Table of Contents

Problem Statement

LendingClub is an American peer-to-peer lending company, headquartered in San Francisco, California. It was the first peer-to-peer lender to register its offerings as securities with the Securities and Exchange Commission, and to offer loan trading on a secondary market.The platform allows anyone to be an investor.The main analysis is to find the risks involved in lending by finding the characteristics of default loans.

Who are the Targets ?

We try to focus on the investors who put their money into the Lending Club.The analysis might help the investors to decide a better investment and a good ROI.The project mainly focuses on the risks in lending a loan.The risk here we mean is the defaulters for the loan.There might be many factors that would be influencing the heavy loan defaults.Our aim to predict a model to that would help the investors decide on their investments.

Data Set

We have used an open source Dataset from Kaggle which contains all the complete information of loan data over the 10 years from 2007-2017.
https://www.kaggle.com/wendykan/lending-club-loan-data

```
#Inspecting the data for total rows and columns
df.info()
#We could see that it contains 74 columns and 887378 rows in it
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 887379 entries, 0 to 887378
Data columns (total 74 columns):
id                             887379 non-null int64
member_id                      887379 non-null int64
loan_amnt                      887379 non-null float64
funded_amnt                    887379 non-null float64
funded_amnt_inv                887379 non-null float64
term                           887379 non-null object
int_rate                       887379 non-null float64
installment                    887379 non-null float64
grade                          887379 non-null object
sub_grade                      887379 non-null object
emp_title                      835922 non-null object
emp_length                     887379 non-null object
home_ownership                 887379 non-null object
annual_inc                     887375 non-null float64
verification_status            887379 non-null object
issue_d                        887379 non-null object
loan_status                    887379 non-null object
pymnt_plan                     887379 non-null object
url                            887379 non-null object
desc                           126029 non-null object
purpose                        887379 non-null object
title                          887228 non-null object
zip_code                       887379 non-null object
addr_state                     887379 non-null object
dti                            887379 non-null float64
delinq_2yrs                    887350 non-null float64
earliest_cr_line               887350 non-null object
inq_last_6mths                 887350 non-null float64
mths_since_last_delinq         433067 non-null float64
mths_since_last_record         137053 non-null float64
open_acc                       887350 non-null float64
pub_rec                        887350 non-null float64
revol_bal                      887379 non-null float64
revol_util                     886877 non-null float64
total_acc                      887350 non-null float64
```

```
verification_status            887379 non-null object
issue_d                        887379 non-null object
loan_status                    887379 non-null object
pymnt_plan                     887379 non-null object
url                            887379 non-null object
purpose                        887379 non-null object
title                          887228 non-null object
zip_code                       887379 non-null object
addr_state                     887379 non-null object
dti                            887379 non-null float64
delinq_2yrs                    887350 non-null float64
earliest_cr_line               887350 non-null object
inq_last_6mths                 887350 non-null float64
mths_since_last_delinq         433067 non-null float64
open_acc                       887350 non-null float64
pub_rec                        887350 non-null float64
revol_bal                      887379 non-null float64
revol_util                     886877 non-null float64
total_acc                      887350 non-null float64
initial_list_status            887379 non-null object
out_prncp                      887379 non-null float64
out_prncp_inv                  887379 non-null float64
total_pymnt                    887379 non-null float64
total_pymnt_inv                887379 non-null float64
total_rec_prncp                887379 non-null float64
total_rec_int                  887379 non-null float64
total_rec_late_fee             887379 non-null float64
recoveries                     887379 non-null float64
collection_recovery_fee        887379 non-null float64
last_pymnt_d                   869720 non-null object
last_pymnt_amnt                887379 non-null float64
next_pymnt_d                   634408 non-null object
last_credit_pull_d             887326 non-null object
collections_12_mths_ex_med     887234 non-null float64
policy_code                    887379 non-null float64
application_type               887379 non-null object
acc_now_delinq                 887350 non-null float64
tot_coll_amt                   817103 non-null float64
tot_cur_bal                    817103 non-null float64
total_rev_hi_lim               817103 non-null float64
dtypes: float64(31), int64(2), object(21)
memory usage: 365.6+ MB
```

Cleaning Steps

Visually inspected the columns needed for analysis firstly,then did an Exploratory data analysis on it.Counted the frequency of the data columns needed for analysis to check for missing values and inconsistent values.That method helped me confirm the data consistency.Made Summary statistics on numeric columns which helped me find outliers if any.

Dealing with missing values

Filtered my data with the minimum missing values based on the length of my data.For instance,if I had 100000 data records,did filtration of data records which have only 80 % of null values as threshold thus getting rid of 20% null values for data analysis.This gave my data with very minimal null values getting rid of most obsolete columns for analysis.

