UpGrad Batch: EPGP ML 52 April 2023

Lending Club Case Study

Presented by Sandeep and Sushil

13-06-2023

Agenda

- Overview
- Analyse relevant variables for analysis
- Manage Missing values
- Standardize the variables' values
- Detect outliers
- Visualize Categorical Data
- Analyse the visual outcome
- Observations
- Conclusion

Overview

Lending Club is a lending platform that lends **money** to people in need at an interest rate based on their credit history and other factors.

When the company receives a loan application, the company has to decide on loan approval based on the applicant's profile. Two types of risks are associated with the bank's decision:

- If the applicant is likely to repay the loan, then not approving the loan results in a loss of business to the company
- If the applicant is not likely to repay the loan, i.e., they are likely to default, then approving the loan may lead to a financial loss for the company

For this case study, the company wants to understand the substantial driving factors behind loan default. We have applied all EDA learning to conclude the driving factors that bind loan default.

Analyse relevant variables for analysis

We consider numerous strategies to eliminate irrelevant variables from the Pandas' data frame. Here is the list of those –

- Find and remove all the null values columns

Checking the null values count in descending order

```
loan_data.isnull().sum().sort_values(ascending=False)
```

Checking the % of null values column wise

```
round(loan_data.isnull().sum()/len(loan_data.index))
```

List out all columns those have 100% null values

```
columns_null_percenta = loan_data.isnull().mean()*100
columns_null_percenta.sort_values(ascending=False)
```

List out the columns those have 97% null values

```
for sn, value in enumerate(loan_data[loan_data.columns[loan_data.isnull().mean()*100 >= 97.0]]):
    print('{0}. {1}'.format(sn, value))
```

Drop all the identified columns

```
for null_value in loan_data[loan_data.columns[loan_data.isnull().mean()*100 >= 97.0]]:
    loan_data = loan_data.drop(null_value, axis=1)
```

Let's check shape of loan_data

```
loan_data.shape
```

(39717, 56)

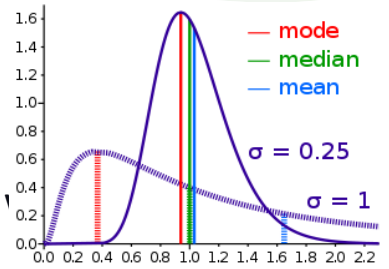
Analyse relevant variables for analysis

- Analyse further and found a couple of variables those have more than 50% null values
  - 'mths_since_last_record': # The number of months since the last public record.
  - 'mths_since_last_delinq': # The number of months since the borrower's last delinquency.
  - 'pub_rec_bankruptcies' : # Number of public record bankruptcies
  - 'last_pymnt_d' : # Last month payment was received
  - 'chargeoff_within_12_mths' : # Number of charge-offs within 12 months
  - 'collections_12_mths_ex_med' : # Number of collections in 12 months excluding medical collections
- Some of these columns plays important role once loan is approved and relationship is established with lending club
  - 'pub_rec' : # Number of derogatory public records
  - 'tax_liens': # Number of tax liens
  - 'url' : #Number of tax liens
  - 'zip_code': # The first 3 numbers of the zip code provided by the borrower in the loan application.
  - 'id' : # A unique LC assigned ID for the loan listing.
  - 'member_id': # A unique LC assigned Id for the borrower member.
  - 'pymnt_plan': # Indicates if a payment plan has been put in place for the loan
  - 'earliest_cr_line': # The month the borrower's earliest reported credit line was opened
  - 'initial_list_status', 'last_credit_pull_d', 'out_prncp', 'out_prncp_inv', 'total_rec_late_fee', 'recoveries'
  - 'collection_recovery_fee', 'last_pymnt_amnt', 'acc_now_delinq', 'delinq_amnt','open_acc'
  - 'inq_last_6mths', 'total_rec_prncp', 'inq_last_6mths', 'revol_bal', 'total_acc', 'revol_util', 'application_type'

Manage Missing Values

Here are methods to treat missing values –

- Mean/Mode/Medium imputation: It is one of the most adopted method to deal with missing values. It consists of replacing the missing data for a given attribute with the mean or median (quantitative characteristic) or mode (qualitative characteristic) of all known values of that variable.



