IDS 572

Assignment 1

Lending Club

Harika Lakhinena

Roshan Dhanasiri

Sathwik Maddi

Part A

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.

Business Model:

Matchmaking Business Model

The business model used by Lending Club is a matchmaking business model; where you identify two or more customer groups and bring them together in your marketplace. This platform connects borrowers looking for unsecured loans with the investors (Lenders) who want to get high returns on their investments.

Peer-to-peer Lending is a system where investors can provide money for the borrowers at a certain interest rate, where both the parties meet their needs. (on agreed conditions)

The entire loan process may be completed online. The organization gathers and analyzes data from the borrower, such as his credit score and annual income. This information is delivered to all investors who are interested in making a financial investment. Following an examination of the borrower's information, the investor determines whether to invest totally, partially, or not at all. The borrower receives the final response, after which he may decide whether or not to proceed with the transaction.

Lending Club has a huge cost advantage over traditional banks. Lending Club's cost savings are passed on to borrowers with the finest credit histories, resulting in reduced interest rates. Ideally Lending Club passes the risk of default loans to the Lenders.

Advantages of Lending Club:

LendingClub saves money by using technology to run its online credit marketplace at a lower cost than traditional lending programs, passing the savings on to borrowers in the form of cheaper rates and providing investors with the opportunity to make a profit.

Now that COVID has hastened Americans' migration to digital banking, the customer benefits of this purchase are especially evident. LendingClub, as the only full-spectrum fintech marketplace bank, will be able to leverage our technology and data-driven platform to provide new products and services to our millions of members that will help them both pay less when they borrow and pay less when they pay back their loans.

Advantages to Customers/Borrowers:

• One of the major advantages is convenience. Borrowers can shop through various loans and interest

rates by sitting at home

• Borrowers enjoy a lower rate of interest as compared to traditional lending systems/banks

• The process is simplified, secure and transparent

• It's not an institution like a bank. All the operations are online, and savings are passed on to the

stakeholder's

Advantages to Creditors/Investors:

• Lenders or investors enjoy higher and faster yield

• It becomes a new asset class, which enables diversification, for the investors

• It is also transparent as all the information related to loans is available for investors

How do platforms make money?

Lending club collects fees from both borrowers and investors. If you are an investor, the following fees apply a service fee of one percent (1%) of the amount of any borrower payments received by the payment due date or during applicable grace periods.

Depending on the loan grade and term, borrowers pay a one-time origination charge ranging from 1.11 percent to 5% of the total loan amount.

In the banking industry, efficiency is assessed by the operating ratio, which is defined as marketing costs divided by the total number of loans outstanding. This normally equates to roughly 5% to 7%. As a result, a $100 loan will cost the bank $5 to $7

When borrowers miss payments and loans become late, LendingClub uses best practices from the banking industry to bring delinquent loans back to “current” status.

Lending Club, on the other hand, has an operating ratio of just 2%, which means that issuing a $100 loan costs the company only $2 since they operate fully online, they save on the number of employees and do not incur any location cost.

2. Your team's ultimate goal is to help a client determine whether s/he should invest in p2p loans. What is the final decision that you will help the client make? What is the objective, and how will you evaluate 'better' vs 'worse' decisions? What is the goal of predictive models for this? What will be the potential target variables?

Our final recommendation to the customer is to invest in loans that will be repaid and will not default. We can determine whether decisions are better or worse by examining loan grade, interest rates, loan amount, loan status, dti, yearly income, employment duration, purpose, state address, and actual return and identifying patterns that forecast where defaults occur. For instance, avoiding investing in a purpose that is prone to default is a wiser choice. Predictive models are used to determine which circumstances result in defaults, and how to avoid them. Actual return, actual term, average yearly return, and recoveries may all be seen as possible goal factors that aid in making more informed selections.

---

3. Data exploration

(a) Take a look at the data attributes. How would you categorize these attributes, in broad terms, considering what they pertain to? What are attribute types - which are numeric, categorical, and date variables? What do you think will be the important attributes to consider for your decision task? Which attributes do you think will help determine performance?

- *<u>Categories of Data Attributes according to what they are related to:</u>*

Borrower characteristics - employer title, homeownership, employment length, annual income, state address, zip code

Loan characteristics - purpose, loan amount, funded amount

Lending Club Platform decisions - grade, subgrade, interest rate, verification status

Loan performance: loan status, installment, dti, revol balance, last payment, total payment

- *<u>Types of Data attributes:</u>*

Qualitative - employer title, homeownership, purpose, grade, subgrade, loan status, verification status,

Quantitative - employment length, annual income, loan amount, funded amount, interest rate, installment, dti, revol balance, last payment, total payment

- Important attributes for decision making: annual income, total payment, dti, grade, subgrade
- Attributes that will help determine performance: funded amount, total payment, actual return, average yearly return

---

(b) How will you calculate performance (returns) from a loan? There are multiple ways for calculating this. Outline two ways to calculate returns based on the data attributes; what are their advantages and disadvantages

A. One of the methods for calculating performance returns from a loan is Annual Return.

#Annualized return %: ((lcdf$total_pymnt - lcdf$funded_amnt)/ lcdf$funded_amnt) *(12/36)*100

lcdf$annRet <- ((lcdf$total_pymnt -lcdf$funded_amnt)/lcdf$funded_amnt)*(12/36)*100

Annual return is calculated using a formula where total amount is subtracted by funded amount and then it is divided by funded amount which is multiplied by 33%. This yields a fraction for the annualized return.

Another method for calculating performance returns on a loan is actual return

Actual return: (lcdf$total_pymnt - lcdf$funded_amnt)

lcdf$actualReturn <- lcdf$total_pymnt -lcdf$funded_amnt

Actual Return is calculated using a formula where total payment is subtracted from the funded amount which gives us the return value of the particular loan.

Advantages and Disadvantages:

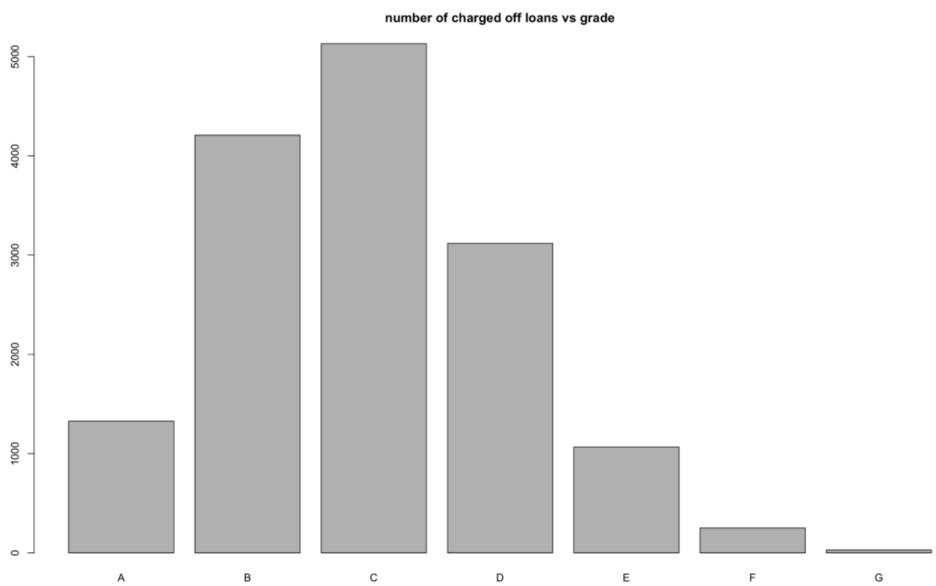
By using Actual Return , we can determine the exact value of the return of a loan after the entire loan is paid off. By using this value we can determine the exact amount of profit earned.

By using Annual return, we can determine the annual return value of a loan which will help investors determine the health of a loan and make a decision to recover based on this value.

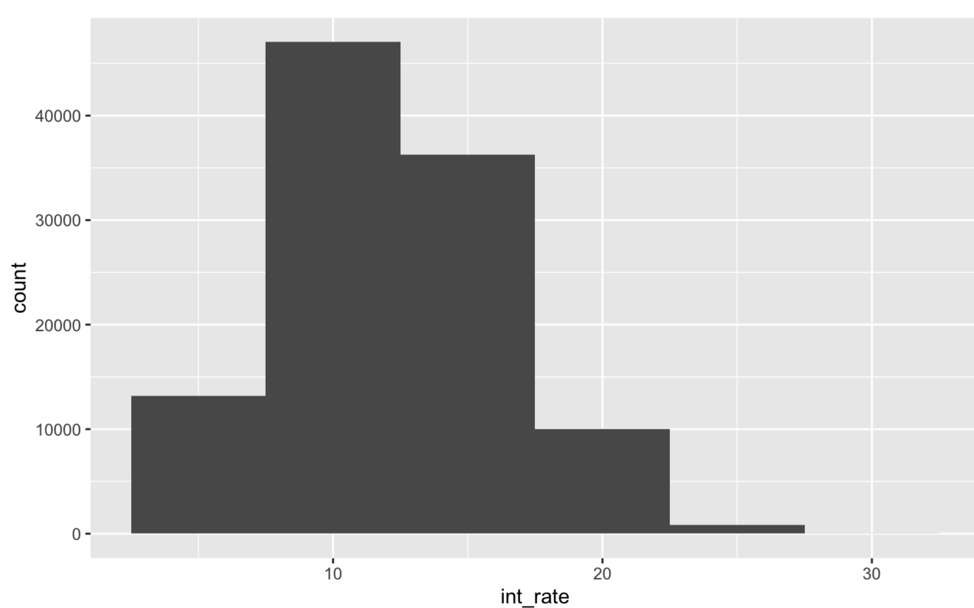
An actual return refers to the actual gain or loss an investor experiences on an investment or in a portfolio.

