Lending Club Case Study

By- Kumar Deblin

Batch: ML C62

Contents

Understanding Data

- Data Information
- RangeIndex: 39717 entries
- Columns: 112 entries
- id to Unnamed: 111
- Data types: float64(75), int64(13), object(24)
- Memory usage: 33.9+ MB

Data Cleaning

- Columns having Null Values can be removed :

```
#To Show how many Columns have Null Values
print("Number of Columns having Null value :",len(loan_df.columns[100*loan_df.isnull().mean()==100]))
```

```
Number of Columns having Null value : 55
```

- Columns having only one unique value in all the rows can be removed:

```
In [362]:
#Searching number of unique values in each vcolumn
loan_df.nunique().sort_values().head(15)
```

```
Out[362]: tax_liens                   1
delinq_amnt                 1
chargeoff_within_12_mths    1
acc_now_delinq              1
application_type            1
policy_code                 1
collections_12_mths_ex_med  1
pymnt_plan                  1
initial_list_status         1
term                        2
next_pymnt_d                2
verification_status         3
pub_rec_bankruptcies        3
loan_status                 3
pub_rec                     5
dtype: int64
```

Data Cleaning

- out_prncp, out_prncp_inv variables are valid for barrowers who already took loan, so these can be removed
- Removing current value rows in “loan status” column as we want analyze the data.
- There is empty space at the start of values:

```
In [315]: #Checking unique characters for 'term' column
          cleaned_loan_df.term.unique()

Out[315]: array([' 36 months', ' 60 months'], dtype=object)
```

```
In [316]: #Removing empty space in values in 'term' column
          cleaned_loan_df['term'] = cleaned_loan_df.term.str.strip()
          cleaned_loan_df.term.unique()

Out[316]: array(['36 months', '60 months'], dtype=object)
```

- After removing columns:
  - Total number of columns remaining: 46
  - Total number of columns dropped: 66

Data Cleaning

- Type conversion and segregating date type into months, years and weekdays in new columns

In [318]:
```
#Removing '%' Character
cleaned_loan_df['int_rate'] = cleaned_loan_df.int_rate.str.strip('%').astype(float)
cleaned_loan_df['revol_util'] = cleaned_loan_df.revol_util.str.strip('%').astype(float)
```

In [319]:
```
#converting to date type
cleaned_loan_df['issue_d'] = pd.to_datetime(cleaned_loan_df.issue_d, format='%b-%y')
#if there are any data before unix timestamp they are converted to 21 century values,
#so subtracting values more than 2020  with 100 will give more actual time
cleaned_loan_df['issue_d'] = cleaned_loan_df['issue_d'].apply(lambda x: x-pd.DateOffset(years=100) if x.year > 2020 else x)
cleaned_loan_df['earliest_cr_line'] = pd.to_datetime(cleaned_loan_df.earliest_cr_line, format='%b-%y')
cleaned_loan_df['earliest_cr_line'] = cleaned_loan_df['earliest_cr_line'].apply(lambda x: x-pd.DateOffset(years=100) if x.yea
```

In [320]:
```
cleaned_loan_df['issue_d_year'] = cleaned_loan_df.issue_d.dt.year
cleaned_loan_df['issue_d_month'] = cleaned_loan_df.issue_d.dt.strftime('%b')
cleaned_loan_df['issue_d_weekday'] = cleaned_loan_df.issue_d.dt.weekday
#data type conversion of year and weekday
cleaned_loan_df['issue_d_year'] = cleaned_loan_df['issue_d_year'].astype(object)
cleaned_loan_df['issue_d_weekday'] = cleaned_loan_df['issue_d_weekday'].astype(object)

#earliest_cr_line
cleaned_loan_df['earliest_cr_line_year'] = cleaned_loan_df.earliest_cr_line.dt.year
cleaned_loan_df['earliest_cr_line_month'] = cleaned_loan_df.earliest_cr_line.dt.strftime('%b')
#data type conversion of year and weekday
cleaned_loan_df['earliest_cr_line_year'] = cleaned_loan_df['earliest_cr_line_year'].astype(object)
```

