Loan Defaulter Prediction Analysis

Technical Report

List of Abbreviations

EDA	Exploratory Data Analysis
ML	Machine Learning
NIR	No Information Rate
SVM	Support Vector Machines
TTC	Time To Complete
URL	Universal Resource Locator
xGB	eXtreme Gradient Boosting

Table of Contents

Abstract

This project is dedicated to the LendingClub Loan Data. LendingClub is a peer-to-peer lending start-up based in San Francisco, California. It was the first peer-to-peer lender to register its offers as securities with the Securities and Exchange Commission (SEC) and to provide loan trading on a secondary market. LendingClub is the world's largest peer-to-peer lending marketplace. Borrowers may use LendingClub to make unsecured personal loans ranging from $1,000 to 40,000 dollars. The normal loan term is three years. Investors may search and browse loan listings on the LendingClub website and choose loans to invest in based on information provided about the borrower, loan size, loan grade, and loan purpose. Interest is how investors gain money. LendingClub earns money by charging borrowers an interest rate.

Section I: Introduction

For the objective of this project, we have built multiple machine learning models with the goal of accurately determining whether a person, given a set of attributes, has a high risk of defaulting on a loan and what factors play a role in determining this likelihood.

In this process, however, we have not only focused on improving accuracy but also on interpreting the results of these predictive models and EDA.

As such, we started by listing down a few business questions regarding loan defaulters that the given data might be able to answer. For example, does variation in loan interest have an impact on loan charge off? Does variation in loan interest have an impact on loan charge off? (See Appendix-2 for the full set of preliminary questions).

Next, we cleaned and munged the given dataset, and followed up with a correlation analysis of the variables. We then experimented with four types of machine learning algorithms (SVM, Gradient Boosting, Naïve Bayes, Random Forest) to get an initial idea of which variables might be important in determining the cancellation of bookings. In parallel, we also performed EDA (Exploratory Data Analysis) to discover patterns, spot anomalies, test hypotheses and check assumptions with the help of statistical and graphical methods. Then, we delved deeper into some of the analyses based on the results of preliminary EDA.

For the models, we compared the performances of our four models, and determined that Random Forest (ranger) is the best one for this case. We then worked towards fine-tuning that ML model.

We provided interpretations with the goal of providing valuable insights to detect 100% of defaulter transactions while reducing inaccuracies in prediction and identifying the factors that influence loan borrowers from defaulting using association rule mining, ML models, and EDA.

Section II: Dataset, Variables and Assumptions

What dataset are we using?

```
        **
        #Read and load data into loanData
        loanData = pd.read_csv("/content/lending_club_loans.csv")

        display(loanData)
        #42542 rows and 115 columns
        #shows 42,542 rows and 115 columns
        **
```

The dataset we are utilizing comprises comprehensive loan data for all loans granted between 2007 and 2011, including the current loan status (Current, Late, etc.) and the most recent payment information along with performance data. Among the additional features are credit ratings, the amount of financial inquiries, addresses with zip codes and states, and collections. There are 115 columns containing information about individual loan accounts and 42538 rows, with each row divided by an individual loan id and a member id; The variable we are predicting is the 'Charged Off' category from the column 'loan status', which indicates the loan status of the loan borrower. In other words, we predicted this dependent variable using the remaining data columns as independent variables.

Which variables did we use?

Initially, we used all the variables to complete our Exploratory Data Analysis (EDA) (see Section III– Step 4 for the full description of our EDA). Appendix-1: Metadata contains the complete description of the most important features (columns) used.

Section III: The Process

This section explains the procedures we performed during EDA and in constructing the final ML model.

Step 1: Preprocessing

i. **Check for NA and the missing values**

We began by downloading the dataset and named it ‘LoanData’ as it is a orginal data. The first thing we did was see somewhere there was NA anywhere in the dataset.

```
            **
print(" \n Description of each column with missing values \n",
loanData.isnull().sum()) #Total number of missing values NaN at each column
in a DataFrame
```

```
print(" \n Total number of missing values NaN in the DataFrame : \n\n",
loanData.isnull().sum().sum()) #Total number of columns with missing values
with axis = 0 for column wise operation

print(" \n Total number of columns with missing values : \n\n",
loanData.isnull().any(axis=0).sum())

#Total number of records with missing values axis = 1 for row wise operation

print(" \n Total number of records with missing values : \n\n",
loanData.isnull().any(axis=1).sum() )
           **
```

As a result, we found that the dataset contains some empty or undefined values. Now we individually imputed null values for each columns.

ii. Change categorical variables into factors

```
loanData.grade = loanData.grade.astype(str) #makes the categorical codes (datatype
characters) as string
           **
```

For example, this code turned the column ‘LoanGrade’ into factor type from character/categorical type to string.

iii. Redefine Column Names

```
           **
loanData.rename(columns={'total_rec_prncp': 'total_principal_recieved'}, inplace=True)

           **
```

When we need to rename certain selected columns, we may use this approach since we just need to supply information for the columns that need to be renamed.

iv. Parsing Dates

Python provides a built-in mechanism for parsing dates called strptime. We use strptime to set a default datetime values for all variables named ‘issue_date’.

```
           **
from datetime import datetime year=[] date_time_converted=[] for i in
range(len(loanData['issue_date'])): date_time_str = loanData['issue_date'][i]
onlyYear = date_time_str[0:3]+'-'+'20'+date_time_str[-2:]
```

```
pd.to_datetime(onlyYear) year.append(onlyYear)
loanData['Issue_Date_Datetime'] = year print(loanData['Issue_Date_Datetime'])
            **
```

Here, we change the datetime format from ‘Dec-12’ to Dec-2012’. It is now convenient to perform time series and trends analysis over certain time.