---

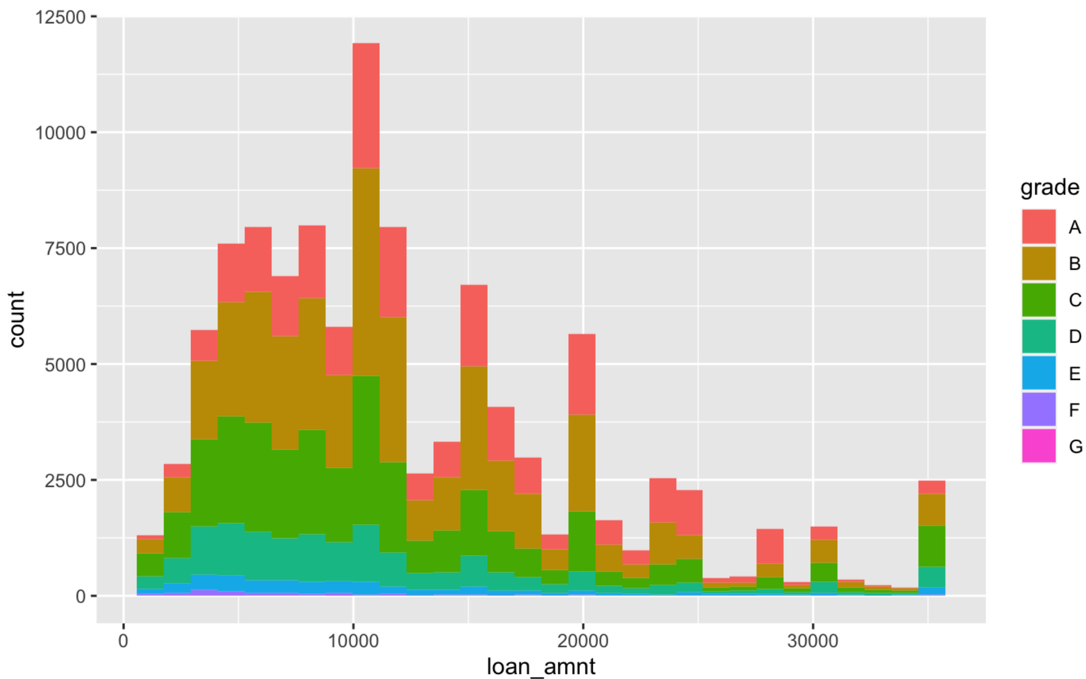
3 (c) Examine the attributes which you think will be useful in your analyses and modeling. Obtain data descriptions and develop some plots to visualize the data. Summarize your observations (you answer should be more than just the figures and plots – what is the ‘story’ from your initial observations)



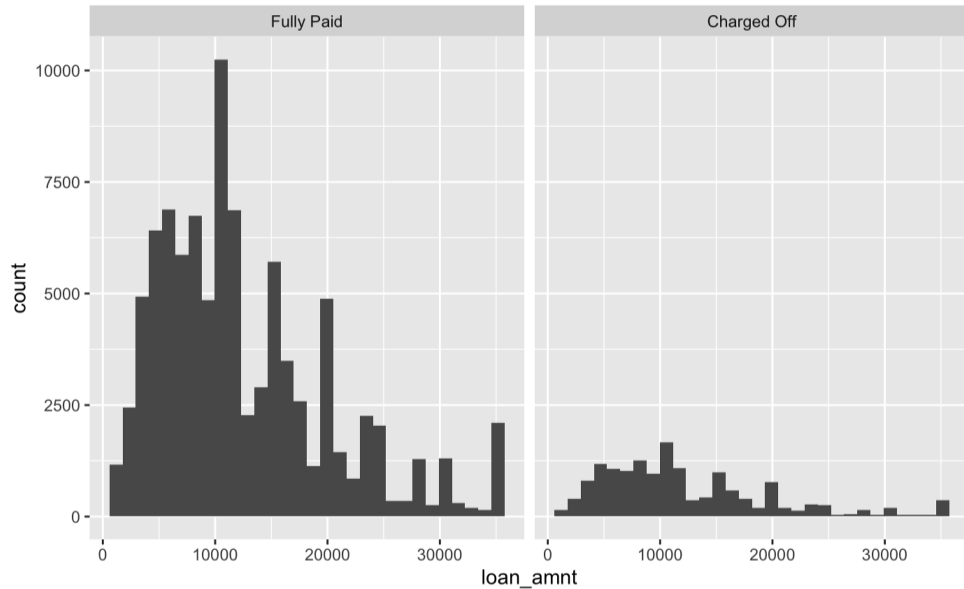
Charged Off loans increase from grade A to C and then decrease to G. This might be because safest loans are anyway paid and riskier loans aren’t granted easily, so a low number of defaults.



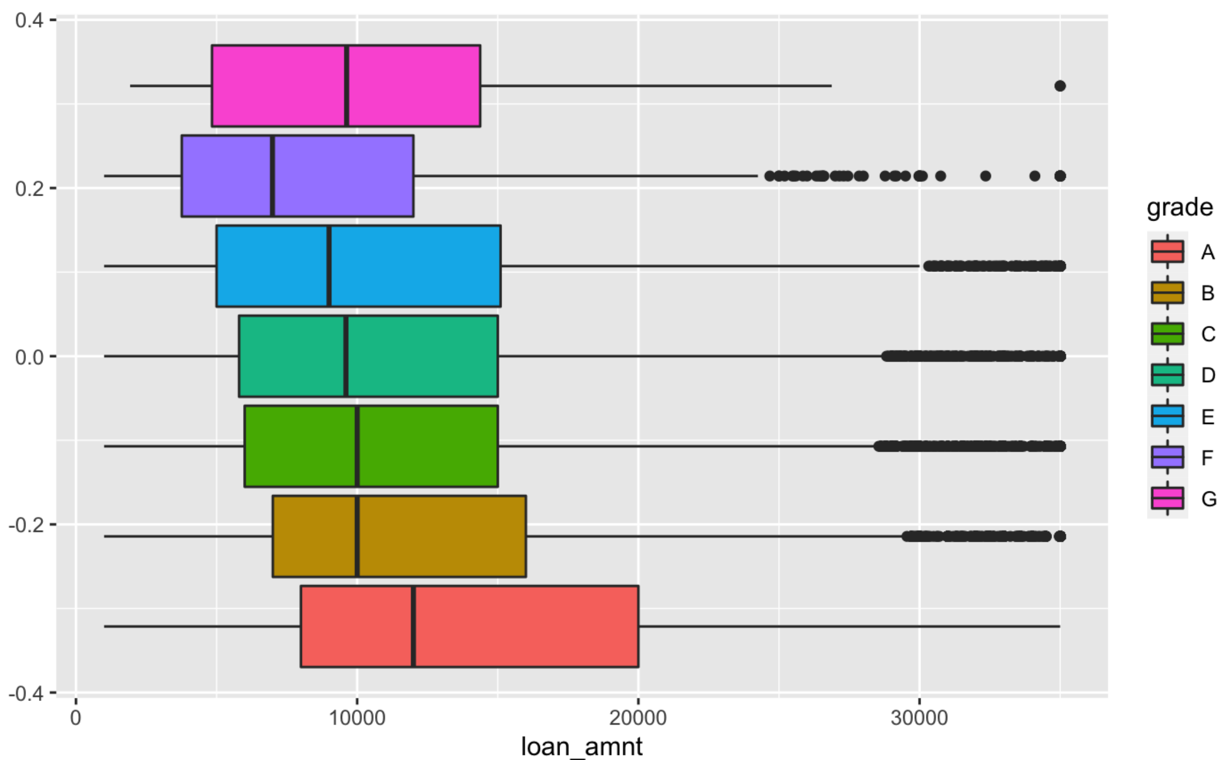
The number of loans issued at lower and higher interest rates is less compared to median interest rates. Higher interest rates are riskier loans, and lower interest rates give less returns. So, lenders might prefer to invest more in loans with median range interest rates



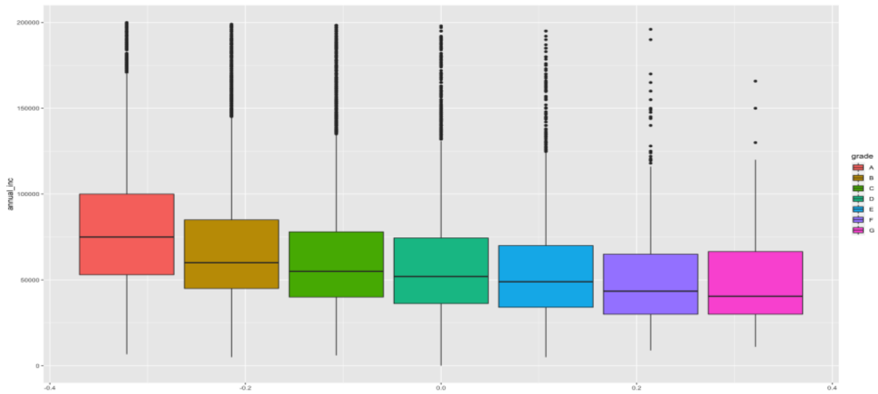
The number of loans issued as per different loan amounts increases and decreases. The returns increase with loan amount so lenders might prefer to invest in the median range (around $10,000), because the returns are better than smaller amounts and the chances of being paid back is better than higher loan amounts. But there's an unusual pattern where the number of loans issued for loan amount $40,000 are more and larger, and part of them belong to grades B and C.



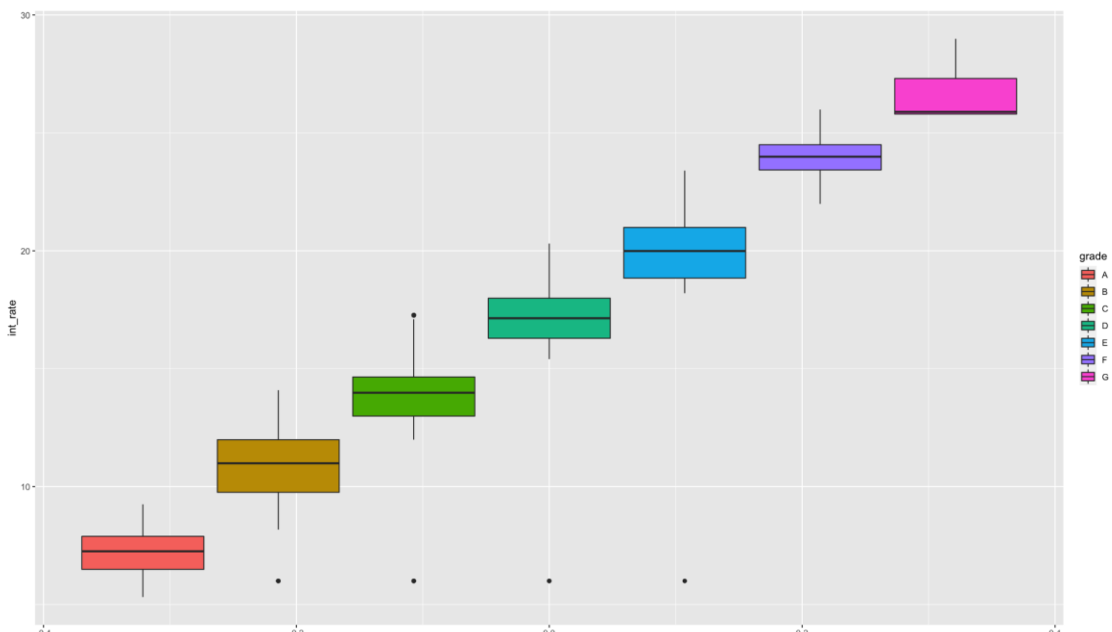
The number of loans that are fully paid and defaulted wrt loan amount follow a similar pattern, they are proportional.



