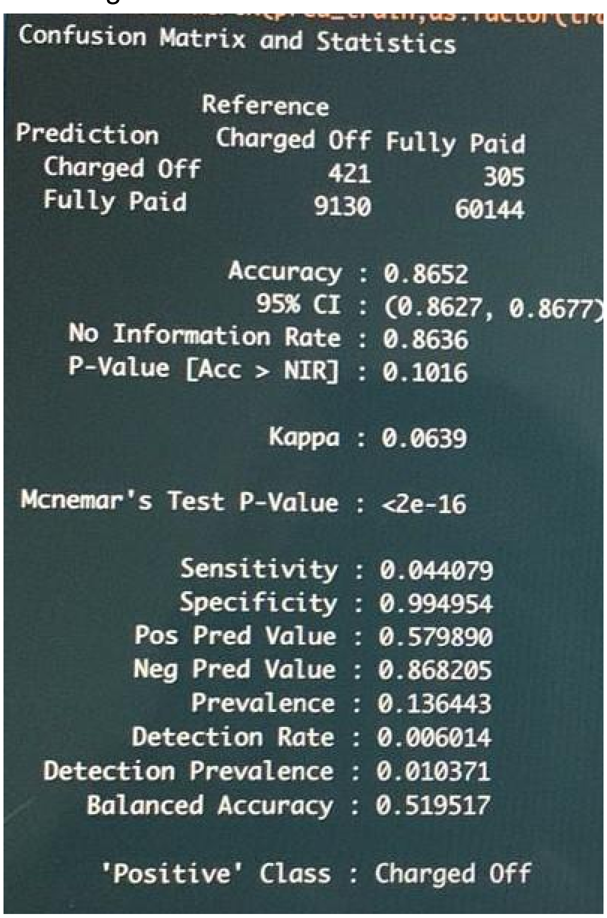
For evaluating the performance, we use the ROC curve and the lift curve for the **rpart** model as shown below:





C5.0-

The image below shows the confusion matrix for **c5.0** model:



```
Confusion Matrix and Statistics

              Reference
Prediction     Charged Off Fully Paid
  Charged Off         421        305
  Fully Paid         9130      60144

               Accuracy : 0.8652
                 95% CI : (0.8627, 0.8677)
    No Information Rate : 0.8636
    P-Value [Acc > NIR] : 0.1016

                  Kappa : 0.0639

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

            Sensitivity : 0.044079
            Specificity : 0.994954
         Pos Pred Value : 0.579890
         Neg Pred Value : 0.868205
             Prevalence : 0.136443
         Detection Rate : 0.006014
   Detection Prevalence : 0.010371
      Balanced Accuracy : 0.519517

       'Positive' Class : Charged Off
```

For the c5.0 model, we tried substituting different values however we found no significant difference in performance.

As we can see from the confusion matrix above, rpart is working better in our case as the number of correctly predicted values are also more and the number of incorrectly predicted values are less than the c5.0 model especially for the charged off values and the values for fully paid.

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 ? For evaluation of models, you should include confusion matrix related measures, as well as ROC analyses and lifts. Explain which performance measures you focus on, and why?

Using bootstrap to create 3 datasets (to train, validate and test) and creating random forest with various configurations - below are the results.

Random Forest Model 1 -

```
> rfModel1
Ranger result

Call:
 ranger(loan_status ~ ., data = df_train, mtry = 6, importance = "impurity",      probability = FALSE)

Type:                             Classification
Number of trees:                  500
Sample size:                      99998
Number of independent variables:  15
Mtry:                             6
Target node size:                 1
Variable importance mode:         impurity
Splitrule:                        gini
OOB prediction error:             5.11 %
```

And it's confusion matrix

```
> cnfm1
Confusion Matrix and Statistics

             Reference
Prediction    Charged Off Fully Paid
  Charged Off        8896       4985
  Fully Paid           65      86052

               Accuracy : 0.9495
                 95% CI : (0.9481, 0.9508)
    No Information Rate : 0.9104
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.7519

 Mcnemar's Test P-Value : < 2.2e-16

            Sensitivity : 0.99275
            Specificity : 0.94524
         Pos Pred Value : 0.64088
         Neg Pred Value : 0.99925
             Prevalence : 0.08961
         Detection Rate : 0.08896
   Detection Prevalence : 0.13881
      Balanced Accuracy : 0.96899

       'Positive' Class : Charged Off
```