Handling Outliers

I used the visual exploratory data analysis to check if there were any outliers.I used the visual exploratory data analysis to find outliers in the data.I used the **box plots** to help me find outliers if any.

Exploratory Data Analysis and Data Story

In this exploration of the lending club loan data we would try to answer two factors that would help the investors obtain comprehensible ideas of investing at Lending club. We shall focus on the below major aspects

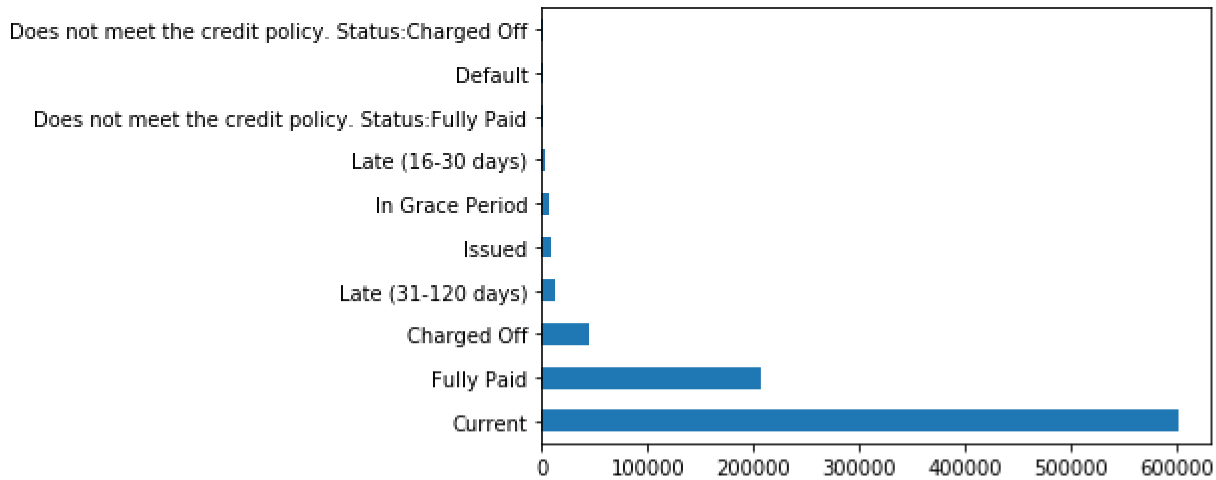
1.Risks in lending

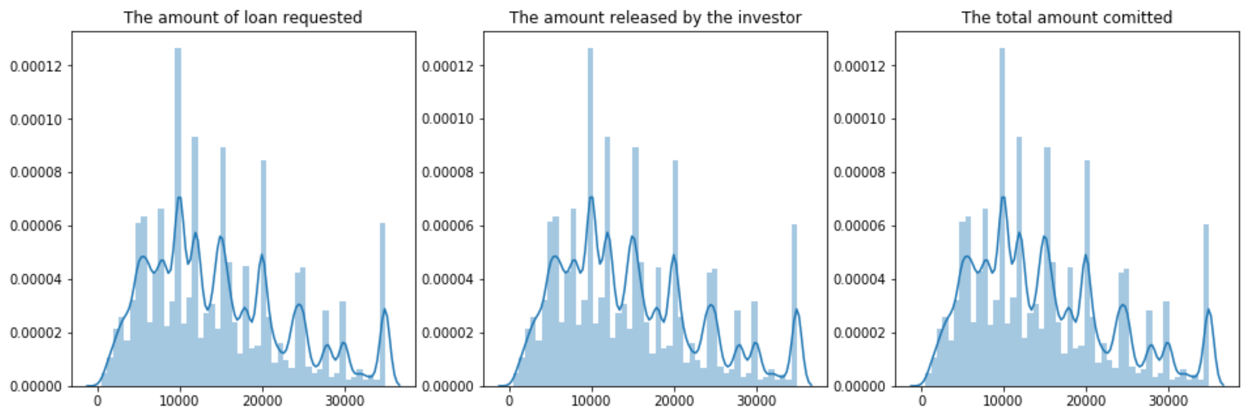
a)How does the interest rate affect the repayment?