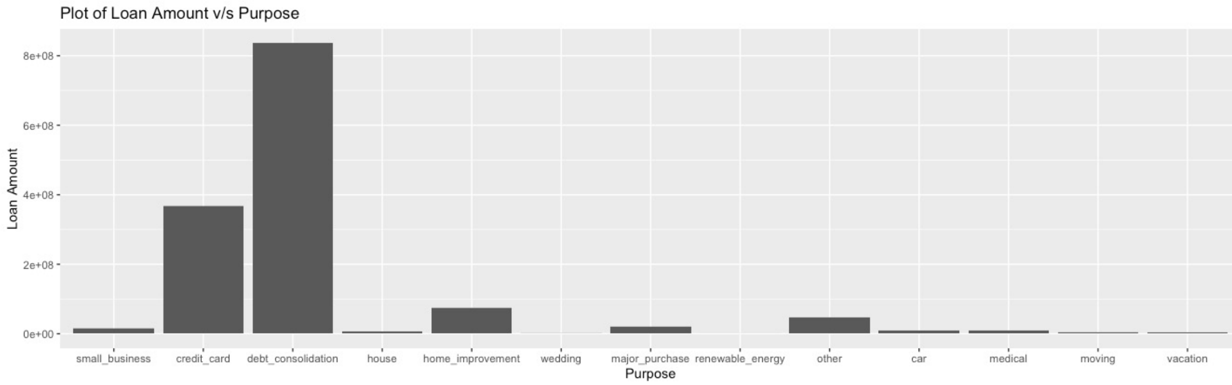
The average loan amount issued for grade A (safest grade) is more, then decreases in Grade B and slightly lowers to Grade F. Unusual pattern is seen for Grade G where the average loan amount increases.



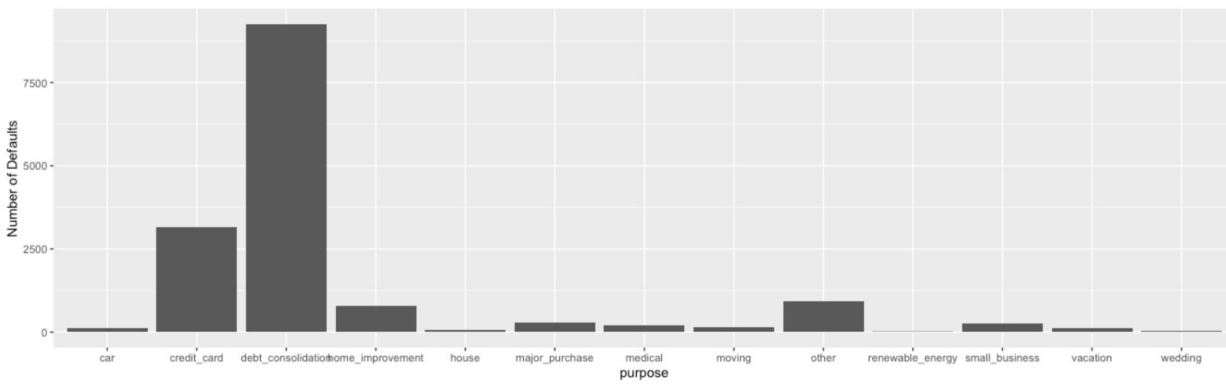
People with higher average annual income tend to pay off loans and their loans are graded safest. As the average annual income decreases, the grade of loans become riskier.



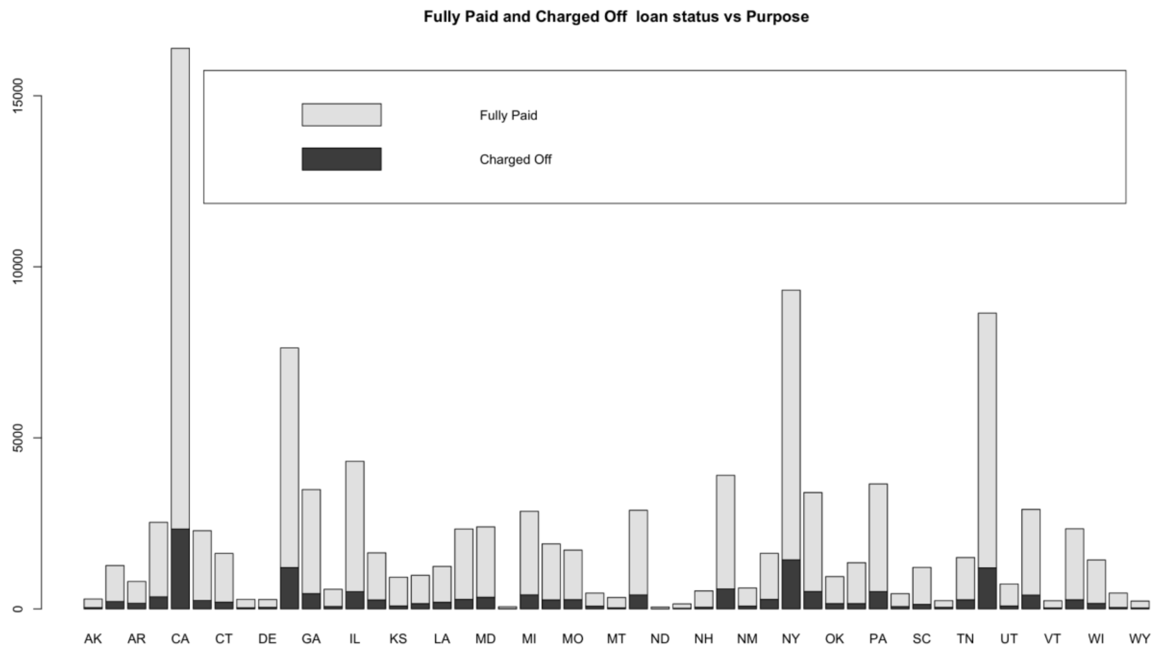
The interest rate of loans increases as the grades of loans become riskier. This is an expected pattern because if loans are riskier the platform assigns higher interest rates.



From the above bar chart, we can see the number of loans taken for each purpose. The highest number of loans are taken for debt consolidation, followed by credit card and home improvement.



Since the number of loans taken for debt consolidation are more, even the defaults are more. Relatively, credit card and home improvement loans. But comparing both the barcharts of loan amount and defaults vs purpose, small business, major purchase, medical, vacation, and other purposes seem to have an increase in defaults pattern wrt loan amount pattern.



Barplot shows the states which have comparatively lower proportion of default loans, which are WV, ND, NH, MT, KS. States which have higher proportions of default loans are AR, SD, TN, NV, AL, MS.

3d i) What are the values for loan_status? Are there values other than “fully paid”, “charged off”? We want to restrict attention to “fully paid” and “charged off” loans, so other values should be removed.
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?

	loan_status	n
1	Charged Off	15377
2	Current	17
3	Fully Paid	94567
4	In Grace Period	2
5	Late (16–30 days)	1
6	Late (31–120 days)	36

The values of loan status other than fully paid and charged off are Current, In grace period, late (16-30 days), Late (31-120 days).

Proportion of defaults (charged off vs fully paid loans) is: 15377: 94567 = 1: 6.15

	grade	nLoans	defaults	defaultRate
	<fct>	<int>	<int>	<dbl>
1	A	23897	1328	0.0556
2	B	37103	4208	0.113
3	C	28619	5130	0.179
4	D	13243	3119	0.236
5	E	3715	1068	0.287
6	F	742	251	0.338
7	G	79	29	0.367

	sub_grade	nLoans	defaults	defaultRate
	<fct>	<int>	<int>	<dbl>
1	A1	3835	100	0.0261
2	A2	3774	155	0.0411
3	A3	3891	188	0.0483
4	A4	5541	371	0.0670
5	A5	6856	514	0.0750
6	B1	6688	570	0.0852
7	B2	7543	765	0.101
8	B3	8119	945	0.116
9	B4	7621	913	0.120
10	B5	7132	1015	0.142

\# … with 25 more rows

Default rate increases with riskier grades because number of defaults increases more than increase in number of loans

---

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?

	grade	nLoans	defaults	defaultRate	avgInterest	stdInterest	avgLoanAMt	avgPmnt
	<fct>	<int>	<int>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
1	A	23897	1328	0.0556	7.21	0.972	14033.	15046.
2	B	37103	4208	0.113	10.9	1.48	12288.	13390.
3	C	28619	5130	0.179	13.9	1.23	11813.	12830.
4	D	13243	3119	0.236	17.3	1.22	11675.	12595.
5	E	3715	1068	0.287	20.0	1.40	11669.	12354.
6	F	742	251	0.338	23.9	0.952	9198.	9633.
7	G	79	29	0.367	26.5	0.955	11202.	11616.

	sub_grade	nLoans	defaults	defaultRate	avgInterest	stdInterest	avgLoanAMt	avgPmnt
	<fct>	<int>	<int>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
1	A1	3835	100	0.0261	5.70	0.348	13771.	14663.
2	A2	3774	155	0.0411	6.43	0.167	13683.	14590.
3	A3	3891	188	0.0483	7.14	0.341	14139.	15194.
4	A4	5541	371	0.0670	7.52	0.360	14327.	15356.
5	A5	6856	514	0.0750	8.28	0.439	14077.	15175.
6	B1	6688	570	0.0852	8.97	0.758	12584.	13617.
7	B2	7543	765	0.101	10.0	0.832	12652.	13752.
8	B3	8119	945	0.116	11.0	0.923	12294.	13391.
9	B4	7621	913	0.120	11.9	0.893	12054.	13252.
10	B5	7132	1015	0.142	12.4	0.943	11866.	12939.

… with 25 more rows

Number of loans issued in each grade decreases from safe to risky loans, because the total number of investments in safe loans are more and risky loans are less.

Loan amounts decrease until grade F and then increase for grade G, which is abnormal.

The average of interest rate by both grade and subgrade, increases from safe loans to risky loans. This is a seen pattern because sub grades are just deeper classifications in grades.