Data Cleaning

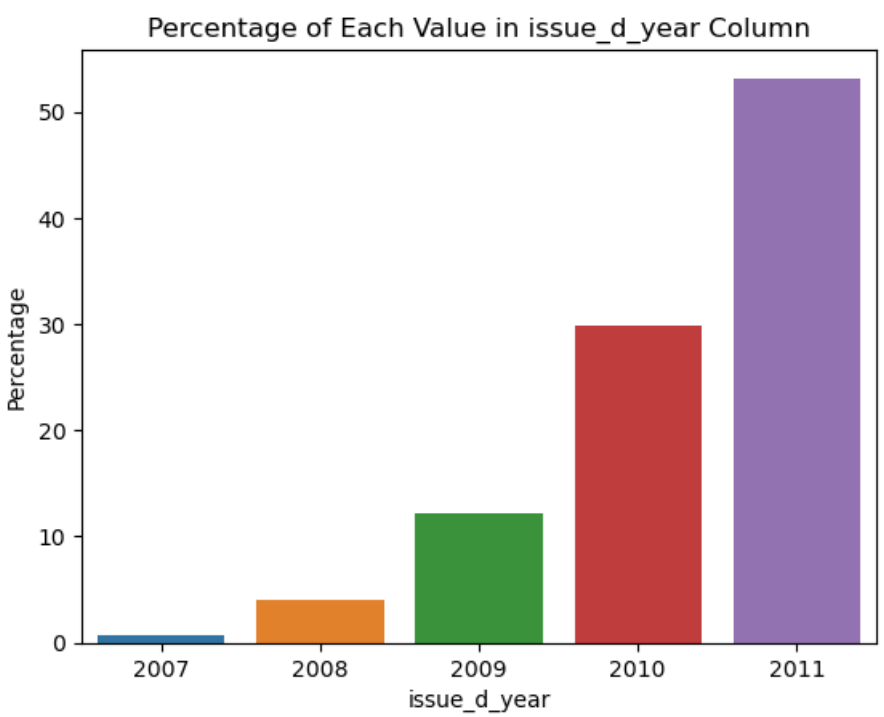
- Adding Approved Loan amount ratio which is a ratio of Funded Amount by investor to Requested Loan amount.

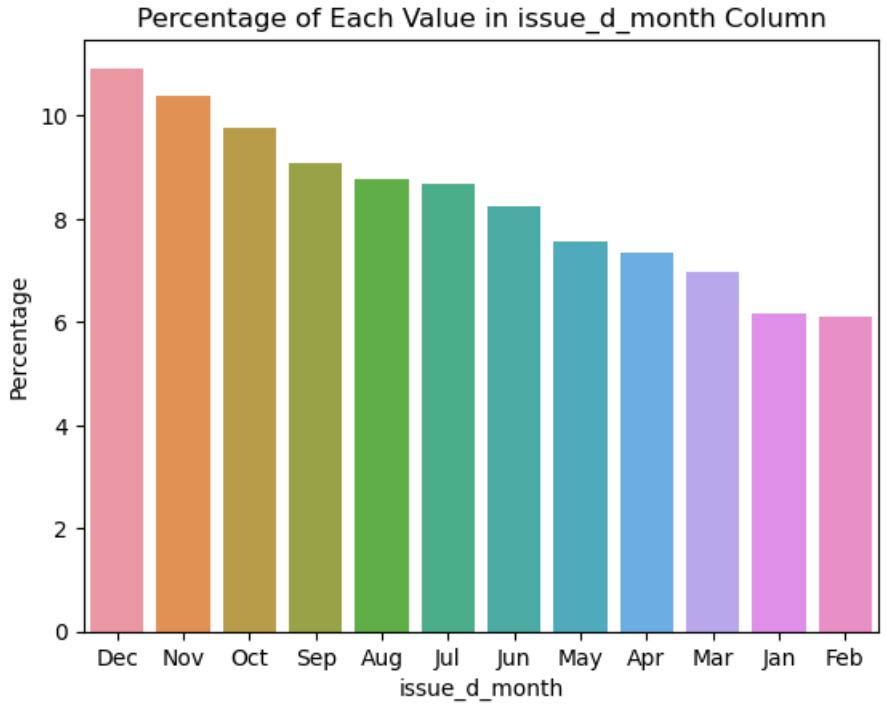
```
#Adding Approved Loan amount ratio which is a ratio of Funded Amount by investor to Requested Loan amount.

cleaned_loan_df['approved_loan_amnt_ratio'] = round(cleaned_loan_df.funded_amnt_inv*100/cleaned_loan_df.loan_amnt,2)
```