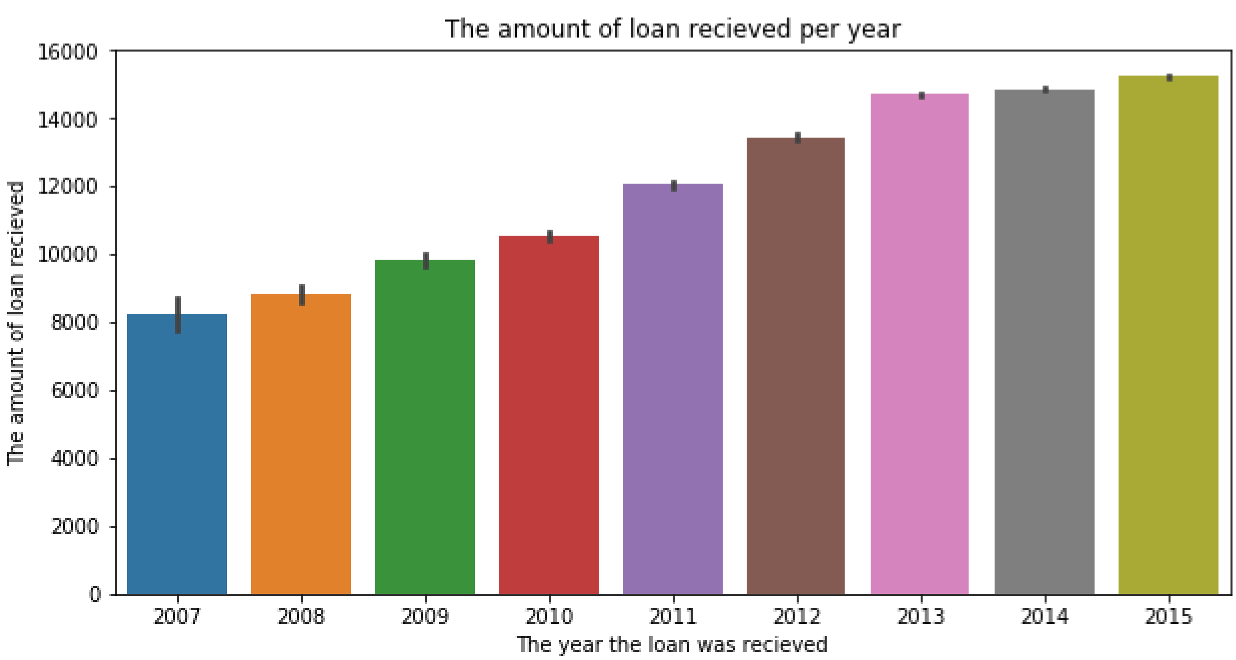
b)What are the states which have heavy defaults?

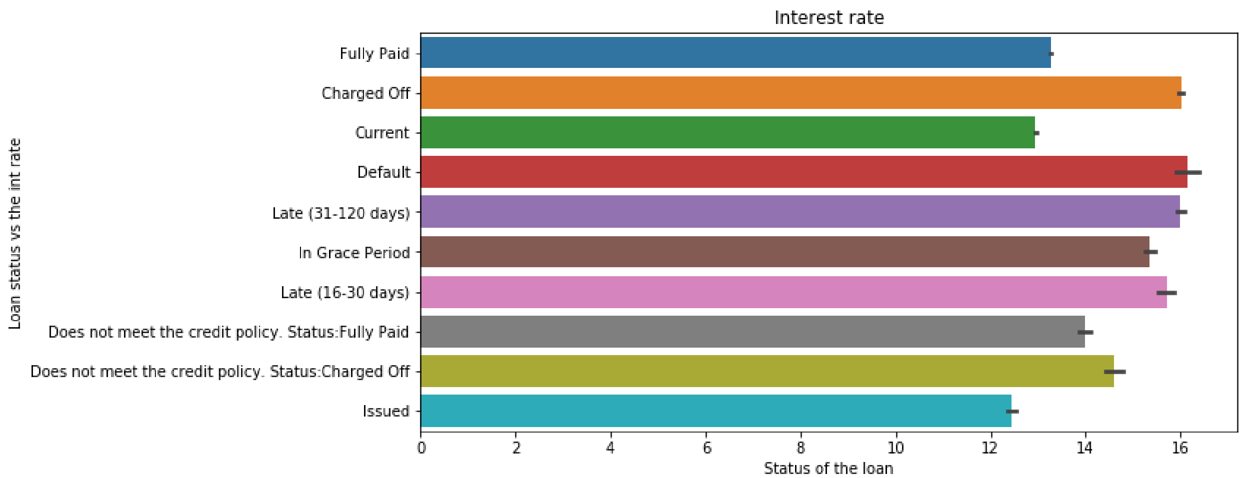
c)Was there any particular year that had a great downfall in repayment ?