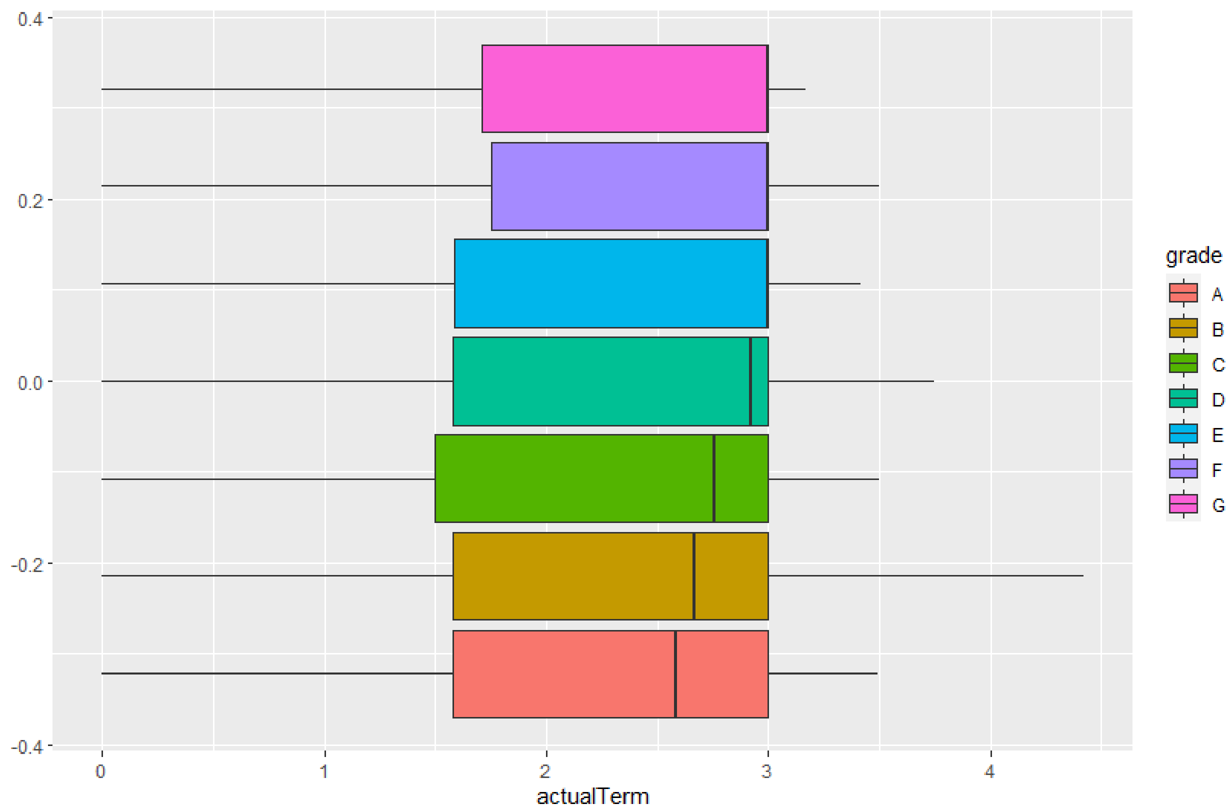
Standard deviation increases and then decreases in both grades and sub grades. This may be because interest rates vary more for medium risk loans, compared to safer grade loans where interest rates are certainly less and for risky loans, interest rates are high.

The minimum interest rates increase from grade A to B and constant from B through E, and there’s a sudden surge in interest rates for F and G.

The maximum interest rate also increases gradually from A to G grade and A1 to G5 subgrade.

---

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 all the grades people having 3 years actual period lie in the 75th percentile whereas the median lies between 2.5 and 3.

(iv) What is 'recoveries'? Can we assume that recoveries are only for Charged_off loans? The data has multiple attributes on recoveries – what is the total amount of recoveries? For charged-off loans, does total_pymnt include recoveries?

The recovery rate is the extent to which principal and accrued interest on defaulted debt can be recovered. (Source: https://www.investopedia.com/terms/r/recovery-rate.asp)

Debt recovery is when a loan—such as a credit card balance—continues to go unpaid, and a creditor hires a third party, known as a collection service, to focus on collecting the money.

Source:
https://www.debt.org/advice/recovery/#:~:text=Debt%20recovery%20is%20when%20a,correlated%20to%20your%20credit%20score.

```
> lcdf %>% group_by(loan_status) %>%summarise(avgRec=mean(recoveries))
# A tibble: 6 x 2
  loan_status          avgRec
  <chr>                 <dbl>
1 Charged Off            926.
2 Current                  0
3 Fully Paid               0
4 In Grace Period          0
5 Late (16-30 days)        0
6 Late (31-120 days)       0
```

```
> lcdf %>% group_by(loan_status) %>%summarise(avgRec=mean(recoveries), avgPmnt=mean(total_pymnt), mean(total_rec_prncp), mean(total
_rec_int), mean(total_rec_late_fee))
# A tibble: 6 x 6
  loan_status        avgRec avgPmnt `mean(total_rec_prncp)` `mean(total_rec_int)` `mean(total_rec_late_fee)`
  <chr>               <dbl>   <dbl>                   <dbl>                 <dbl>                      <dbl>
1 Charged Off          926.   7880.                   5194.                 1756.                      3.78
2 Current                0   15500.                  12537.                 2930.                     33.2
3 Fully Paid             0   14679.                  12741.                 1937.                      0.762
4 In Grace Period        0   12795.                  10751.                 2032.                     12.0
5 Late (16-30 days)      0   38451.                  30505.                 7891.                     54.2
6 Late (31-120 days)     0   12846.                  10479.                 2340.                     27.8
```

---

v) 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 is calculated by calculating the difference of total payment received by the investor and the amount he funded in proportion to the funded amount and then is multiplied by the actual term.

The actual term is the difference from the issue date to the last payment date.

Percentage annual return = 100*((Payment – Funded Amount)/(Funded Amount))*(1/Actual Term)

Returns from 'charged off' loans are negative as expected. As loan grade changes, i.e., as it becomes riskier, the returns become more negative.

Average return values are less than the average interest rates. There are multiple reasons behind this. One of the reasons is that the loans are completed before their original term. Other reasons include adjustments in return rate by Lending Club based on future charged off rate and the service fee(1%) that the lender pays.

If I wanted to invest in loans, I would diversify my investments across loans from different grades based on average returns and default rate

```
> lcdf %>% group_by(grade) %>% summarise(average_annual_return= mean(annRet),average_int_rate=mean(int_rate))
# A tibble: 7 x 3
  grade average_annual_return average_int_rate
  <fct>                 <dbl>            <dbl>
1 A                      2.35             7.21
2 B                      2.95            10.9
3 C                      2.85            13.9
4 D                      2.81            17.3
5 E                      2.53            20.0
6 F                      2.95            23.9
7 G                      1.51            26.5
```

```
# A tibble: 7 x 8
  grade nLoans defaults defaultRate avgInterest stdInterest avgLoanAMt avgPmnt
  <fct>  <int>    <int>       <dbl>       <dbl>       <dbl>      <dbl>   <dbl>
1 A      24857     1369      0.0551        7.21       0.973     14349.  15388.
2 B      37891     4264      0.113        10.9        1.48      12505.  13633.
3 C      29162     5206      0.179        13.9        1.23      12048.  13094.
4 D      13463     3165      0.235        17.3        1.22      11898.  12836.
5 E       3791     1090      0.288        20.0        1.40      11922.  12623.
6 F        753      252      0.335        23.9        0.955      9435.   9953.
7 G         83       31      0.373        26.5        0.952     11585.  11938.
```

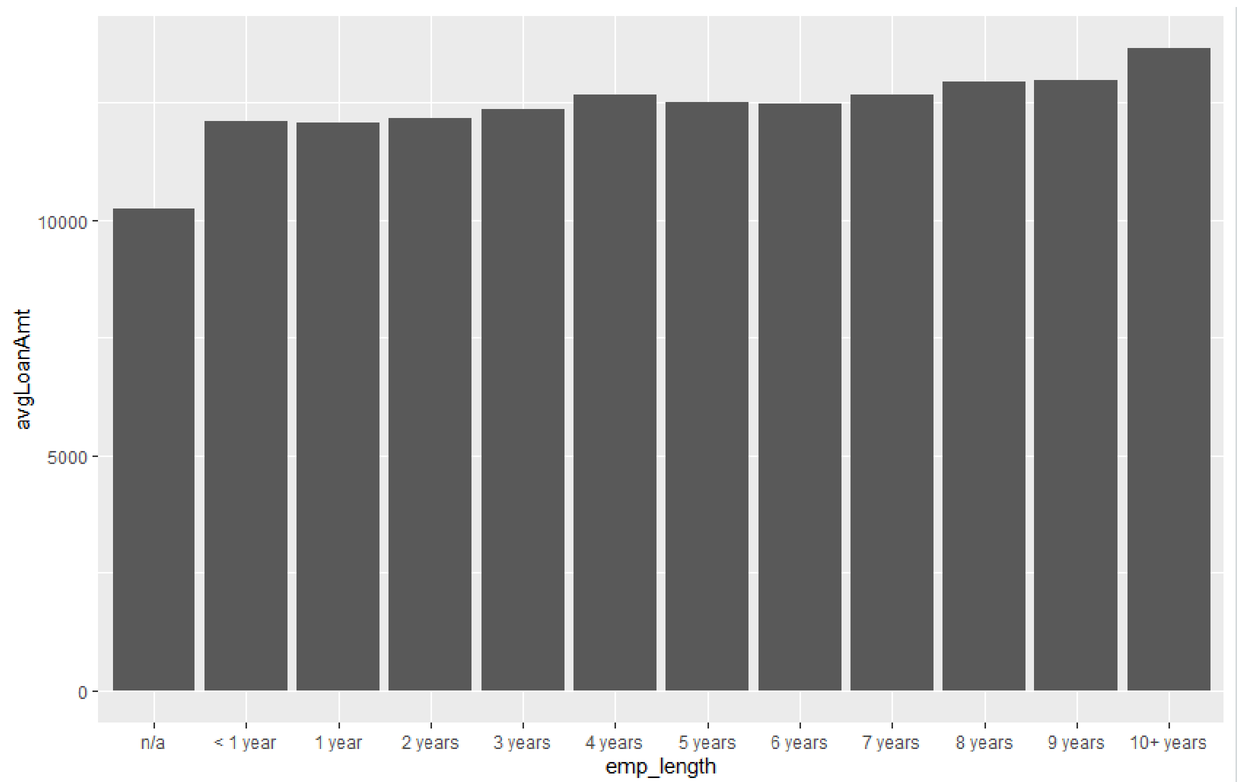
(vi) What are people borrowing money for (purpose)? Examine how many loans, average amounts, etc. by purpose? Do loan amounts vary by purpose? Do defaults vary by purpose? Does loan-grade assigned by Lending Club vary by purpose?

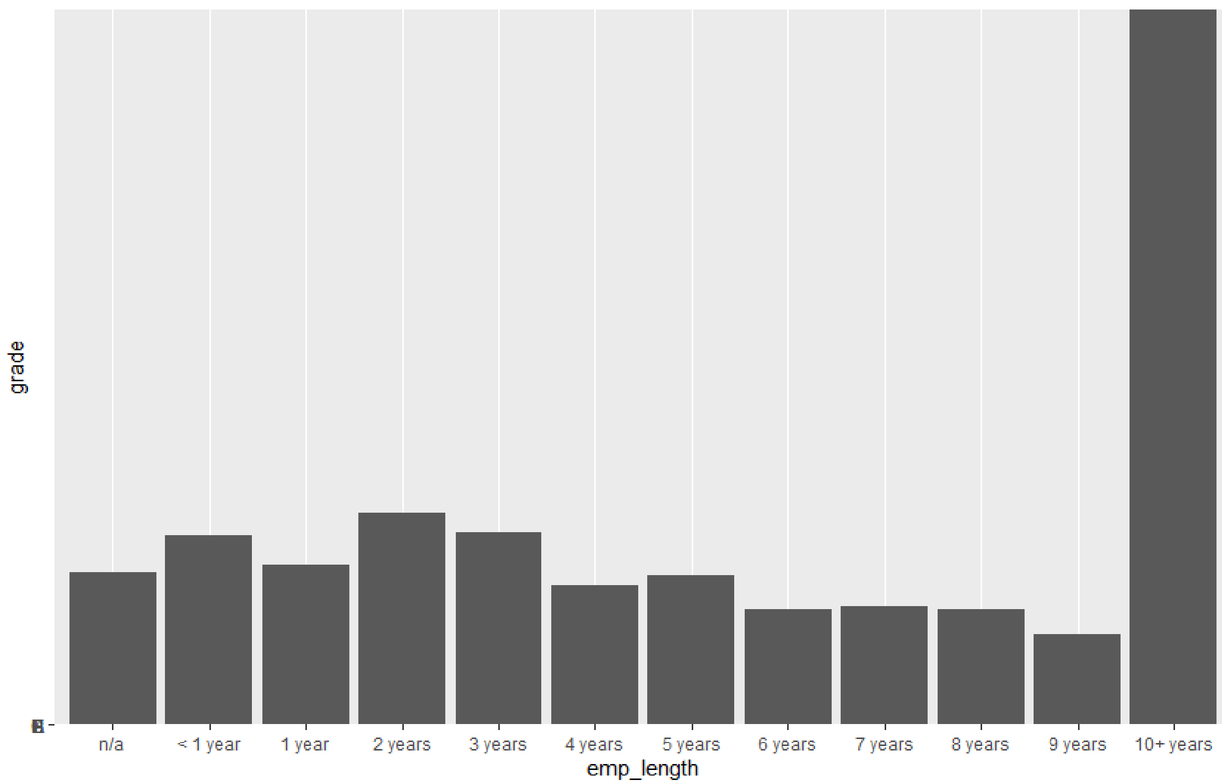
Debt consolidation and credit cards comprise ~82% of the total loans. There's quite a big spread for average loan amounts by purpose. Debt consolidation, credit card, house, and small business have the high loan amounts(~$13k) being sanctioned as they require quite a big capital and on the other side of the spectrum is vacation which has the lowest average loan amount(~$5.6k) Default rate also has varied quite a bit where at one end is a small business with highest default rate at around 22% which is possible since they are susceptible to collapse and on the other end car, credit has lowest default rate at around 11%.