- Data deletion: This method uses when the nature of missing data is "**Missing completely at random**" or we have a good amount of data, and the data loss v non-random missing values can bias the model output.

Example: Handle the missing values of the emp_length variable.

```
In [36]: loan_data['emp_length'] = loan_data['emp_length'].fillna(loan_data['emp_length'].median())

In [37]: loan_data['emp_length'] = pd.to_numeric(
             loan_data['emp_length'].apply(lambda x: str(x).replace('nan', str(loan_data['emp_length'].median()))))

In [38]: loan_data[[ 'emp_length', 'title', 'emp_title', ]].isnull().sum()

Out[38]: emp_length       0
         title           11
         emp_title     2459
         dtype: int64
```

Standardize the variables' values

Here are a few columns we identified to transform into the standard value.

- Interest rate column has percentage symbol(“%”) and needs to remove

```
In [252]: loan_data['int_rate'] = pd.to_numeric(loan_data['int_rate'].apply(lambda x: x.replace('%', '')))
```

- Employee length column has less than and plus symbols(“<“, “+”) and need to remove

```
In [34]: loan_data['emp_length'] = loan_data['emp_length'].apply(
             lambda x: 0 if '<' in str(x).split()[0] else str(x).split()[0]
         )
```

```
In [35]: loan_data['emp_length']= loan_data['emp_length'].apply(
             lambda x: str(x).replace('+', '') if '+' in str(x) else x
         )
```

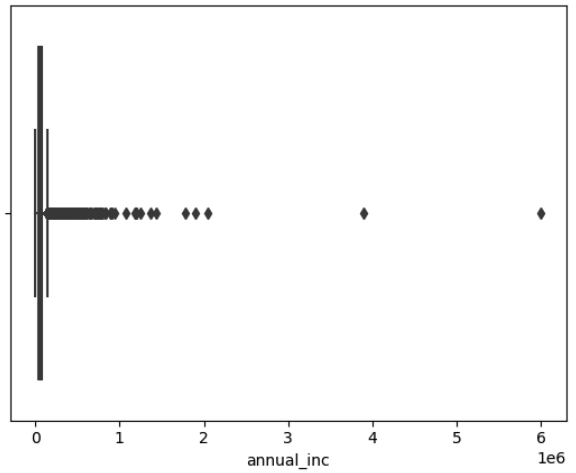
Detect Outliers

We identified few columns and did analyse on those columns and found most of the column contains continuous data values except for the annual income column.

We can identify outliers in the attached screenshot in the right topmost section, and we have also determined that above 0.95 percentile values are outliers.
We have explained how to overcome with

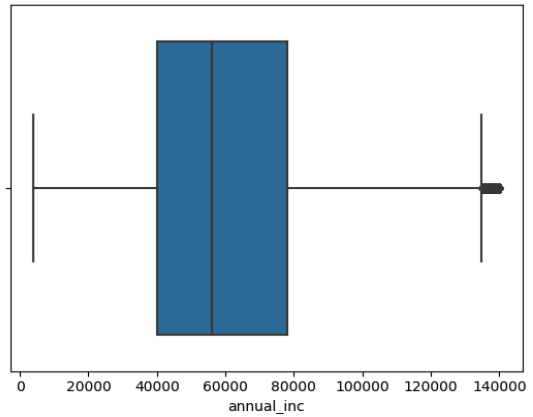
these outliers.

```
In [194]: sns.boxplot(data=dataframe, x="annual_inc")
Out[194]: <Axes: xlabel='annual_inc'>
```




```
In [198]: per_95_annual_inc = dataframe.annual_inc.quantile(0.95)
          dataframe = dataframe[dataframe.annual_inc <= per_95_annual_inc]

In [199]: sns.boxplot(data=dataframe, x="annual_inc")
Out[199]: <Axes: xlabel='annual_inc'>
```