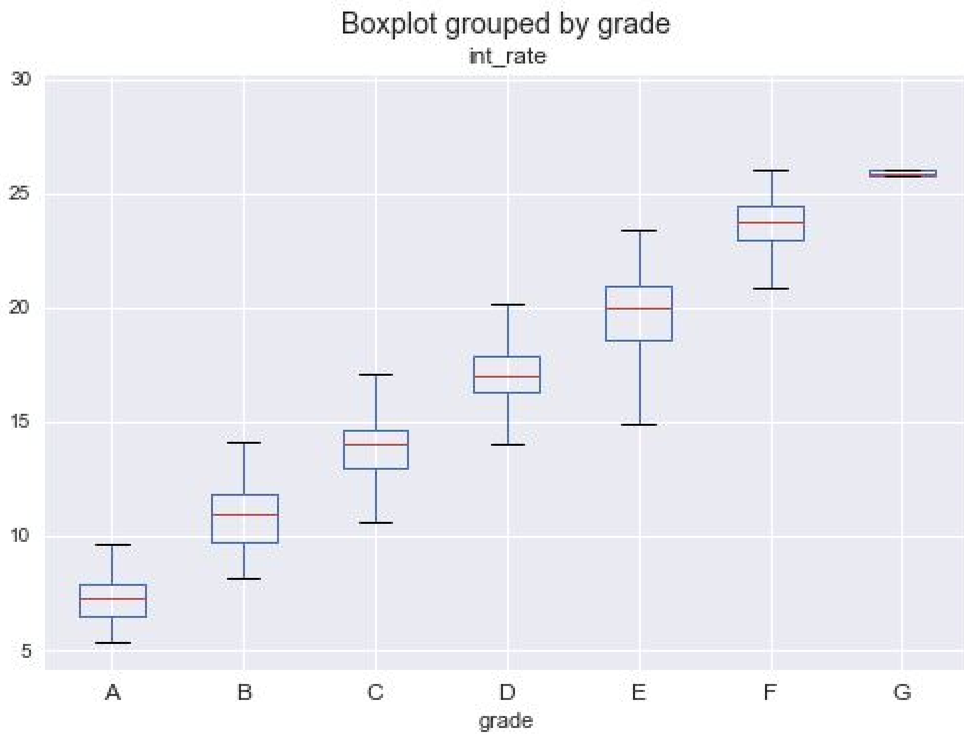
d)Does the employer grade have any impact on the interest rate ?

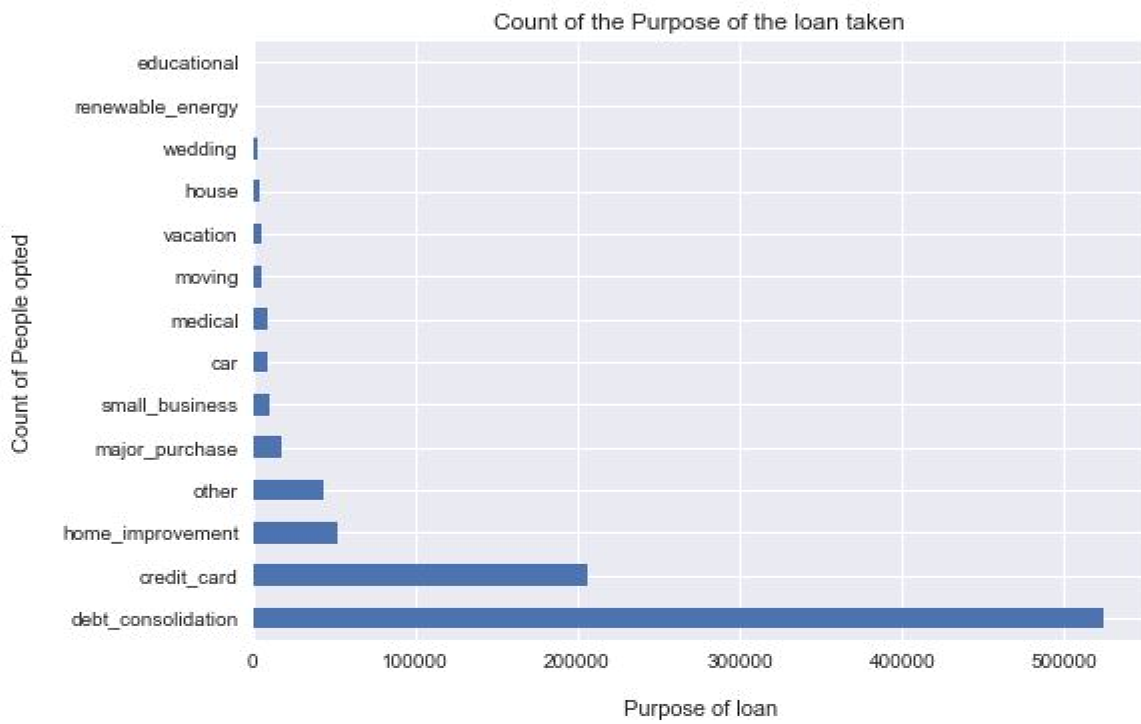

Does not meet the credit policy. Status:Charged Off
Default
Does not meet the credit policy. Status:Fully Paid
Late (16-30 days)
In Grace Period
Issued
Late (31-120 days)
Charged Off
Fully Paid
Current
0
100000
200000
300000
400000
500000
600000