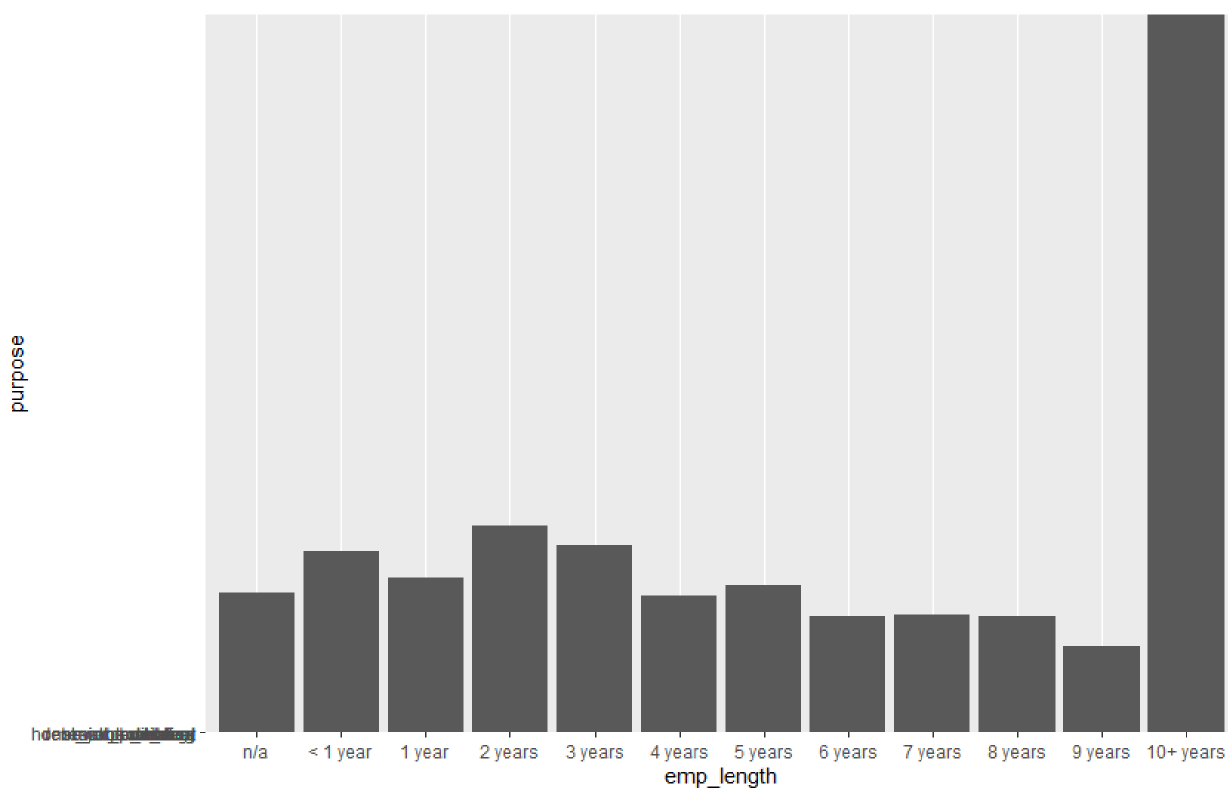
(vii) Consider some borrower characteristics like employment-length, annual-income, fico-scores (low, high). How do these relate to loan attributes like, for example, loan_amout, loan_status, grade, purpose, actual return, etc.

```
# A tibble: 12 x 2
   emp_length     n
   <chr>      <int>
 1 < 1 year    8791
 2 1 year      7339
 3 10+ years  34365
 4 2 years     9806
 5 3 years     8891
 6 4 years     6506
 7 5 years     6980
 8 6 years     5450
 9 7 years     5577
10 8 years     5456
11 9 years     4190
12 n/a         6649
```

	n/a	< 1 year	1 year	2 years	3 years	4 years	5 years	6 years	7 years	8 years	9 years	10+ years
A	1230	1984	1555	2198	1911	1470	1581	1213	1275	1234	890	8316
B	2177	2945	2446	3291	3108	2209	2404	1914	1940	1938	1508	12011
C	1832	2338	2074	2639	2361	1736	1832	1456	1504	1431	1133	8826
D	1007	1110	941	1221	1102	819	859	675	652	630	475	3972
E	324	337	278	376	329	224	248	154	172	179	152	1018
F	76	70	35	74	67	45	50	34	27	41	30	204
G	3	7	10	7	13	3	6	4	7	3	2	18




grade
n/a
< 1 year
1 year
2 years
3 years
4 years
5 years
6 years
7 years
8 years
9 years
10+ years
emp_length


purpose
n/a
< 1 year
1 year
2 years
3 years
4 years
5 years
6 years
7 years
8 years
9 years
10+ years
emp_length

emp_length	nLoans	ln_status_chrgof	ln_status_fp	defaultRate	avgIntRate	avgLoanAmt
n/a	6649	1345	5301	0.2022861	12.53279	10249.10
< 1 year	8791	1269	7515	0.1443522	12.07674	12106.20
1 year	7339	1097	6239	0.1494754	12.17336	12081.83
2 years	9806	1327	8471	0.1353253	12.12269	12181.19
3 years	8891	1265	7622	0.1422787	12.12850	12347.37
4 years	6506	895	5608	0.1375653	12.08256	12660.42
5 years	6980	983	5990	0.1408309	12.13365	12516.45
6 years	5450	757	4692	0.1388991	12.16497	12476.46
7 years	5577	737	4838	0.1321499	12.08804	12655.65
8 years	5456	724	4732	0.1326979	11.97778	12934.48
9 years	4190	598	3589	0.1427208	12.05757	12969.00
10+ years	34365	4380	29970	0.1274553	11.84191	13665.10

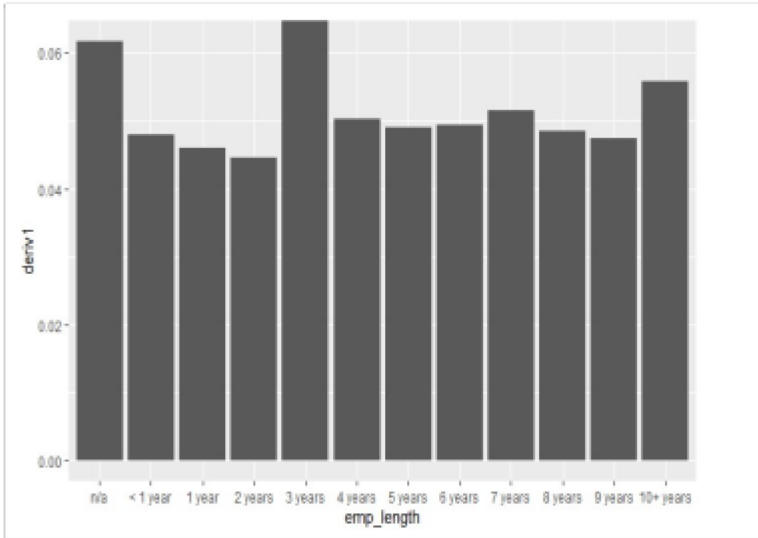
(viii) Generate some (at least 3) new derived attributes which you think may be useful for predicting default. and explain what these are. For these, do an analysis as in the questions above (as reasonable based on the derived variables).

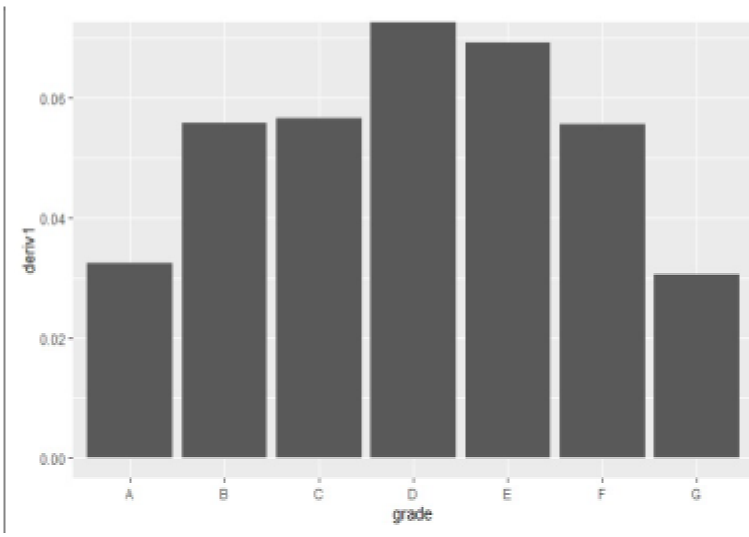
The attributes derived are:

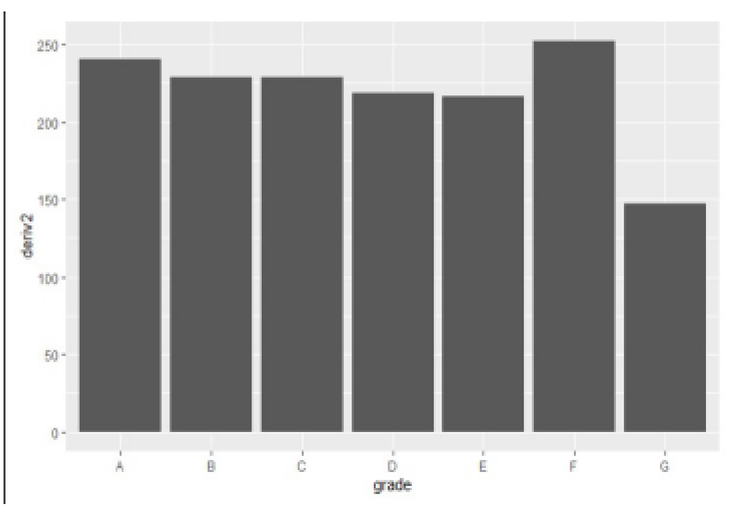
a) num_acc_ratio - It is the ratio between the number of accounts having dues for over 120 days (num_accts_ever_120_pd) and number of currently active accounts (num_actv_rev_tl). This will help in understanding and figuring out account information of applicants that allows finding if the applicant often has dues in their accounts.