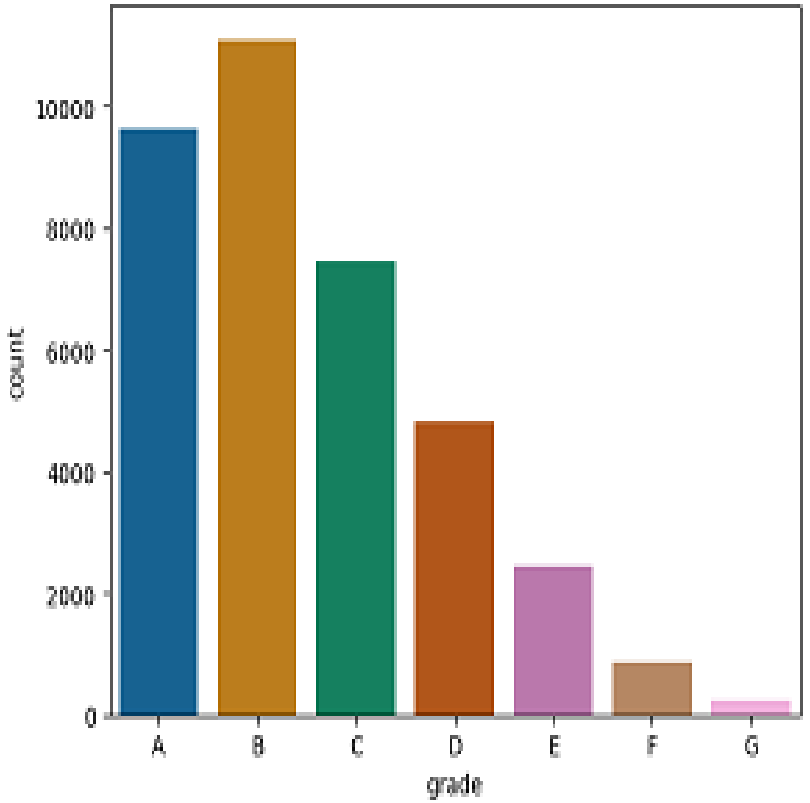


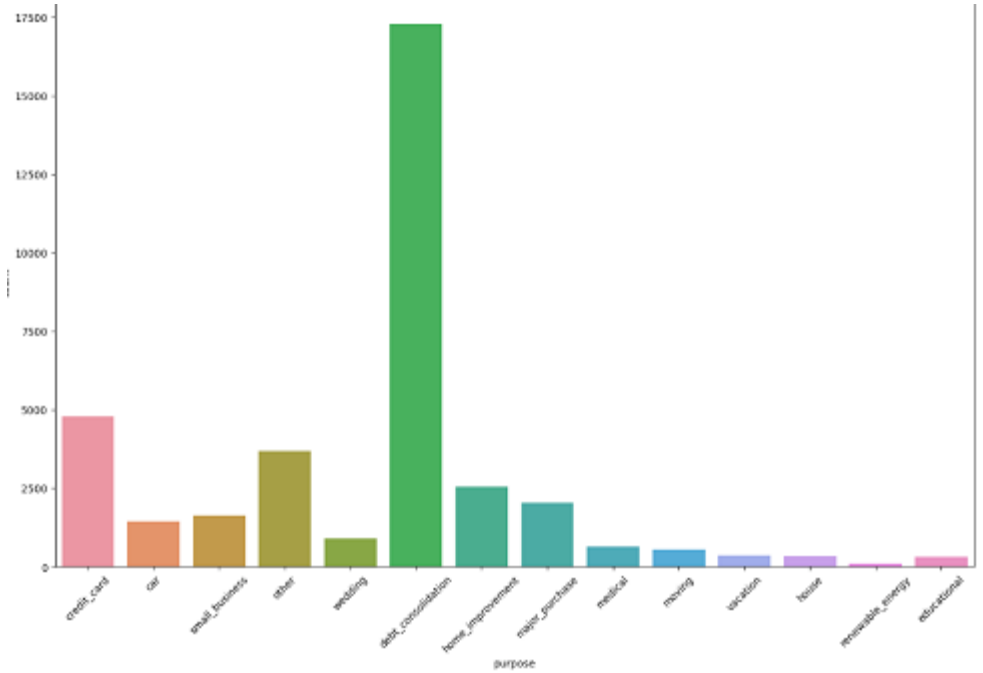
Univariate Analysis

Univariate analysis statistical technique used to examine and summarize data on a single variable.