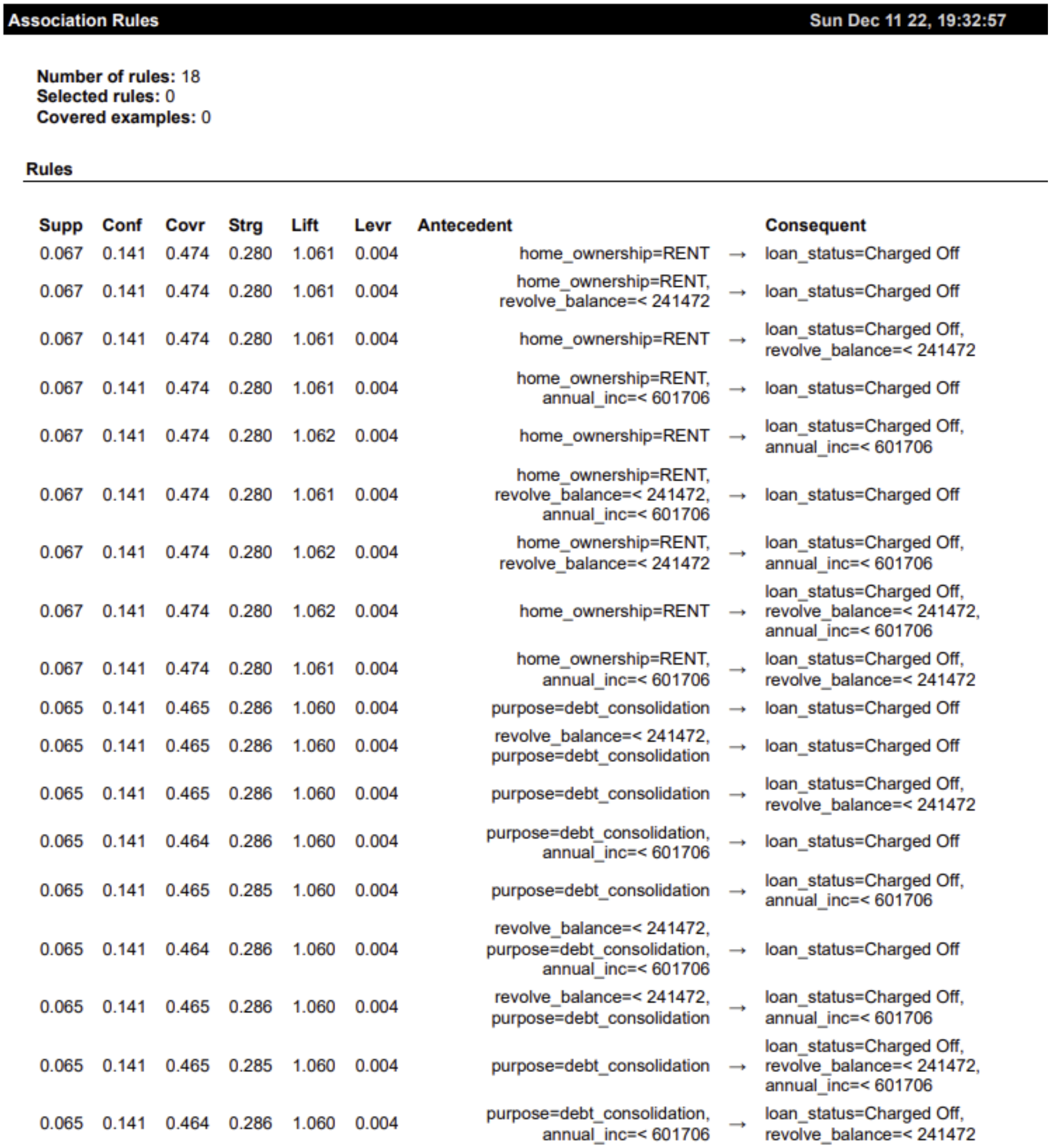
Step 3: Experimenting and selecting the Machine Learning (ML) algorithm

Note: This step was done in parallel to *Step 4: Exploratory Data Analysis*. Both of these steps provided inputs to each other. For example, we pursued further EDA on variables determined as important by ML models. On the other hand, we used these models to check whether the variables deemed important through EDA are also ranked as important

variables by different ML methods. In other words, we used these ML models to see if they concur with our intuition-based EDA questions.

Phase I: Association rule mining

In this section we used the Association rules mining to understand which features have impact on defaulted borrowers whose loan status is “Charged Off”. We have identified Top 18 association rules. It is observed that people were found to provide collateral home ownership as rent, revolving balance less than 241k and their major purpose of loan being debt consolidation were the major factors in influencing loan defaulting.