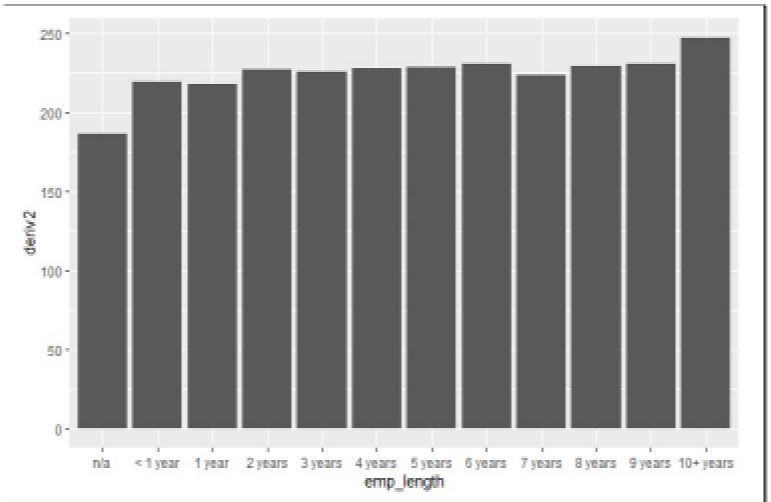
b) inc_instal_ratio - It calculates the ratio between the annual income of applicants and the value of the installments they are going to pay. This will help in understanding which applicants have a higher share ratio of income and installment amount.

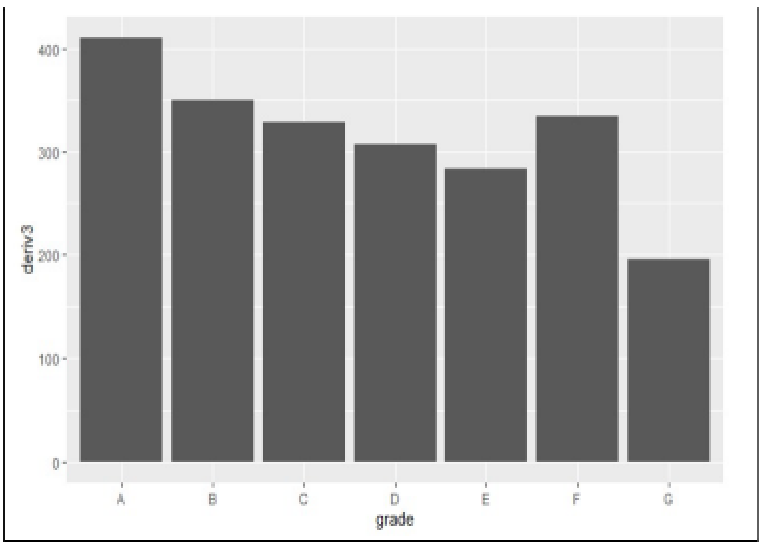
c) available_bal_instal_ratio - This ratio is one between the difference of total balance in the applicant’s account from the revolving balance of the person and the installment.

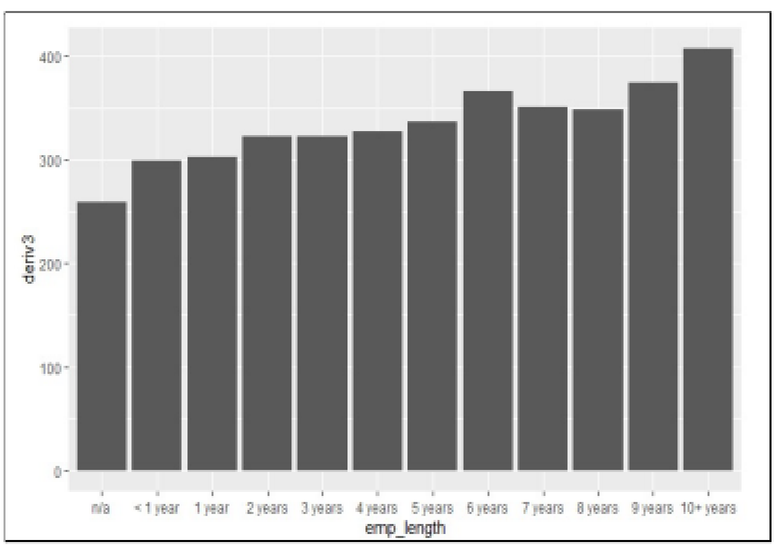

0.06
0.04
0.02
0.00
deriv1
n/a
< 1 year
1 year
2 years
3 years
4 years
5 years
6 years
7 years
8 years
9 years
10+ years
emp_length


0.06
0.04
0.02
0.00
deriv1
A
B
C
D
E
F
G
grade


250
200
150
100
50
0
deriv2
A
B
C
D
E
F
G
grade


250
200
150
100
50
0
deriv2
n/a
< 1 year
1 year
2 years
3 years
4 years
5 years
6 years
7 years
8 years
9 years
10+ years
emp_length


400
300
200
100
0
deriv3
A
B
C
D
E
F
G
grade


400
300
200
100
0
deriv3
n/a
< 1 year
1 year
2 years
3 years
4 years
5 years
6 years
7 years
8 years
9 years
10+ years
emp_length

	emp_length	nLoans	ln_status_chrgof	deriv1	deriv2	deriv3	ln_status_fp	defaultRate
	<fct>	<int>	<int>	<dbl>	<dbl>	<dbl>	<int>	<dbl>
1	n/a	6148	1296	0.0617	186.	259.	4852	0.211
2	< 1 year	8104	1204	0.0479	219.	299.	6900	0.149
3	1 year	6649	960	0.0460	218.	303.	5689	0.144
4	2 years	8987	1206	0.0446	227.	322.	7781	0.134
5	3 years	8046	1088	Inf	225.	322.	6958	0.135
6	4 years	5892	775	0.0503	228.	327.	5117	0.132
7	5 years	6046	841	0.0490	229.	337.	5205	0.139
8	6 years	4712	632	0.0494	230.	366.	4080	0.134
9	7 years	5124	712	0.0515	223.	351.	4412	0.139
10	8 years	4990	698	0.0485	229.	349.	4292	0.140
11	9 years	3908	522	0.0474	230.	375.	3386	0.134
12	10+ years	31394	3851	0.0558	247.	407.	27543	0.123

... with 4 more variables: avgIntRate <dbl>, avgLoanAmt <dbl>, avgActRet <dbl>,
avgActTerm <dbl>

3d2 *Summary:*

From Grade A to G	Number of Charged Off loans ↑ and ↓ Average loan amount ↓ and ↑ Annual income ↓ Interest rate ↑ Default rate ↑ Actual term ↑

From our preliminary analysis, we observed that lenders tend to invest money in grades B and C, median range interest rates and median range loan amount.

Attributes like annual income, employment length, purpose, region, dti can be good indicators to minimize charged off loans.

3.e) 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 many rows with missing values in the data set. Missing values may be because of many reasons, some maybe due to human error or maybe just because they were literally not available.

There are certain columns with more than 60% of the data values missing. We are eliminating these columns because it does not make sense to impute them because the significance of the columns may be lost, and the output may be something totally unexpected.

There are different ways in which we impute data, like mean imputation, in which we substitute the missing data by the mean of the remaining variables. Mean imputation does not generally preserve the relationship among variables.

Another way is regression imputation in which we can impute by predicting the missing values from one or more non missing values. The disadvantage however is that it might still lead to biased parameter estimates.

So, in general there is no particular method for imputation. It all is subjective to the kind of values we are imputing.

So, there were 59 variables which had 60% of rows with NA values and hence it was viable to remove those from our data set.

In a particular case a variable named "months since last Delinquency" has missing values.NO what we can do here is check the average of this variable per grade and then impute the missing values in the respective grade instead of taking average of entire column because its values will be generally high for lower grades compared to higher grade.

---

4. 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 for leakage considerations, and explain why.

In statistics and machine learning, leakage (also known as data leakage or target leakage) is the use of information in the model training process which would not be expected to be available at prediction time, causing the predictive scores (metrics) to overestimate the model's utility when run in a production environment.

[Source: https://en.wikipedia.org/wiki/Leakage_(machine_learning)]

Leakage Variables	funded_amnt_inv, total_pymnt, total_pymnt_inv, total_rec_prncp, total_rec_int
Variables which are not useful	funded_amnt_inv, term, emp_title, pymnt_plan,hardship_flag, title, zip_code, title, zip_code, 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, policy_code,annRet,annual_return,pct_annual_r eturn,collections_12_mths_ex_med,inq_last_6 m ths,actualTerm,installment, emp_length,verification_status,num_tl_30dpd,a c c_now_delinq,chargeoff_within_12_mths,num_ tl _90g_dpd_24m,delinq_amnt,tax_liens,pub_rec, d elinq_2yrs,initial_list_status,tot_coll_amt,num_ a ccts_ever_120_pd,mths_since_last_delinq,mths _ since_recent_inq,percent_bc_gt_75,debt_settle

ment_flag,earliest_cr_line,pub_rec_bankruptcies

,application_type,openTototal,creditPerRevolv_a

cc

Variables with missing values:

Variables	Proportions (%)
mths_since_last_delinq	48.24
revol_util	0.05
bc_open_to_buy	0.96
bc_util	1.4
mo_sin_old_il_acct	3.54
mnths_since_recent_bc	0.94
mnths_since_recent_inq	11.16
num_tl_120dpd_2m	5.87
percent_bc_gt_75	1.07

Variables which have more than 60% missing values:

1	id	**29**	sec_app_earliest_cr_line
2	member_id	**30**	sec_app_inq_last_6mths
3	url	**31**	sec_app_mort_acc
4	desc	**32**	sec_app_open_acc

5	mths_since_last_record	33	sec_app_revol_util
6	last_pymnt_d	34	sec_app_open_act_il
7	next_pymnt_d	35	sec_app_num_rev_accts
8	mths_since_last_major_derog	36	sec_app_chargeoff_within_12_mths
9	annual_inc_joint	37	sec_app_collections_12_mths_ex_med
10	dti_joint	38	sec_app_mths_since_last_major_derog
11	verification_status_joint	39	hardship_type
12	open_acc_6m	40	hardship_reason
13	open_act_il	41	hardship_status
14	open_il_12m	42	deferral_term
15	open_il_24m	43	hardship_amount
16	mths_since_rcnt_il	44	hardship_start_date
17	total_bal_il	45	hardship_end_date
18	il_util	46	payment_plan_start_date
19	open_rv_12m	47	hardship_length
20	open_rv_24m	48	hardship_dpd
21	max_bal_bc	49	hardship_loan_status
		50	orig_projected_additional_accrued_interest
22	all_util	51	hardship_payoff_balance_amount
		52	hardship_last_payment_amount
23	inq_fi	53	debt_settlement_flag_date
24	total_cu_tl	54	settlement_status
25	inq_last_12m	55	settlement_date
26	mths_since_recent_bc_dlq	56	settlement_amount
27	mths_since_recent_revol_delinq	57	settlement_percentage
28	revol_bal_joint	58	settlement_term

5. 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).