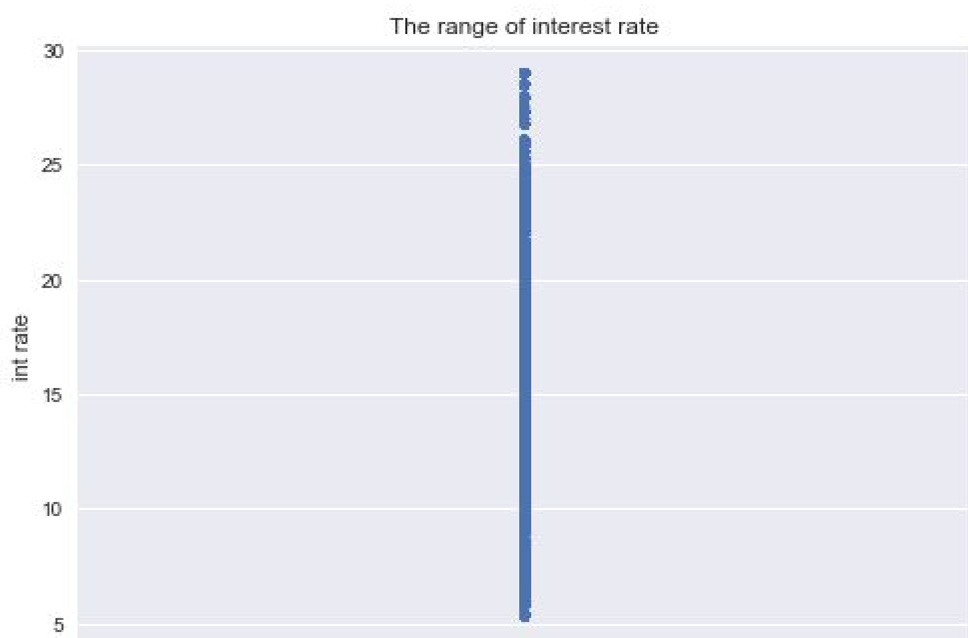

The amount of loan requested
The amount released by the investor
The total amount comitted
0.00012
0.00010
0.00008
0.00006
0.00004
0.00002
0.00000
0
10000
20000
30000


The amount of loan recieved per year
The amount of loan recieved
16000
14000
12000
10000
8000
6000
4000
2000
0
2007
2008
2009
2010
2011
2012
2013
2014
2015
The year the loan was recieved


Interest rate
Fully Paid
Charged Off
Current
Default
Late (31-120 days)
In Grace Period
Late (16-30 days)
Does not meet the credit policy. Status:Fully Paid
Does not meet the credit policy. Status:Charged Off
Issued
Loan status vs the int rate
0
2
4
6
8
10
12
14
16
Status of the loan