Association Rules Sun Dec 11 22, 19:32:57

Number of rules: 18
Selected rules: 0
Covered examples: 0

Rules

Supp	Conf	Covr	Strg	Lift	Levr	Antecedent		Consequent
0.067	0.141	0.474	0.280	1.061	0.004	home_ownership=RENT	→	loan_status=Charged Off
0.067	0.141	0.474	0.280	1.061	0.004	home_ownership=RENT, revolve_balance=< 241472	→	loan_status=Charged Off
0.067	0.141	0.474	0.280	1.061	0.004	home_ownership=RENT	→	loan_status=Charged Off, revolve_balance=< 241472
0.067	0.141	0.474	0.280	1.061	0.004	home_ownership=RENT, annual_inc=< 601706	→	loan_status=Charged Off
0.067	0.141	0.474	0.280	1.062	0.004	home_ownership=RENT	→	loan_status=Charged Off, annual_inc=< 601706
0.067	0.141	0.474	0.280	1.061	0.004	home_ownership=RENT, revolve_balance=< 241472, annual_inc=< 601706	→	loan_status=Charged Off
0.067	0.141	0.474	0.280	1.062	0.004	home_ownership=RENT, revolve_balance=< 241472	→	loan_status=Charged Off, annual_inc=< 601706
0.067	0.141	0.474	0.280	1.062	0.004	home_ownership=RENT	→	loan_status=Charged Off, revolve_balance=< 241472, annual_inc=< 601706
0.067	0.141	0.474	0.280	1.061	0.004	home_ownership=RENT, annual_inc=< 601706	→	loan_status=Charged Off, revolve_balance=< 241472
0.065	0.141	0.465	0.286	1.060	0.004	purpose=debt_consolidation	→	loan_status=Charged Off
0.065	0.141	0.465	0.286	1.060	0.004	revolve_balance=< 241472, purpose=debt_consolidation	→	loan_status=Charged Off
0.065	0.141	0.465	0.286	1.060	0.004	purpose=debt_consolidation	→	loan_status=Charged Off, revolve_balance=< 241472
0.065	0.141	0.464	0.286	1.060	0.004	purpose=debt_consolidation, annual_inc=< 601706	→	loan_status=Charged Off
0.065	0.141	0.465	0.285	1.060	0.004	purpose=debt_consolidation	→	loan_status=Charged Off, annual_inc=< 601706
0.065	0.141	0.464	0.286	1.060	0.004	revolve_balance=< 241472, purpose=debt_consolidation, annual_inc=< 601706	→	loan_status=Charged Off
0.065	0.141	0.465	0.286	1.060	0.004	revolve_balance=< 241472, purpose=debt_consolidation	→	loan_status=Charged Off, annual_inc=< 601706
0.065	0.141	0.465	0.285	1.060	0.004	purpose=debt_consolidation	→	loan_status=Charged Off, revolve_balance=< 241472, annual_inc=< 601706
0.065	0.141	0.464	0.286	1.060	0.004	purpose=debt_consolidation, annual_inc=< 601706	→	loan_status=Charged Off, revolve_balance=< 241472

Interpretation

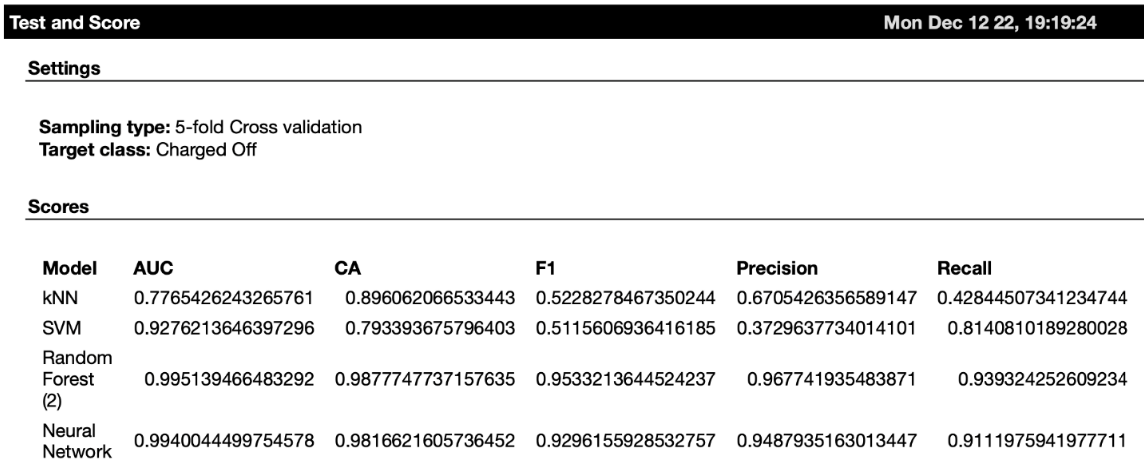
This graph shows the distributions of loan interest rate records in the dataset. The average interest rate.

Phase I: Experimenting with different ML algorithms

The project involved testing multiple machine learning methods to streamline the evaluation process and simplify the amount of code needed. Four methods were used to test and compare their performance.

All models used the same training control method of repeated k-fold cross validation, with 5 (k) separations and 3 repeats. Below table is model results:

**
Model Results



Test and Score Mon Dec 12 22, 19:19:24

Settings

Sampling type: 5-fold Cross validation
Target class: Charged Off

Scores

Model	AUC	CA	F1	Precision	Recall
kNN	0.7765426243265761	0.896062066533443	0.5228278467350244	0.6705426356589147	0.42844507341234744
SVM	0.9276213646397296	0.793393675796403	0.5115606936416185	0.3729637734014101	0.8140810189280028
Random Forest (2)	0.995139466483292	0.9877747737157635	0.9533213644524237	0.967741935483871	0.939324252609234
Neural Network	0.9940044499754578	0.9816621605736452	0.9296155928532757	0.9487935163013447	0.9111975941977711