Random Forest Model 2 -

```
> rfModel2
Ranger result

Call:
 ranger(loan_status ~ loan_amnt + grade + sub_grade, data = df_train,      num.trees = 1000, importance = "im
purity", probability = FALSE)

Type:                             Classification
Number of trees:                  1000
Sample size:                      99998
Number of independent variables:  3
Mtry:                             1
Target node size:                 1
Variable importance mode:         impurity
Splitrule:                        gini
OOB prediction error:             13.79 %
>
```

And it’s confusion matrix

```
> cnfm2
Confusion Matrix and Statistics

             Reference
Prediction    Charged Off Fully Paid
  Charged Off          14      13867
  Fully Paid            1      86116

               Accuracy : 0.8613
                 95% CI : (0.8592, 0.8635)
    No Information Rate : 0.9998
    P-Value [Acc > NIR] : 1

                  Kappa : 0.0017

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

            Sensitivity : 0.933333
            Specificity : 0.861306
         Pos Pred Value : 0.001009
         Neg Pred Value : 0.999988
             Prevalence : 0.000150
         Detection Rate : 0.000140
   Detection Prevalence : 0.138813
      Balanced Accuracy : 0.897320

       'Positive' Class : Charged Off
```

Random Forest Model 3 -

```
> rfModel3
Ranger result

Call:
 ranger(loan_status ~ purpose + annual_inc + emp_length, data = df_train,      num.trees = 500, importance = 
 "impurity", probability = FALSE)

Type:                             Classification
Number of trees:                  500
Sample size:                      99998
Number of independent variables:  3
Mtry:                             1
Target node size:                 1
Variable importance mode:         impurity
Splitrule:                        gini
OOB prediction error:             13.78 %
> 
```

and it's confusion matrix

```
> cnfm3
Confusion Matrix and Statistics

             Reference
Prediction    Charged Off Fully Paid
  Charged Off          12      13869
  Fully Paid            0      86117

               Accuracy : 0.8613
                 95% CI : (0.8591, 0.8634)
    No Information Rate : 0.9999
    P-Value [Acc > NIR] : 1

                  Kappa : 0.0015

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

            Sensitivity : 1.0000000
            Specificity : 0.8612906
         Pos Pred Value : 0.0008645
         Neg Pred Value : 1.0000000
             Prevalence : 0.0001200
         Detection Rate : 0.0001200
   Detection Prevalence : 0.1388128
      Balanced Accuracy : 0.9306453

       'Positive' Class : Charged Off
```

Random Forest Model 4 -

```
> rfModel4
Ranger result

Call:
 ranger(loan_status ~ purpose + int_rate, data = df_train, num.trees = 1000,      importance = "impurity", pr
obability = FALSE)

Type:                             Classification
Number of trees:                  1000
Sample size:                      99998
Number of independent variables:  2
Mtry:                             1
Target node size:                 1
Variable importance mode:         impurity
Splitrule:                        gini
OOB prediction error:             13.78 %
```

and it's confusion matrix

```
> cnfm4
Confusion Matrix and Statistics

             Reference
Prediction    Charged Off Fully Paid
  Charged Off          27      13854
  Fully Paid           16      86101

               Accuracy : 0.8613
                 95% CI : (0.8591, 0.8634)
    No Information Rate : 0.9996
    P-Value [Acc > NIR] : 1

                  Kappa : 0.003

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

            Sensitivity : 0.627907
            Specificity : 0.861398
         Pos Pred Value : 0.001945
         Neg Pred Value : 0.999814
             Prevalence : 0.000430
         Detection Rate : 0.000270
   Detection Prevalence : 0.138813
      Balanced Accuracy : 0.744652

       'Positive' Class : Charged Off
```

Variable importance for various models - Below, are the weighted variable importance for various models displayed above.