- Summarizing Columns:
  - Total number of columns dropped: 66
  - Total number of columns remaining: 46
  - New Columns added: 5
  - Final Total number of columns remaining: 51

Univariate Analysis

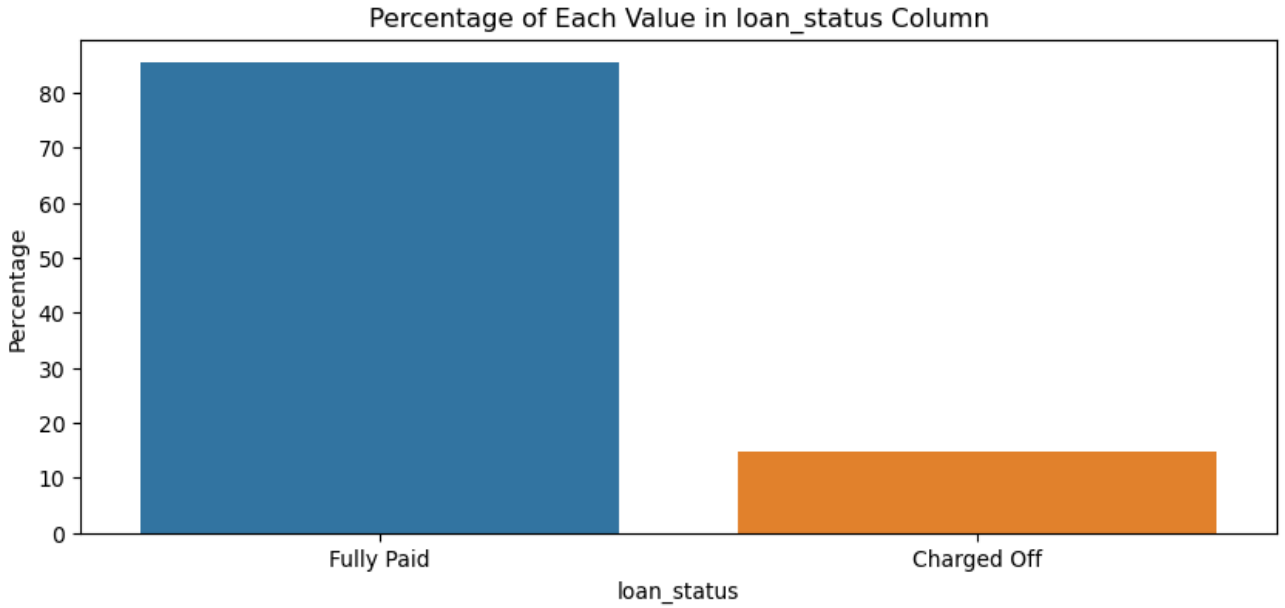
- Lending club has really expanded year by year, the number of loan issued are doubled every year.
- Also the issued month of loans is also increasing from January to December. In the final quarter of year there are more loans issued this could be because of vacation and christmas