**

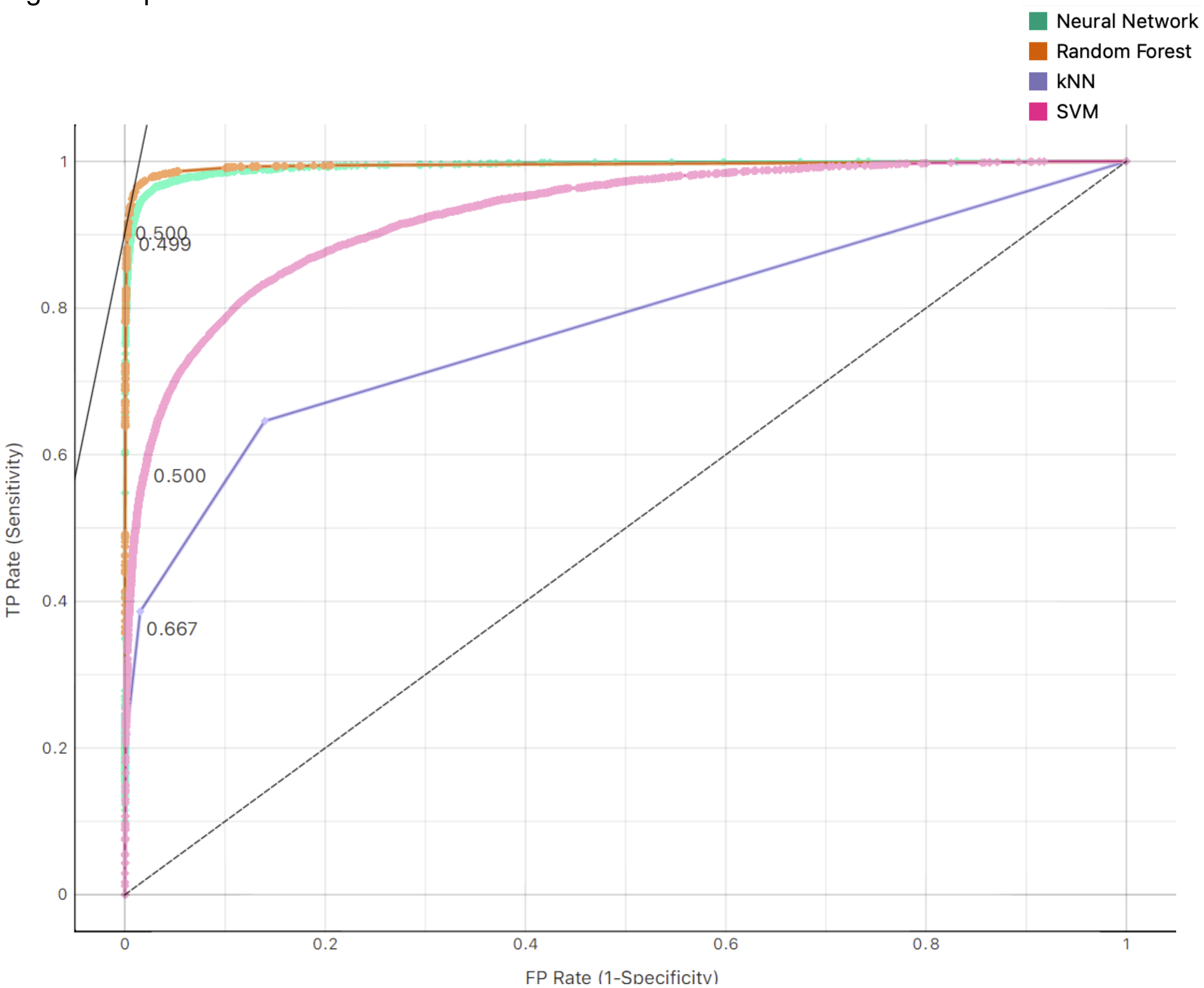
The four ML models that we experimented with at this stage are:

1. KNN
2. Support Vector Machines with Linear Kernel
3. Random Forest
4. Neural Network

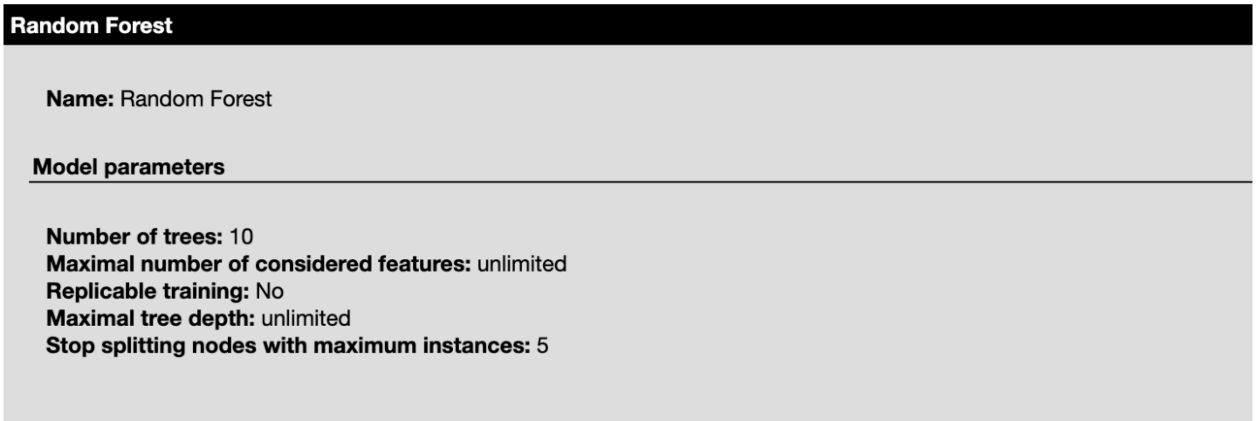
We selected the targeted column as loan status and discretized the variables fed into the model. Then we set the targeted variables as ‘Charged-Off’ for all the four models.

Phase II: Selecting the final model and fine-tuning it

The ROC curve analysis is performed as shown below for comparing and selecting the final model. The Random Forest ROC curve indicated by orange color is closer to True positive and higher compared to other models.



From the above Random Forest is a type of ensemble learning method, which means it combines the predictions of multiple models to improve the overall accuracy. We have used this algorithm to create a large number of decision trees, each of which is trained on a random subset of the data. The final prediction is made by averaging the predictions of all the trees. Random forest is a versatile and widely-used algorithm, and is often used for classification and regression tasks. Among other models, Random Trees gives the best prediction compare to other models.