Random forest model 1 -

```
> modelImp1
               X        loan_amnt         int_rate      installment            grade        sub_grade
     0.973166582      0.543174663      0.518763837      0.744952656      0.107693274      0.352115360
      emp_length   home_ownership       annual_inc          purpose              dti          pub_rec
     0.428632510      0.132251818      0.799578274      0.261834707      0.996612225      0.124312507
application_type      avg_cur_bal       tax_liens
     0.001172584      1.000000000      0.046298521
```

Random forest model 2 -

```
> modelImp2
loan_amnt     grade sub_grade
0.4289019 0.7800203 1.0000000
```

Random forest model 3 -

```
> modelImp3
  purpose annual_inc emp_length
0.2829229  1.0000000  0.3405299
> 
```

Random forest model 4 -

```
> modelImp4
  purpose  int_rate
0.1353751 1.0000000
> 
```

Out of the above 4 models we selected Random Forest Model 3 since it has an accuracy of 0.8613 though it has an OOB error of 13.7%. The other models had a lower OOB error value but are overfitted.

Based on various parameters the value of the importance of variables varies. It depends how important that variable is to a particular model based on the model's parameters.

Confusion matrix is one of the most important parameters to compare the models. We can also use ROC and lift curves. The higher the AUC in the ROC curve, the better is the model. Based on these parameters, we can choose which model is the best.

Q7. The purpose of the model is to help make investment decisions on loans. How will you evaluate the models on this business objective? Consider a simplified scenario - for example, that you have $100 to invest in each loan, based on the model's prediction. So, you will invest in all loans that are predicted to be 'Fully Paid'. Key questions here are: how much, on average, can you expect to earn after 3 years from a loan that is paid off, and what is your potential loss from a loan that has to be charged off ?

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

The avg interest for Fully Paid and Charged Off loans comes around:

```
> df %>% group_by(loan_status) %>% summarise(mean(int_rate))
# A tibble: 2 x 2
  loan_status `mean(int_rate)`
  <fct>                  <dbl>
1 Fully Paid              11.7
2 Charged Off             13.9
> 
```

We can see that for every 100$ that is invested in the Fully Paid Loan for 3 years we are getting around 35$ as investment back to us but for Charged Off Loans we are losing around 41.7$. We can still recover some of the amount for Charged Off Loans but the lender is still losing around 50% of the money he has invested for these loans.

The value for profit which we will be using is 35$ and for the value of loss we will be assuming 41.7$. If we compare the random forest and the decision tree model we find that the decision tree is the better of the two. The accuracy of the random forest and the one for the decision tree is around 86% for both. However, when we look at the confusion matrix for both the models, we see that the random forest model is skewed towards fully paid, that is 85% of the correctly predicted values are fully paid and almost the remaining 15% of the values which were predicted to be charged off were actually fully paid. However, with the decision tree model, there is a good balance in the confusion matrix and hence we decided to opt for the decision tree model.

(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) above.

The threshold value of the decision tree is 0.5 and for random forest is 0.6 and if we compare among the two we find that the decision tree is better of the two in predicting the profit.
The Actual Profit and the expected profit for Fully Paid loans comes around:

```
  loan_status `sum(ActualProfit)` `sum(df$ExpectedProfit)` `sum(funded_amnt)`
  <chr>                     <dbl>                    <dbl>              <int>
. Fully Paid           166607244.               378729996.     .   1104577250
```

We find that the decision tree gives us a better result in predicting the Actual Profit as opposed to the random tree which gives a less accurate prediction of the Profit.

We will sort the data in the top 10 decile in the descending order. We will then access the 1st decile and break down the data in 20 parts in the descending order again. Based upon the results that we find in this group, we can choose which loan we should sponsor so that we have high certainty of getting back the money with the interest.

Citation:

https://cran.r-project.org/web/packages/ranger/ranger.pdf
https://www.cnbc.com/2019/02/21/personal-loans-surge-to-a-record-138-billion-in-us-as-fintechs-lead-new-lendingcharge.html 2
https://www.alliedmarketresearch.com/peer-to-peer-lending-market 3
https://www.lendingclub.com/info/statistics.action 4
https://www.lendingclub.com/public/how-peer-lending-works.action
https://www.prnewswire.com/news-releases/lendingclub-receives-regulatory-approvals-to-acquireradius-bancorp-301210498.html