Boxplot grouped by grade
int_rate
30
25
20
15
10
5
A
B
C
D
E
F
G
grade





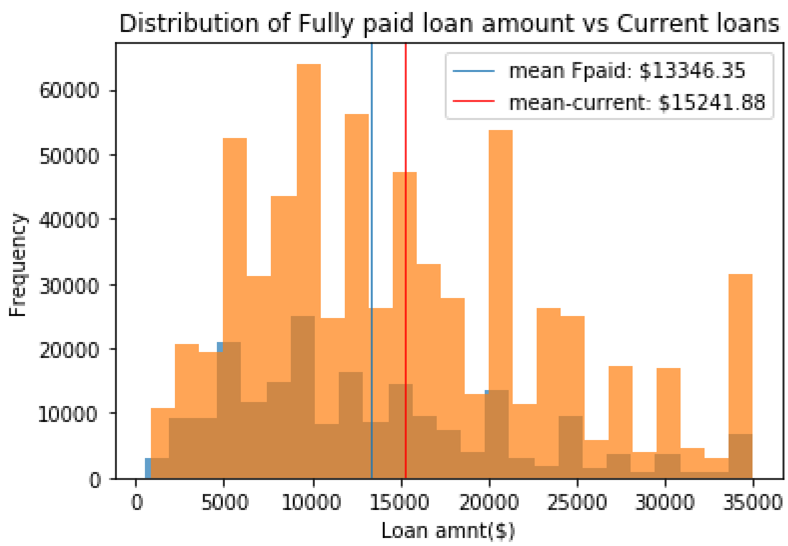
Statistical Data Analysis

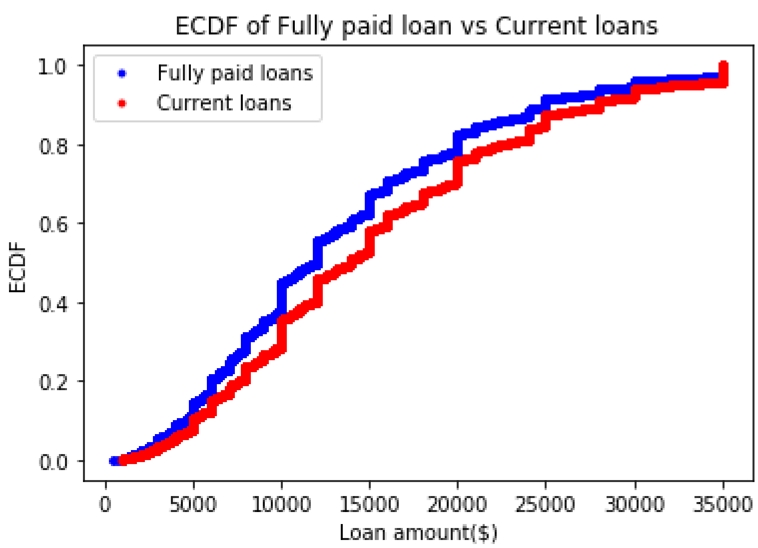
Let's try to find out the most key factors on deciding the loan defaulters.For that,we could frame a null hypothesis and an alternate hypothesis and strength our analysis with the mathematical results.

Null Hypothesis : The mean loan amount with a default loan status is the same as that of the non default loan status

Alternate Hypothesis : The mean loan amount with a default loan status is not the same as that of the non default loan status

We perform the t test to find the p value and see if we accept or reject the null hypothesis.





```
st.ttest_ind(fullypaid_stats,current_stats,equal_var=True)
```

```
Ttest_indResult(statistic=-88.37767852521944, pvalue=0.0)
```

We end up getting a very low p value and hence neglect the null hypothesis.
Thus we could infer that both the groups are different and the loan amount plays an imporatant feature in the repayment process

5.Machine Learning

5.1.One hot encoding

Since we have some categorical variables for the analysis and the machne learning algorithms doesn't take categorical and string variables directly, we have to creat dummy variables for them. We can either encode them using label encoder available for python, but it would be wrong in our analysis since a lot of these variables have multiple categories. Just using weights can cause discrepencies in the algorithm. Instead, we will one hot encode these so that we have a 1 wherever that category turns up and 0 otherwise. This will also create seperate columns for each level of category. Also, we'll be dropping one of the categories so that we have N-1 columns instead of N.

5.2.Model Selection

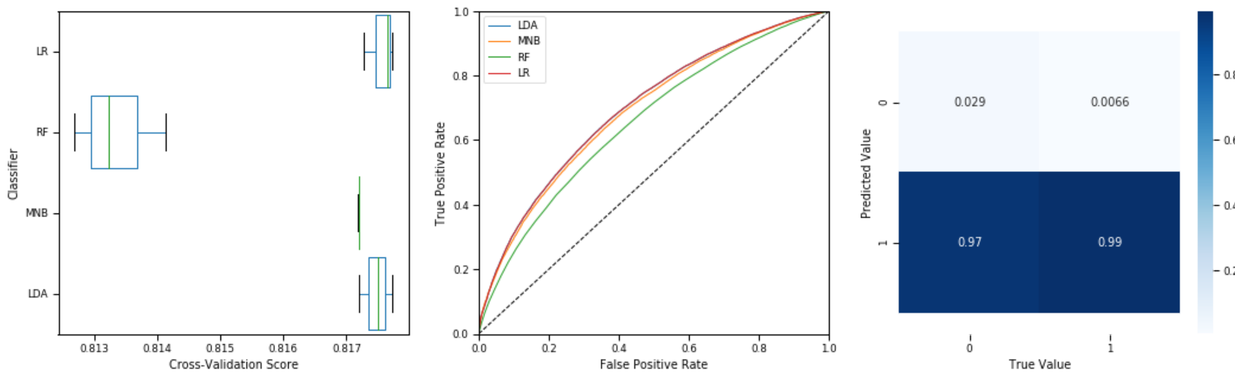
We are now ready to build some models. The following would be our approach for building and selecting the best model:

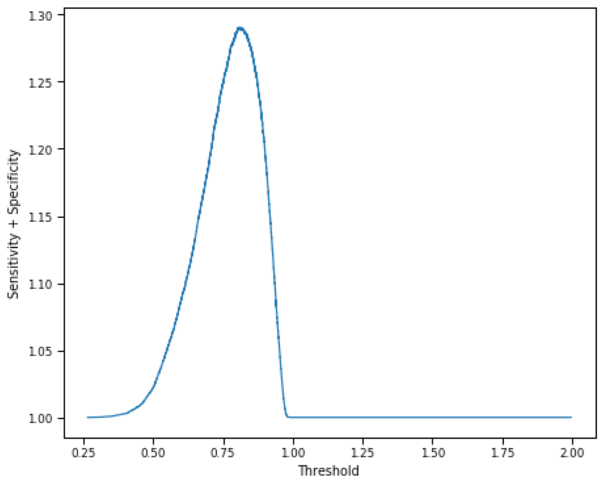
1. Build a model on the imbalance dataset we got from data cleaning.