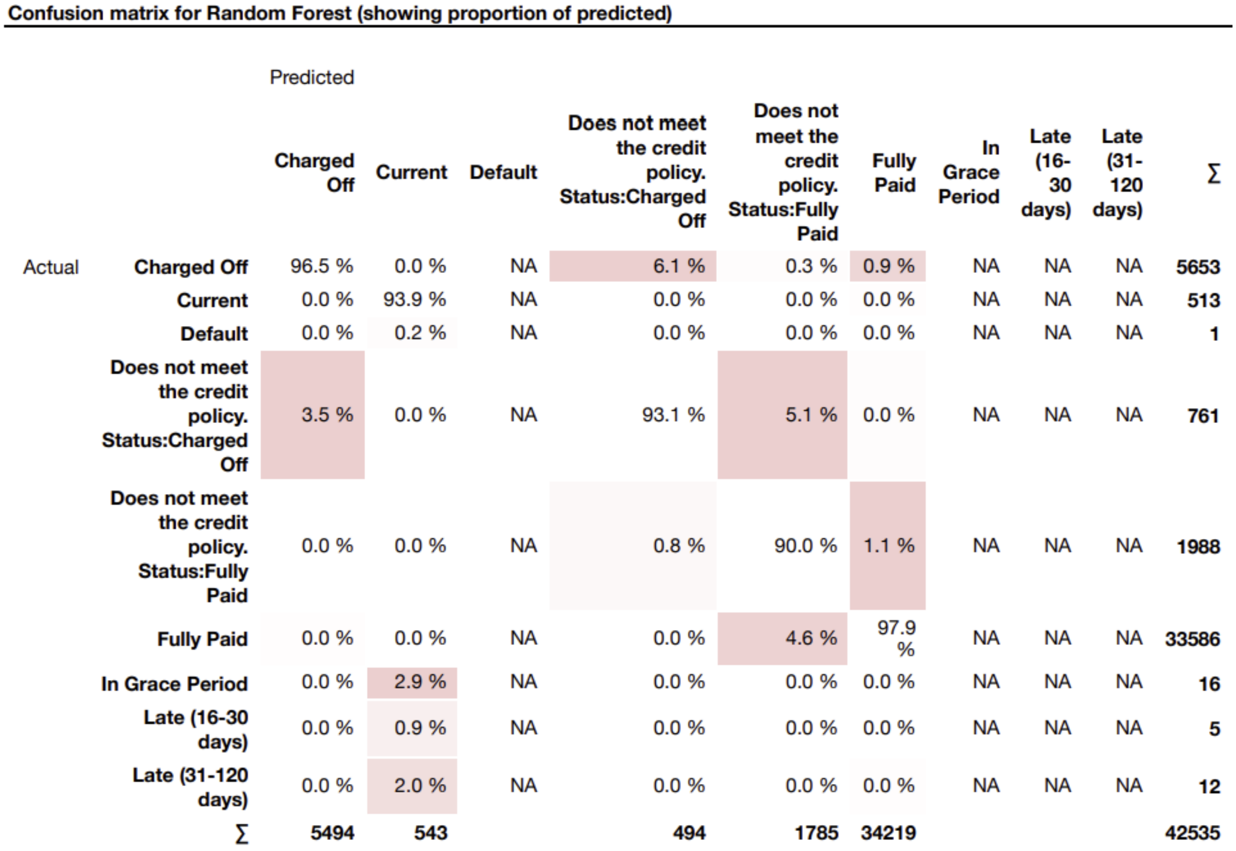
Random Forest

Name: Random Forest

Model parameters

Number of trees: 10
Maximal number of considered features: unlimited
Replicable training: No
Maximal tree depth: unlimited
Stop splitting nodes with maximum instances: 5

Above image is the model parameter to perform the random forest that is fed into training the model. Here, we have split the tress into smaller parts, not more than 5 which results in 95% accuracy. Thus, the best score was achieved by Random Forest after the balanced data and the model is trained in appropriate fine tuning.



Confusion matrix for Random Forest (showing proportion of predicted)

		Predicted									
		Charged Off	**Current**	**Default**	**Does not meet the credit policy. Status:Charged Off**	**Does not meet the credit policy. Status:Fully Paid**	**Fully Paid**	**In Grace Period**	**Late (16-30 days)**	**Late (31-120 days)**	**Σ**
Actual	**Charged Off**	96.5 %	0.0 %	NA	6.1 %	0.3 %	0.9 %	NA	NA	NA	**5653**
	Current	0.0 %	93.9 %	NA	0.0 %	0.0 %	0.0 %	NA	NA	NA	**513**
	Default	0.0 %	0.2 %	NA	0.0 %	0.0 %	0.0 %	NA	NA	NA	**1**
	Does not meet the credit policy. Status:Charged Off	3.5 %	0.0 %	NA	93.1 %	5.1 %	0.0 %	NA	NA	NA	**761**
	Does not meet the credit policy. Status:Fully Paid	0.0 %	0.0 %	NA	0.8 %	90.0 %	1.1 %	NA	NA	NA	**1988**
	Fully Paid	0.0 %	0.0 %	NA	0.0 %	4.6 %	97.9 %	NA	NA	NA	**33586**
	In Grace Period	0.0 %	2.9 %	NA	0.0 %	0.0 %	0.0 %	NA	NA	NA	**16**
	Late (16-30 days)	0.0 %	0.9 %	NA	0.0 %	0.0 %	0.0 %	NA	NA	NA	**5**
	Late (31-120 days)	0.0 %	2.0 %	NA	0.0 %	0.0 %	0.0 %	NA	NA	NA	**12**
	Σ	**5494**	**543**		**494**	**1785**	**34219**				**42535**

From the confusion matrix we understand that the Random Forest has predicted 96.5% of actual information of loan defaulter records (Charged Off status) correctly.