In this case, we'll use AUC (area under the curve) as the ROC graph's under-the-curve measure. We can also use accuracy as a metric; however, accuracy is less derivable when data is skewed to one datapoint of the answer variable. For example, 88 percent of the loans in this set are paid in full, so if we guess that all of the loans will be paid in full, we'll be 88 percent correct, which isn't bad but isn't particularly useful.

```
> aucAll[aucAll>0.5]
            int_rate          installment                grade            sub_grade
           0.6710002            0.5141885            0.5896832            0.6154455
          annual_inc          loan_status              purpose           addr_state
           0.5739206            1.0000000            0.5060059            0.5232720
                 dti       inq_last_6mths mths_since_last_delinq           revol_bal
           0.5417009            0.5401899            0.5178456            0.5114131
          revol_util            total_acc            out_prncp        out_prncp_inv
           0.6088658            0.5902207            0.9411765            0.9411765
         total_pymnt      total_pymnt_inv      total_rec_prncp        total_rec_int
           0.7400587            0.7399210            0.7979832            0.6510411
  total_rec_late_fee           recoveries collection_recovery_fee     last_pymnt_amnt
           0.6492393            0.8885673            0.8639201            0.5811296
        tot_coll_amt          tot_cur_bal      total_rev_hi_lim acc_open_past_24mths
           0.5268405            0.5579658            0.5453020            0.6372279
         avg_cur_bal       bc_open_to_buy              bc_util     mo_sin_old_il_acct
           0.5831957            0.6214151            0.5896507            0.5508035
mo_sin_old_rev_tl_op mo_sin_rcnt_rev_tl_op        mo_sin_rcnt_tl mths_since_recent_inq
           0.5532144            0.6120200            0.5759190            0.5123944
      num_actv_bc_tl      num_actv_rev_tl          num_bc_sats            num_bc_tl
           0.5773665            0.5235683            0.5306591            0.6000929
      num_op_rev_tl         num_rev_accts   num_rev_tl_bal_gt_0        pct_tl_nvr_dlq
           0.5365140            0.6153021            0.5300233            0.5097450
     percent_bc_gt_75       tot_hi_cred_lim     total_bal_ex_mort total_il_high_credit_limit
           0.5509718            0.5741519            0.5022078            0.5290683
              annRet
           0.9739871
```

We have used AUC measure to identify the relationship between predictor (independent variable)
(Numeric variable) and dependent variable.
annualRet gives 0.98 AUC
And for the following variable the AUC is more 0.75

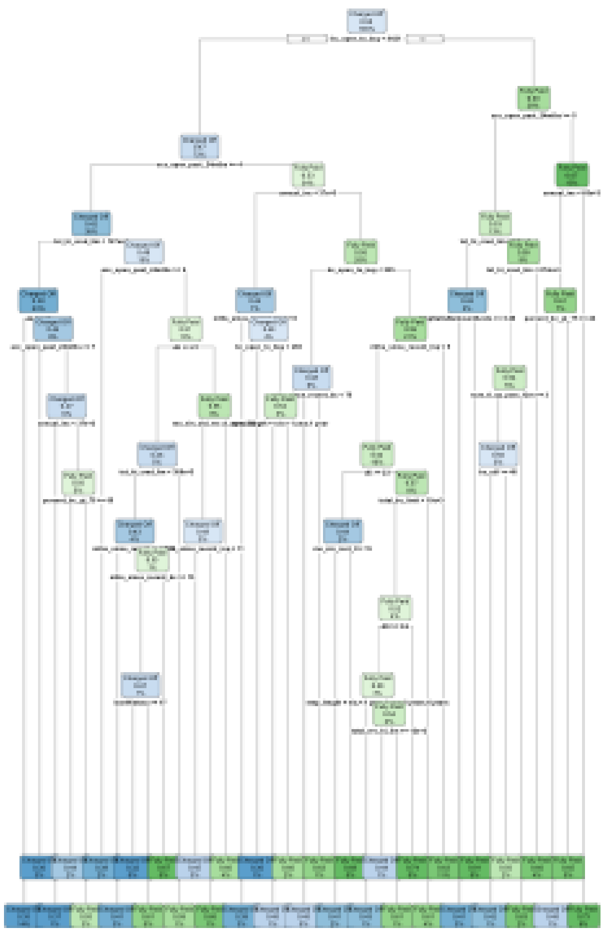
total_rec_prncp
last_pymnt_amnt
total_pymnt_inv
total_pymnt

But these variables have the issue of data leakage as they were generated after the loan was granted/funded or closed.
Rest of the variables' AUC lies around 0.5

Part-B

6. Develop decision tree models to predict default.
(a) Split the data into training and validation sets. What proportions do you consider, why?

Data partitioning is a crucial aspect of data mining. Normally, data sets are divided into two categories: training and testing. The proportions we're looking at are 75% for the training set and 25% for the testing set. We're considering giving the majority of the data to training sets so that the model may learn all of the different types of possible patterns for defining the problem.



(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?

For the Decision Tree Model, we first removed the variables with 100% missing values, then divided the data 70:30 into training and validation. Then, using Training Data, we created a Decision Tree. We found that the actual term, actual return where 2 variables which had a very high importance so we removed them.

The model performance:

–rpart:

Variable importance

```
> lcDT1b$variable.importance
                   bc_util             bc_open_to_buy                  revol_bal
              4161.2716727               4141.9912265               3880.3285924
                annual_inc          mo_sin_old_il_acct                  total_acc
              3124.8785256               2920.6575099               2523.7327100
      mths_since_recent_bc                    open_acc           percent_bc_gt_75
              2484.4521366               2264.0460993               1923.3531614
    mths_since_last_delinq        mths_since_recent_inq                    purpose
              1741.8654321               1657.9258385               1573.8605812
            inq_last_6mths              home_ownership                delinq_2yrs
               725.1004369                581.2624767                511.6966176
       initial_list_status                     pub_rec       pub_rec_bankruptcies
               389.6483343                310.3940108                199.2926853
                tax_liens collections_12_mths_ex_med             acc_now_delinq
                83.7138671                 30.6825293                 16.4798668
  chargeoff_within_12_mths                 delinq_amnt             num_tl_120dpd_2m
                14.5468985                 13.7374315                  0.7305759
```

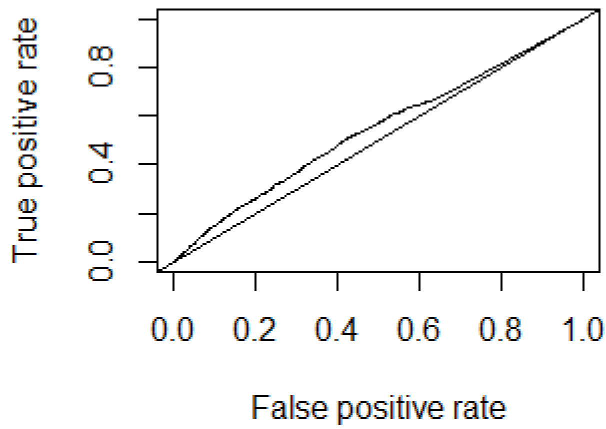
Confusion matrix for training data:

```
> table(pred = predTrn, true=lcdfTrn$loan_status)
             true
pred          Fully Paid Charged Off
  Fully Paid       54382         979
  Charged Off      16547       10550
```

Confusion matrix for Test data: Accuracy 64.10%

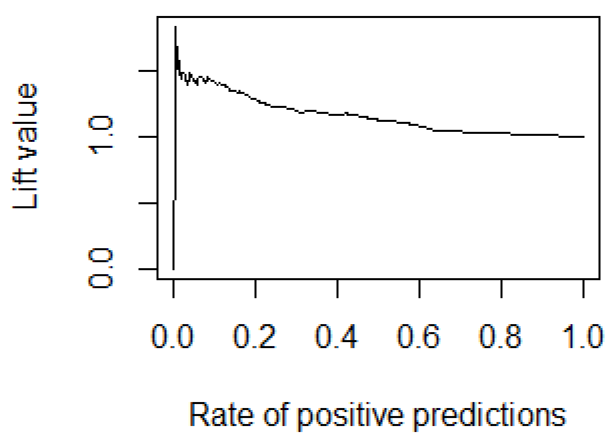
```
> table(pred = predict(lcDT1b,lcdfTst, type='class'), true=lcdfTst$loan_status)
             true
pred          Fully Paid Charged Off
  Fully Paid       16111        2329
  Charged Off       7527        1519
```

ROC Curve:



AUC is 0.5440

LIFT Curve:



C50:

Variable Importance:

```
Attribute usage:

100.00% bc_open_to_buy
100.00% mths_since_recent_bc
 97.78% annual_inc
 89.05% inq_last_6mths
 87.83% purpose
 61.71% mo_sin_old_il_acct
 60.51% home_ownership
 52.76% collections_12_mths_ex_med
 44.54% percent_bc_gt_75
 44.19% tax_liens
 43.10% mths_since_recent_inq
 38.43% bc_util
 36.73% mths_since_last_delinq
 35.58% revol_bal
 32.63% total_acc
 26.49% open_acc
 21.80% delinq_2yrs
 21.42% pub_rec_bankruptcies
 16.44% initial_list_status
 15.76% acc_now_delinq
 14.71% pub_rec
 12.25% chargeoff_within_12_mths
```

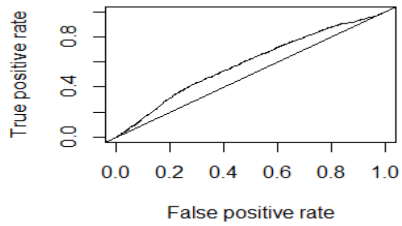
Confusion Matrix for training data:

```
             Reference
Prediction    Fully Paid Charged Off
  Fully Paid       43511        2848
  Charged Off      27418        8681
```