2. Balance the dataset by using equal amount of default and 'fully paid' loans.

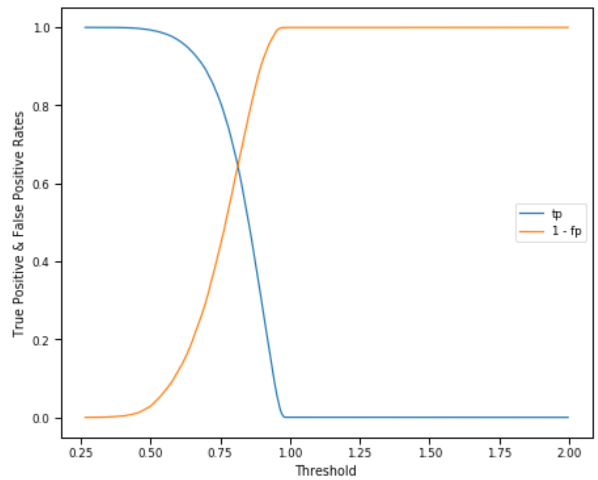
Let's try some models on the train dataset With 3 fold cross validation. We are going to use the following 4 machine learning algorithms:

1. Linear Discriminant Analysis
2. Multinomial Naive Bayes
3. Random Forest (tree based model)
4. Logistic Regression

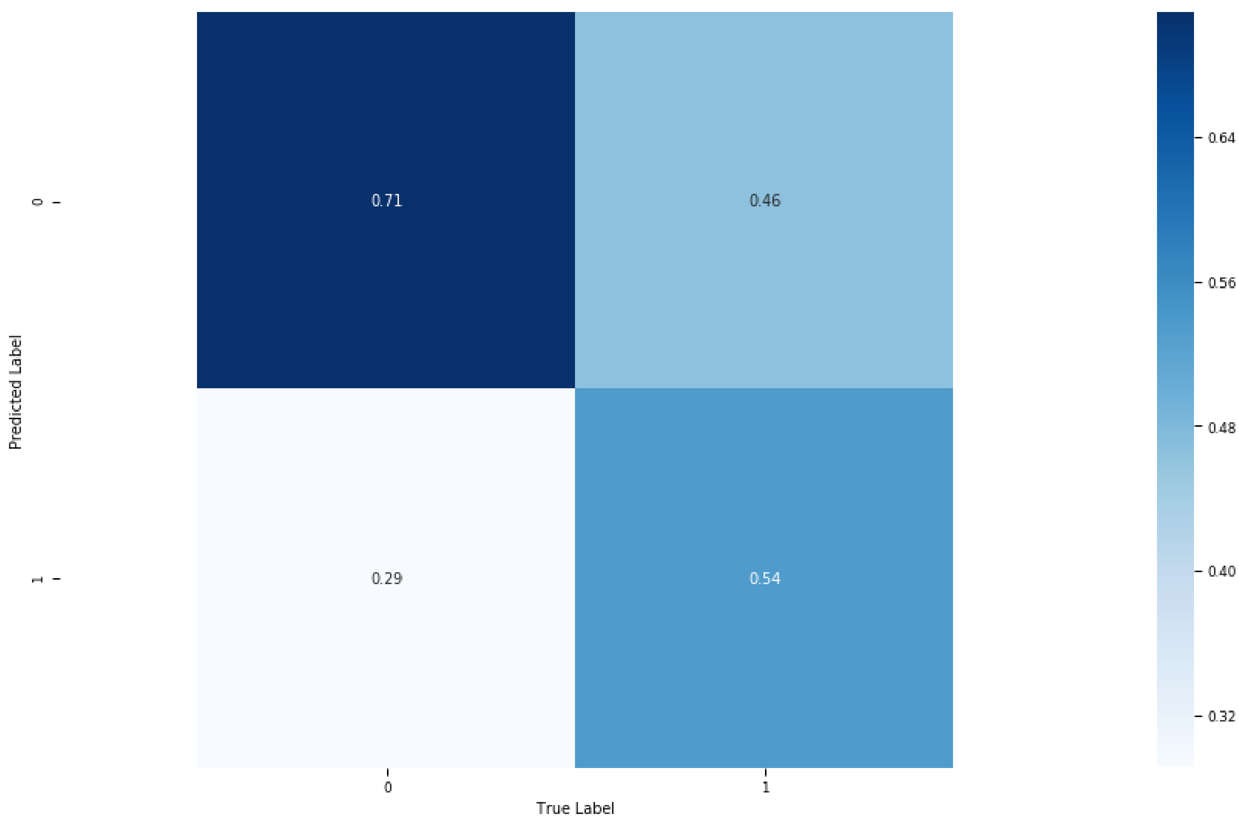




Sensitivity + Specificity
Threshold



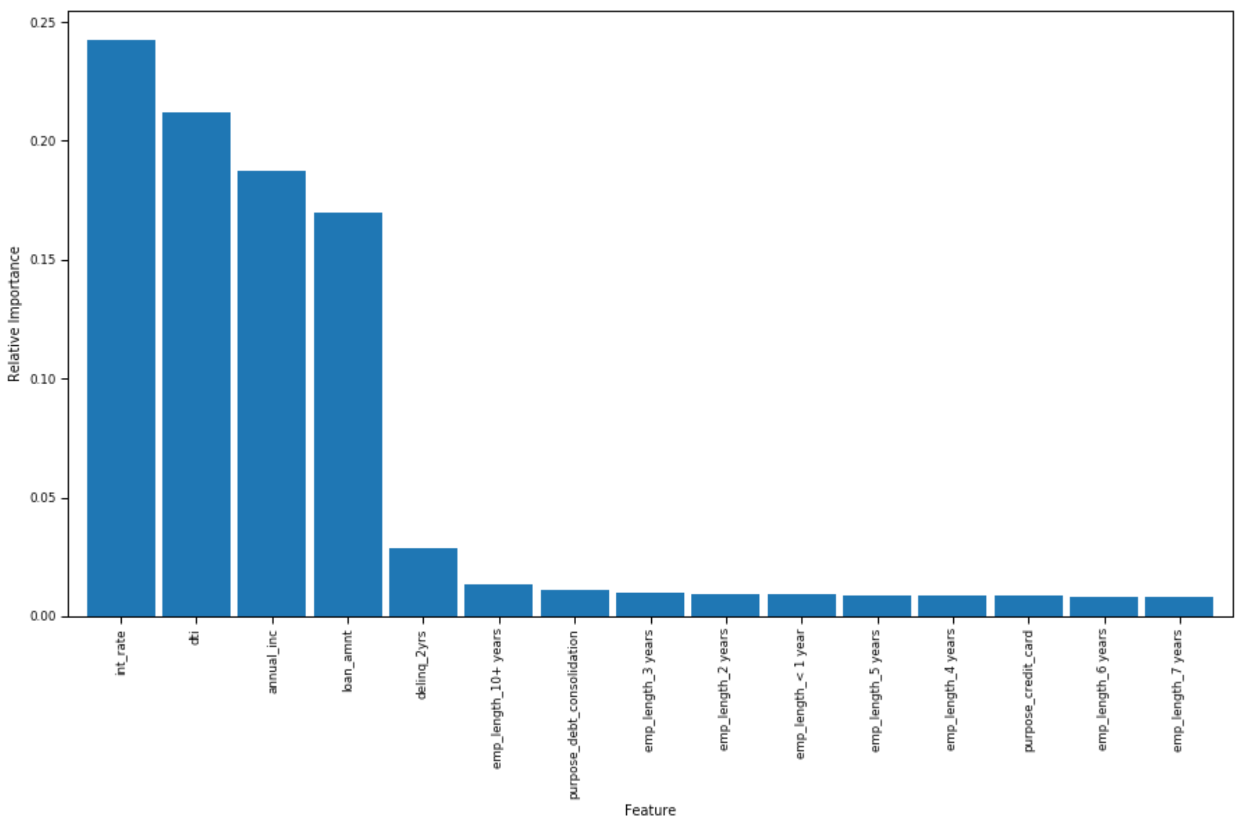
True Positive & False Positive Rates
Threshold
tp
1 - fp



Predicted Label
True Label
0.71
0.46
0.29
0.54

The optimum threshold for the classifier have increased out models prediction power of Default (0). Even now the model doesn't provide a lot of prediction power and we have to train the model again using a different algorithm with some tweaks.

We use the variable importance to see what are the most important variables that are used.



6.Conclusion

We have successfully built an machine learning algorithm to predict the people who might default on their loans. This can be further used by LendingClub for their analysis. Also, we might want to look on other techniques or variables to improve the prediction power of the algorthm. One of the drawbacks is just the limited number of people who defaulted on their loan in the 8 years of data (2007-2015) present on the dataset. We can use an updated dataframe which consist next 3 years values (2015-2018) and see how many of the current loans were paid off or defaulted or even charged off. Then these new data points can be used for predicting them or even used to train the model again to improve its accuracy.