Step 4: Exploratory Data Analysis

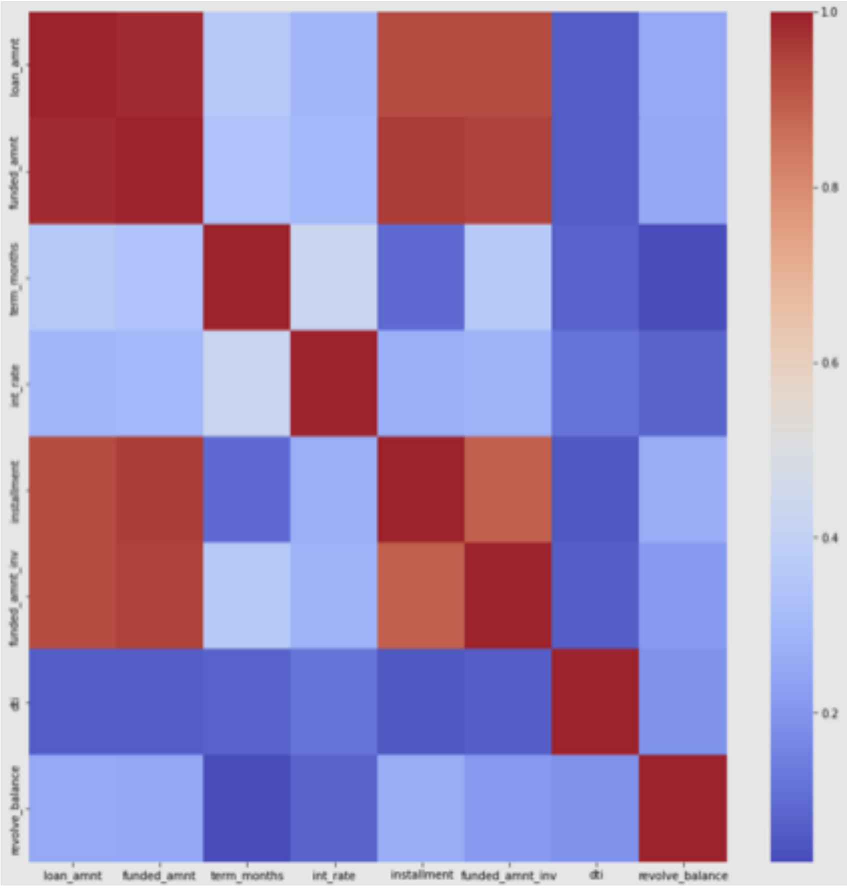
Exploratory data analysis, or EDA, is an important part of understanding the LendingClub dataset. It involves examining the relationships between different variables in the data and gaining intuition about how to interpret the results of subsequent analysis. Before moving on to more complex modeling techniques, it is important to understand the distribution of the data and to ask specific questions about the information contained in the dataset.

Section I: On the numeric variables, we conducted the following dataset-wide correlation test. The correlation matrix depicts the correlation between all the possible pairs of values in a table.

```
**
#calculate the correlation
corr = corrLoanData.corr()

# Plot the heatmap
plt.figure(figsize=(14,14))
sns.heatmap(corr,
        xticklabels=corr.columns,
        yticklabels=corr.columns,
        cmap='coolwarm')
**

**Output**
```



Results

1. FicoRangeHigh and FicoRangeLow have high correlation (positive)
2. FicoRangeHigh and IntRate have high correlation (negative)
3. FundedAmount and LoanAmount have high correlation (positive)
4. Delinq_2yrs and mthns_since_last_delinq have almost no correlation
5. LoanAmount has slightly positive correlation with FundedAmount, Installment, FundedAmountInv, and TotalPayment
6. InterestRate has slightly negative correlation with FicoRangeHigh, FicoRangeLow, and TotalInterestRecieved

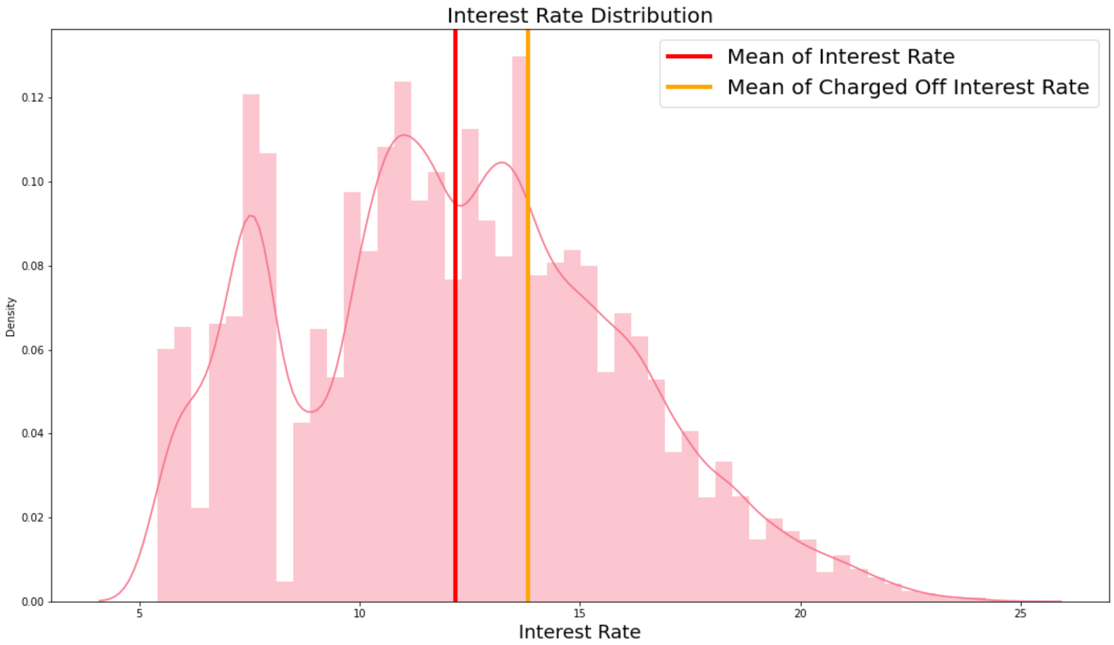
Interpretation

- Through this very preliminary correlation comparison, it seems that LoanAmount, FundedAmount, LoanStatus, Purpose, LoanInterest and DTI are important variables in determining ChargedOff variable. So, to start with, we performed further EDA on these variables

Q. Does variation in loan interest have an impact on loan charge off?

```
**Code
# Distribution of interest rates
sns.set_palette("husl")
f=plt.figure(figsize=(18,10))
sns.distplot(loanData['int_rate'], hist='density')
plt.axvline(x=loanData.int_rate.mean(), color='red', linestyle='-',
lw=4, label='Mean of Interest Rate')
plt.axvline(x=loanDataChargedOff.int_rate.mean(), color='orange',
linestyle='-', lw=4, label='Mean of Charged Off Interest Rate')
plt.title('Interest Rate Distribution', fontsize=20)
plt.xlabel('Interest Rate', fontsize=18)
plt.legend(fontsize=20)
plt.show()
**
```