Target variable is loan_status which is plotted against the all continuous and categorical variables.

We have analysed against ‘grade’ and ‘purpose’ variables against the loan_status.



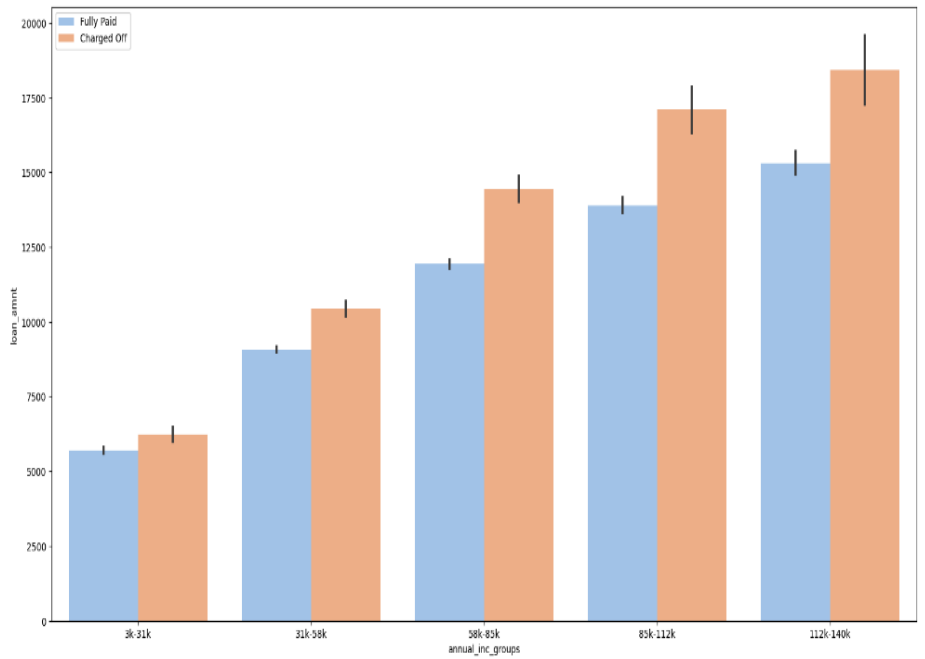


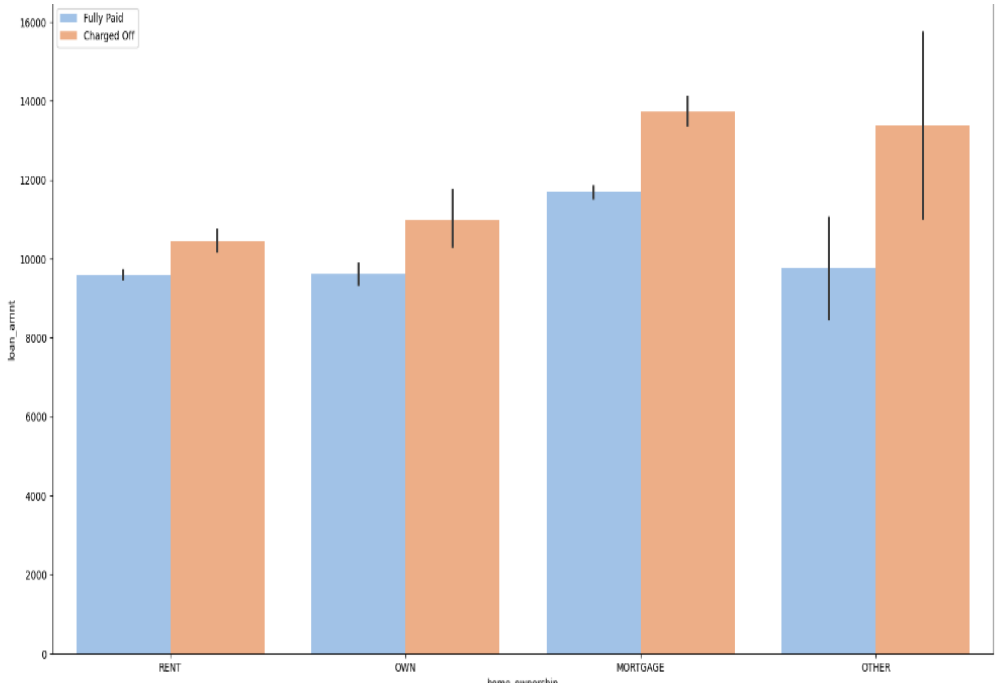
Bivariate Analysis

Bivariate analysis statistical technique is used to examine and summarize data on two variables.