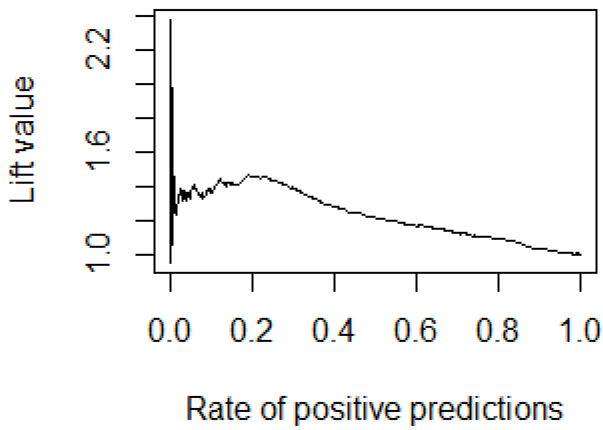
Confusion Matrix for Test Data: 57.44%

```
> table(pred = predTst[,'Fully Paid' ] > CTHRESH, true=lcdfTst$loan_status)
       true
pred    Fully Paid Charged Off
  FALSE       9948        2107
  TRUE       13690        1741
```

ROC Curve



Lift Curve:



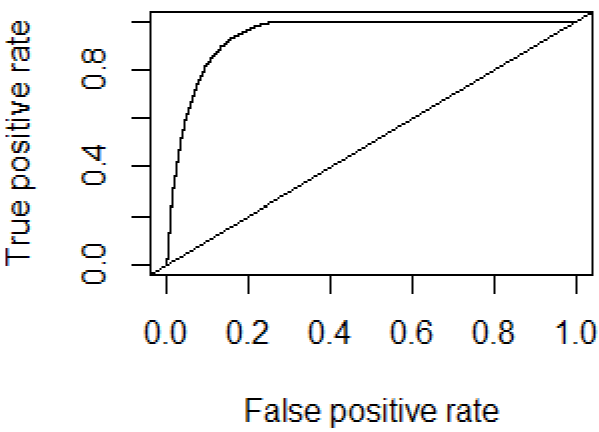
(c) Identify the best tree model. Why do you consider it best?
Describe this model – in terms of complexity (size). Examine variable importance.
How does this relate to your uni-variate analysis in Question 4 above? Briefly describe how variable importance is obtained (the process used in decision trees).

When compared to c50, the Decision Tree model appears to be a better model because its accuracy is higher, and when comparing the confusion matrix, the Decision Tree model has less misclassifications of both Fully Paid and Charged Off.
When we look at the table of variable importance, we can see that
we'll see that bc_util, bc_open_to_buy and revol_bal has higher significance as compared to the other variables and it's backed by our analysis.

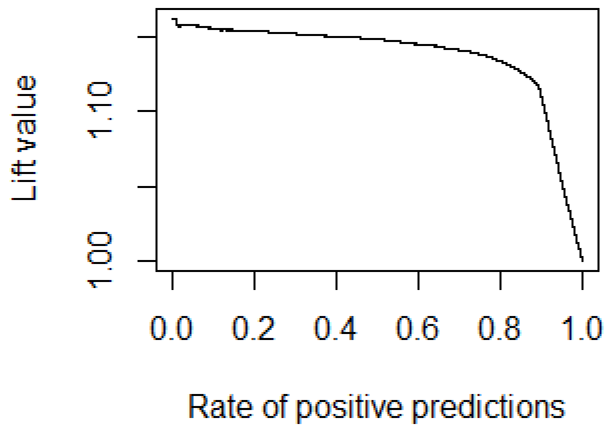
7. Develop a random forest model. (Note the 'ranger' library can give faster computations) What parameters do you experiment with, and does this affect performance? Describe the best model in terms of number of trees, performance, variable importance. Compare the performance of random forest and best decision tree model from the previous question. Do you find the importance of variables to be similar/different? Which model would you prefer, and why?

```
> table(pred = scoreTrn$predictions[, "Fully Paid"] > 0.7, actual=lcdfTrn$loan_status)
       actual
pred    Fully Paid Charged Off
  FALSE        565        8802
  TRUE       70364        2727
> scoreTst <- predict(rfModel1,lcdfTst)
> table(pred = scoreTst$predictions[, "Fully Paid"] > 0.7, actual=lcdfTst$loan_status)
       actual
pred    Fully Paid Charged Off
  FALSE        203        2961
  TRUE       23435         887
```

ROC Curve:



AUC: 0.9470



8. 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?

One can consider the average interest rate on loans for expected profit – is this a good estimate of your profit from a loan? For example, suppose the average int_rate in the data is 12%; so, after 3 years, the $100 will be worth (100 + 3*12) = 136, i.e a profit of $36. Now, is 12% a reasonable value to expect – what is the return you calculate from the data? Explain what value of profit you use.

For the loans that are fully paid, the company on average makes a profit of 15.21% per year. Considering we have invested $100 in paid loans; we will be earning $15.21 a year making the combined profit for three years to be $45.63.
While in the case of charged-off loans, the company incurs a loss of 8.22% per year, so the combined loss for three years will be 24.66%
Considering we have $100 in charged-off loans, we incur a loss of $24.66 over three years.

For a loan that is charged off, will the loss be the entire invested amount of $100? The data shows that such loans do show some partial returned amount. Looking at the returned amount for charged off loans, what proportion of invested amount can you expect to recover? Is this overly optimistic? Explain which value of loss you use. (15377)

No, for a charged-off loan, the loss isn't the entire invested amount. As considered in question, for a loan amount of $100, the loss isn't the entire $100. Every borrower does repay some amount of the entire borrowed amount. This results in a reduction of the total loss incurred by LendingClub.

Upon detailed analyses of the data, it can be observed that a large chunk of the borrowed amount is repaid by the borrowers. To calculate the loss incurred by charged-off loans, we can find the total amount paid by the borrowers (total_pymnt) and subtract it from the total borrowed amount (funded_amnt). Then we can find the percentage of the same to understand the proportion of the recovered amount of the charged-off loans.

The results obtained cannot be called overly optimistic as it includes the average recovered amount from a dataset of 15,377 charged-off loans of the database. Considering the massive sample size and the process of calculating the amount, the data cannot be termed as overly optimistic.

The loss value we can consider is the difference between the total amount paid by the borrowers (total_pymnt) and the total borrowed amount (funded_amnt).

You should also consider the alternate option of investing in, say in bank CDs (certificate of deposit); let's assume that this provides an interest rate of 2%. Then, if you invest $100, you will receive $106 after 3 years (not considering reinvestments, etc), for a profit of $6. Considering a confusion matrix, we can then have profit/loss amounts with each cell, as follows:

		Predicted	
		FullyPaid	ChargedOff
Actual	FullyPaid	*profitValue*	$6
	ChargedOff	*lossValue*	$6

(a) Compare the performance of your models from Questions 6, 7 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.

Decision Tree:

```
> # Or, to set the predTrnCT values as factors, and then get the confusion matrix
> table(predictions=factor(predTrnCT, levels=c("Fully Paid", "Charged Off")), actuals=lcdfTrn$loan_status)
             actuals
predictions   Fully Paid Charged Off
  Fully Paid       48787         411
  Charged Off      22142       11118
> predProbTst=predict(lcDT1b,lcdfTst, type='prob')
> predProbTst=predict(lcDT1b,lcdfTst, type='prob')
> predTstCT = ifelse(predProbTst[, 'Charged Off'] > CTHRESH, 'Charged Off', 'Fully Paid')
> table(predTstCT , true=lcdfTst$loan_status)
             true
predTstCT     Fully Paid Charged Off
  Charged Off       9299        1810
  Fully Paid       14339        2038
> 
```

0.5

```
> # Or, to set the predTrnCT values as factors, and then get the confusion matrix
> table(predictions=factor(predTrnCT, levels=c("Fully Paid", "Charged Off")), actuals=lcdfTrn$loan_status)
             actuals
predictions   Fully Paid Charged Off
  Fully Paid       54382         979
  Charged Off      16547       10550
> predProbTst=predict(lcDT1b,lcdfTst, type='prob')
> predTstCT = ifelse(predProbTst[, 'Charged Off'] > CTHRESH, 'Charged Off', 'Fully Paid')
> table(predTstCT , true=lcdfTst$loan_status)
             true
predTstCT     Fully Paid Charged Off
  Charged Off       7527        1519
  Fully Paid       16111        2329
```

0.7

```
> table(predictions=factor(predTrnCT, levels=c("Fully Paid", "Charged Off")), actuals=lcdfTrn$loan_status)
             actuals
predictions   Fully Paid Charged Off
  Fully Paid       59330        2326
  Charged Off      11599        9203
> predProbTst=predict(lcDT1b,lcdfTst, type='prob')
> predTstCT = ifelse(predProbTst[, 'Charged Off'] > CTHRESH, 'Charged Off', 'Fully Paid')
> table(predTstCT , true=lcdfTst$loan_status)
             true
predTstCT     Fully Paid Charged Off
  Charged Off       5765        1193
  Fully Paid       17873        2655
```

Random Forest:

0.3

```
> table(pred = scoreTrn$predictions[, "Fully Paid"] > 0.3, actual=lcdfTrn$loan_status)
       actual
pred    Fully Paid Charged Off
  FALSE          0           9
  TRUE       70929       11520
> scoreTst <- predict(rfModel1,lcdfTst)
> scoreTst <- predict(rfModel1,lcdfTst)
> table(pred = scoreTst$predictions[, "Fully Paid"] > 0.3, actual=lcdfTst$loan_status)
       actual
pred    Fully Paid Charged Off
  FALSE          0           4
  TRUE       23638        3844
```

0.5

```
> table(pred = scoreTrn$predictions[, "Fully Paid"] > 0.5, actual=lcdfTrn$loan_status)
       actual
pred    Fully Paid Charged Off
  FALSE          0        4046
  TRUE       70929        7483
> scoreTst <- predict(rfModel1,lcdfTst)
> table(pred = scoreTst$predictions[, "Fully Paid"] > 0.5, actual=lcdfTst$loan_status)
       actual
pred    Fully Paid Charged Off
  FALSE          0        1357
  TRUE       23638        2491
```

0.7

```
> table(pred = scoreTrn$predictions[, "Fully Paid"] > 0.7, actual=lcdfTrn$loan_status)
       actual
pred    Fully Paid Charged Off
  FALSE       9934        1541
  TRUE       60995        9988
> scoreTst <- predict(rfModel1,lcdfTst)
> table(pred = scoreTst$predictions[, "Fully Paid"] > 0.7, actual=lcdfTst$loan_status)
       actual
pred    Fully Paid Charged Off
  FALSE        203        2961
  TRUE       23435         887
> |
```

From the above images we can see that the random forest model with 0.7 as threshold is performing better than the same model with different threshold and the decision tree models.

The Misclassification rate with threshold = 0.7 is

(887+203)/27500 = 0.0396 = 3.96%

(b) Another approach to determining the optimal threshold for implementing the model 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 analysis 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 total profits obtained from using a model is greater than investing in the safe CDs. The profit that we would have obtained from the safe CDs can be calculated by multiplying the rate of interest (5%) to the duration (3 years) with the total amount in hand.

If we consider the total amount to be $10000.
In the case of the CD, where the rate of interest per annum is 5% and duration is 3 years as in the case with our model.

The total profit through this method will be:
Profit earned from CD = 10000 * 3 * 0.05
= 1500

Based on our analysis, the Random Forest model is the best model with threshold = 0.7 and number of trees = 200 with the accuracy of ~94%.