Output

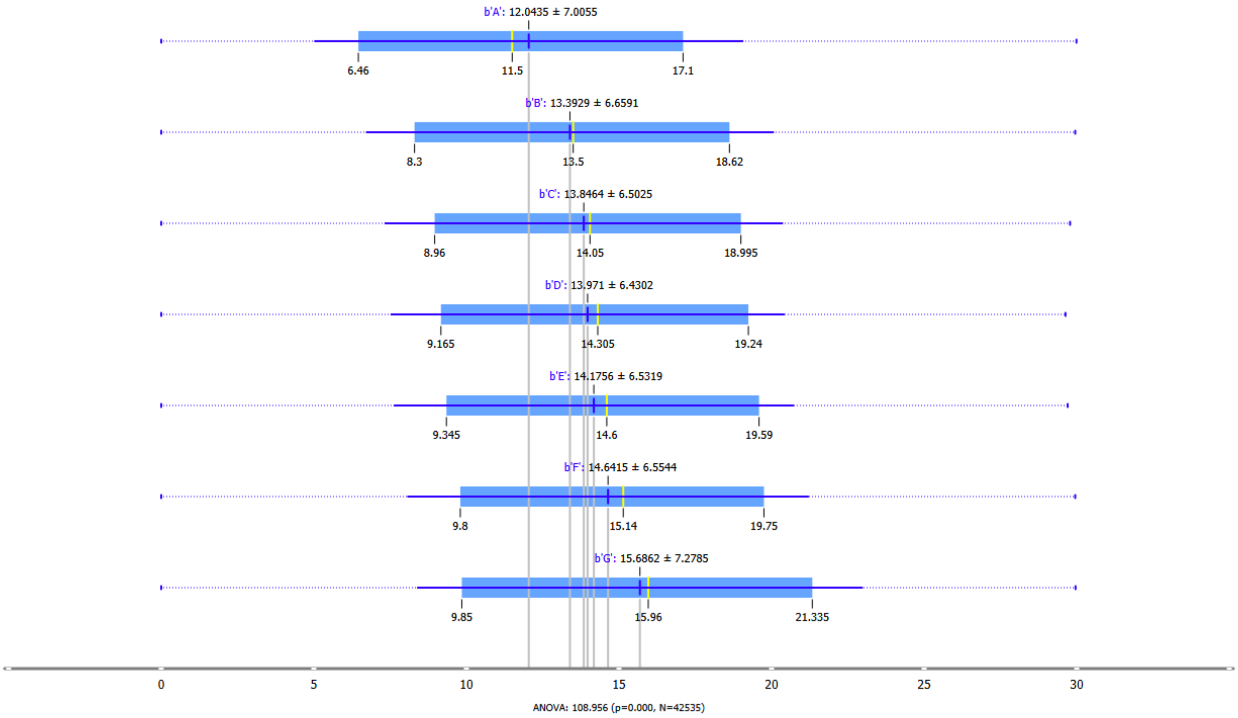


Interpretation

This graph shows the distributions of loan interest rate records in the dataset. The average interest rate for all loans is 12.16% is represented by the solid red line. The solid orange line represents the average interest rate is 13.82% for defaulted loans. The interest rate difference between defaulted and non-defaulted loans is 1.66%.

Q. How does Debt to income (DTI) ratio affect the Loan grade?

```
**Output**
```



	G grade loan	A grade loan
Mean DTI	15.96	12.04

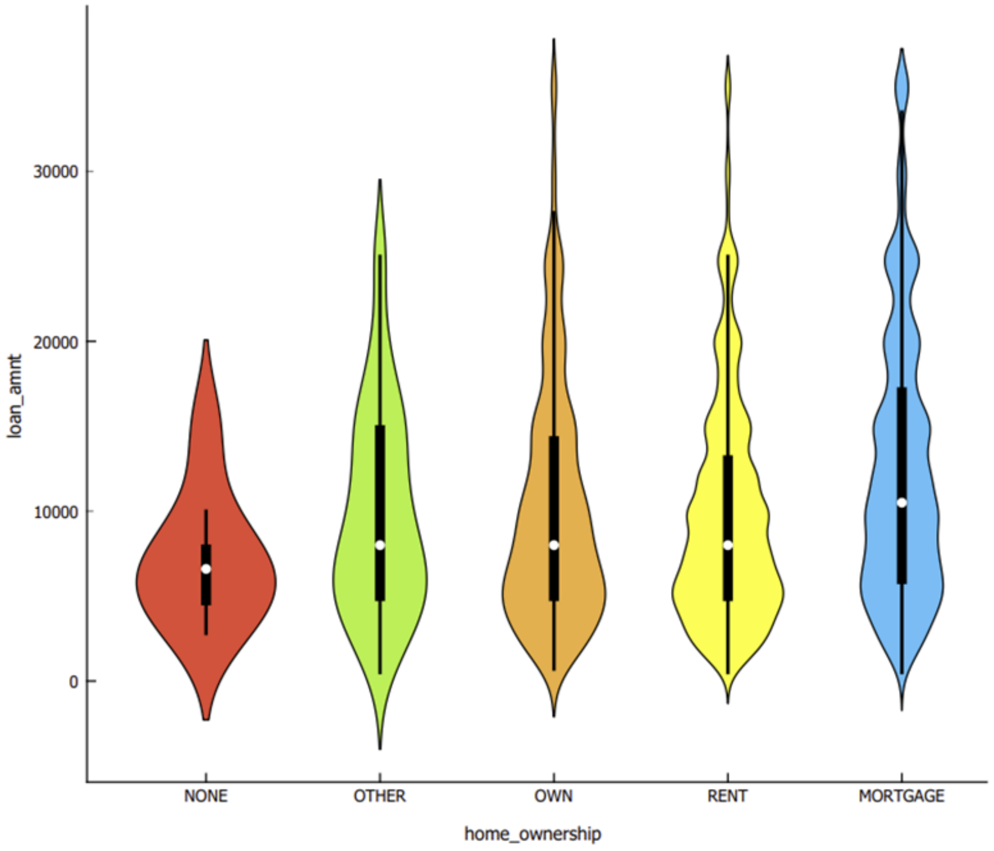
Interpretation

From this box plot, we have observed that having a lower Debt to income ratio (DTI) results in a better loan grade. Here grade A loan category is mostly associated with borrowers who have lower average dti of value 12.04, whereas, borrowers who own grade 'G' loans have the highest average DTI of 15.96. There is 28% difference in DTI, between the above-mentioned loan grade borrower categories.