Target variable is loan_status which is plotted against one continuous and one categorical variables.

We have analysed between loan income group and loan amount , home owner and loan amount and we can clearly see that higher income group have higher loan amount and mortgage ownership and income group is above 14k to be default.





Univariate Observations

- **Loan Grade**: Loans with a grade of 'G' have a higher likelihood of defaulting.
- **Sub Grade Level**: Loans with a total grade of 'F5' have a higher probability of defaulting.
- **House Ownership**: Applicants who have house_ownership listed as 'RENT' are more likely to default on loans compared to other types of house ownership.
- **Verification**: Loans that are not verified have a higher probability of defaulting.
- **Loan Purpose**: Applicants who use the loan to clear other debts, specifically for debt consolidation, have a higher probability of defaulting.
- **Loan Term**: Loans with a term of 36 months are more likely to default compared to longer-term loans.
- **Interest Rate**: Applicants who receive loans with an interest rate in the range of 13-17% are more likely to default.
- **Employment Length**: Applicants who has more than 10 years of working status are likely to default.
- **Income Range**: Applicants with an income between the range of 31,201-58,402 have a higher probability of defaulting on loans.
- **Loan Amount**: Loans with amounts between 5,429-10,357 have a higher likelihood of defaulting.
- **Funded Amount**: Loans with a funded amount by investors between 5,000-10,000 are associated with a higher probability of defaulting.

Bivariate Observations - I

- **Loan Grade and Loan Amount:** Loans with a grade of 'F' and a loan amount between 15,000-20,000 are more likely to default.
- **Home Ownership and Loan Amount**: Applicants with 'MORTGAGE' as their home ownership status and a loan amount in the range of 14,000-16,000 have a higher probability of defaulting.
- **Home Ownership and Income**: Applicants with 'MORTGAGE' as their home ownership status and an income of 60,000-70,000 have a higher probability of defaulting.
- **Loan Purpose and Loan Amount**: Applicants who have taken a loan for a small business and the loan amount is greater than 14,000 are associated with a higher likelihood of defaulting.
- Loan Purpose and Income: Applicants who take a loan for 'home improvement' and have an income in the range of 60,000-70,000 are more likely to default.
- **Interest Rate and Income:** Applicants who receive loans with an interest rate in the range of 21-24% and have an income of 70,000-80,000 are more likely to default.
- **Loan Amount and Interest Rate :** Applicants who have taken a loan in the range of 30,000-35,000 and are charged an interest rate of 15-17.5% have a higher probability of defaulting.
- **Loan Grade and Interest Rate** : Loans with a grade of 'G' and an interest rate above 20% are associated with a higher probability of defaulting
- **Loan Purpose and Interest**: Applicants who has taken loan for house are likely to be defaulting the loan.
- **Income Group and Interest**: Applicants who has taken loan for house and interest above 14% are likely to be defaulting the loan.

Bivariate Observations - II

- **Loan Purpose and Installment**: Applicants who has taken the loan for small business and monthly installment above 400 are a higher probability of defaulting.
- **Income Group and Installment**: Applicants are in the amount group 112k-140k and monthly installment above 500 are high change to be default.
- **Employment Length and Loan Purpose** : Applicants with an employment length of more 5 years and a loan purpose are credit card or vacation are a higher probability of defaulting.
- **Employment Length and Loan Amount** : Applicants with an employment length of 10 years and a loan amount in the range of 12,000-14,000 have a higher probability of defaulting.
- **Funded Amount and Interest**: Applicants who have taken a loan in the range of 30,000-35,000 and are charged an interest rate of 15-17.5% have a higher probability of defaulting.
- **Home Ownership and Installment**: Applicants with 'Other' as their home ownership status and monthly installment above 400 are a higher probability of defaulting.
-

Thank You

(Sandeep & Sushil)