Q. What type of collateral home ownership influence the loan amount applied for by the borrower?

We also checked if collateral home ownership has any impact on determining the factor for loan defaulting.

```
**Output**
```

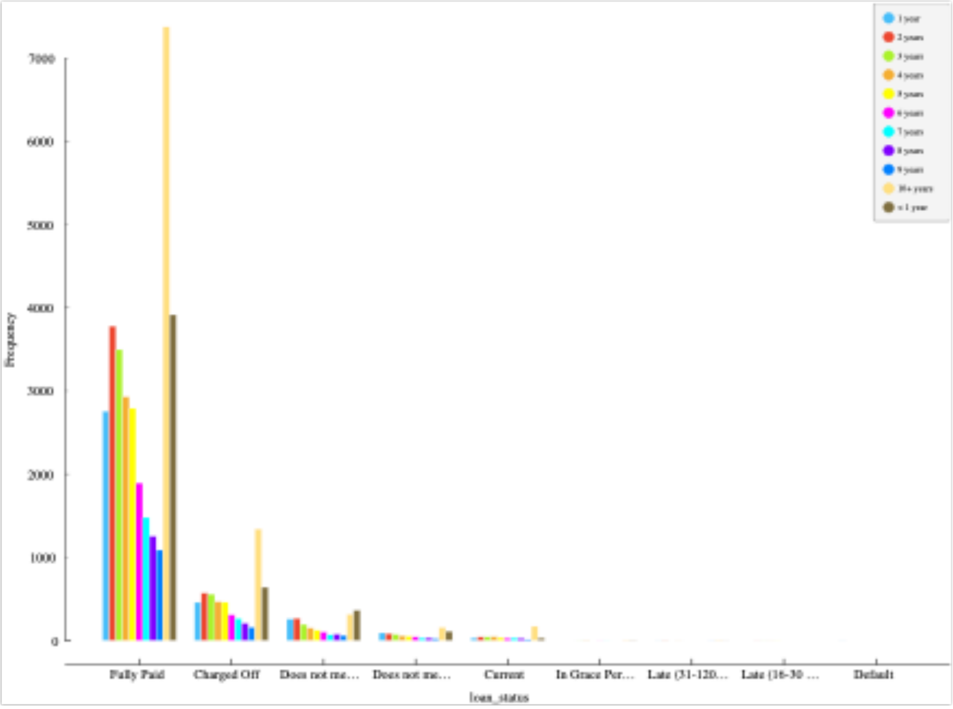


Interpretation

- This graph depicted the mean and density distribution of loan amounts based on home ownership type. Borrowers are classified into seven types of ownership based on their ownership.
- Borrowers with mortgages had the highest average loan amount up to $35,000, while renters had the lowest.

Q. Is there any relationship between employee length and loan status?

```
**Output**
```

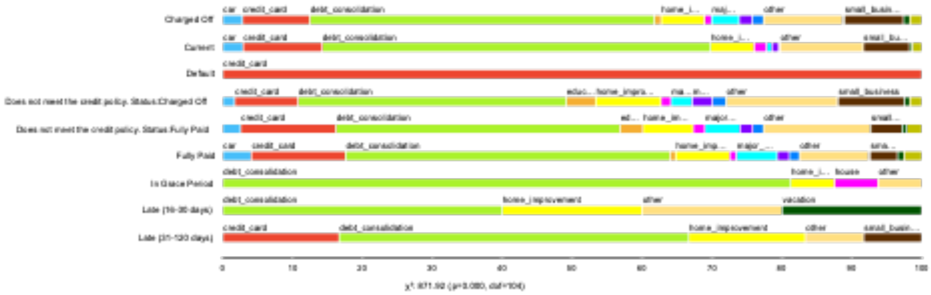


Interpretation

- The distribution graph displays the relationship between two features – employee length and loan status, showing the possible values for one variable and their frequency of occurrence with the other variable.
- In the distribution graph, it can be seen that the most common employment duration for fully paid borrowers is 10+ years, with a minimum of 9 years. The plot is based on the frequency of each category in the variable.
- Thus, Charged-off borrowers tend to have shorter employment duration compared to those who fully paid off their loans.

Q. What are the specific reasons for borrowers to avail a loan?

```
**Output**
```



Interpretation

- The above box plot helps to identify the reasons for availing a loan for each loan categories
- The main reason for a charged off borrower to take a loan is debt consolidation, credit card clearance, and home improvement.
- Thus, it can be inferred that refinancing (debt consolidation and credit card clearance) is an overall reason for all the category of the borrower

Appendix-1: Metadata

Variable	Description
dti	A ratio calculated using the borrower's total monthly debt payments on the total debt obligations, excluding mortgage and the requested LC loan, divided by the borrower's self-reported monthly income.
Emp_length	Employment length in years. Possible values are between 0 and 10 where 0 means less than one year and 10 means ten or more years.
Funded amount	The total amount committed to that loan at that point in time.
Home_ownership	The home ownership status provided by the borrower during registration. Our values are: RENT, OWN, MORTGAGE, OTHER.
Int_rate	Interest Rate on the loan
Purpose	A category provided by the borrower for the loan request.
Revol_bal	Total credit revolving balance
Total_Payment	Payments received to date for total amount funded
Fico_range_high	The upper boundary range the borrower's FICO at loan origination belongs to.
Fico_range_low	The lower boundary range the borrower's FICO at loan origination belongs to.