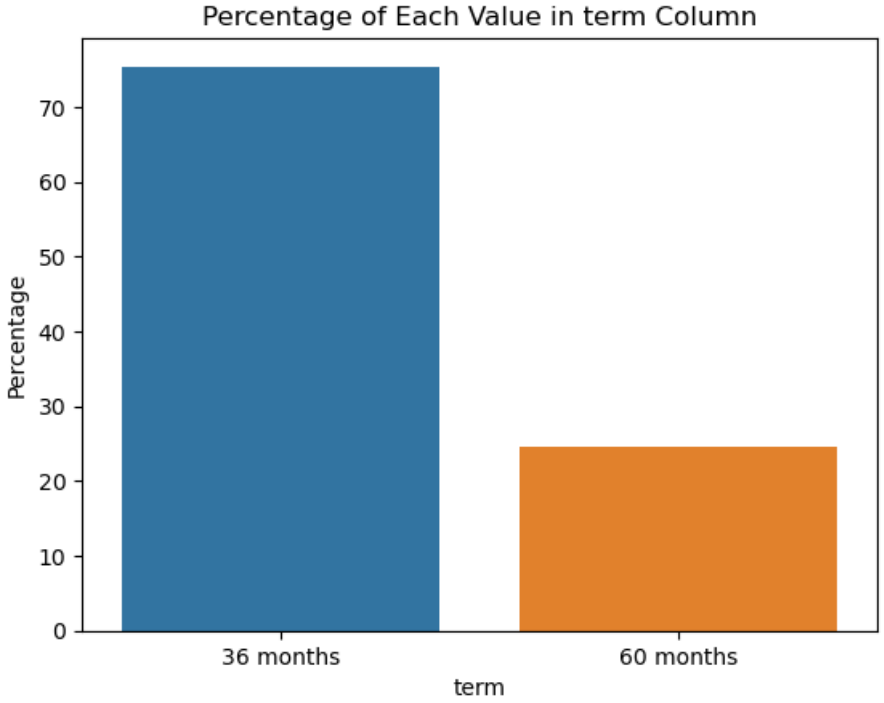




Univariate Analysis

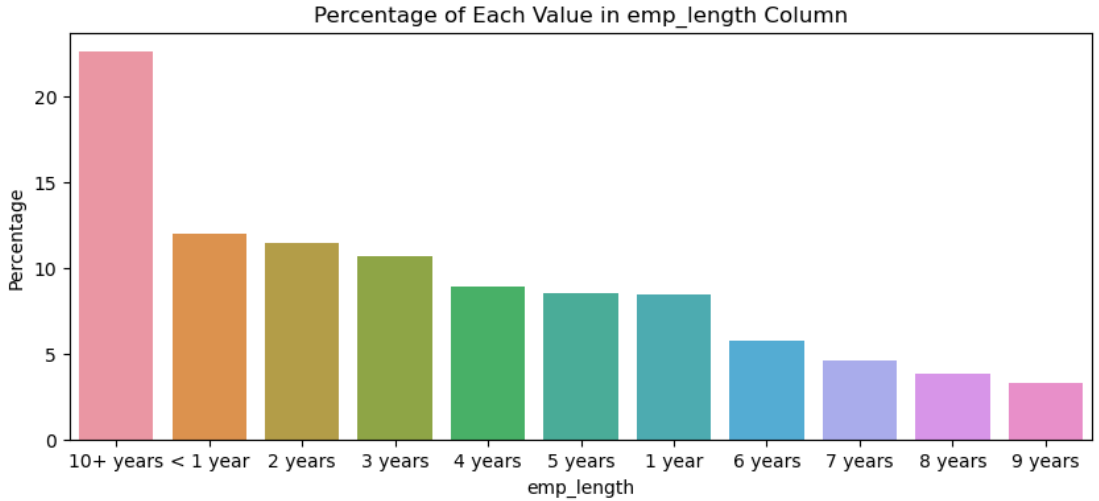
- Charged off borrowers are around 15% and fully paid is 85%
- Only two loan terms 36 and 60 months

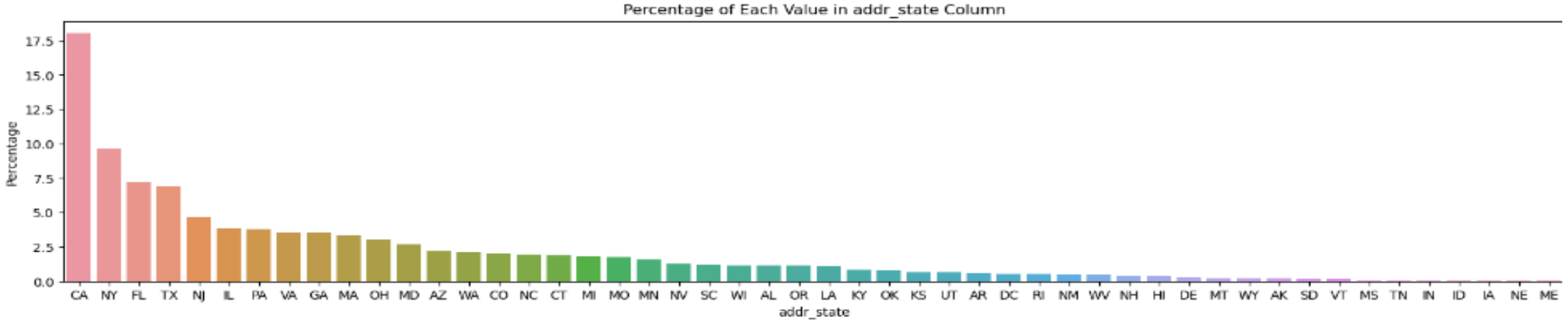




Univariate Analysis

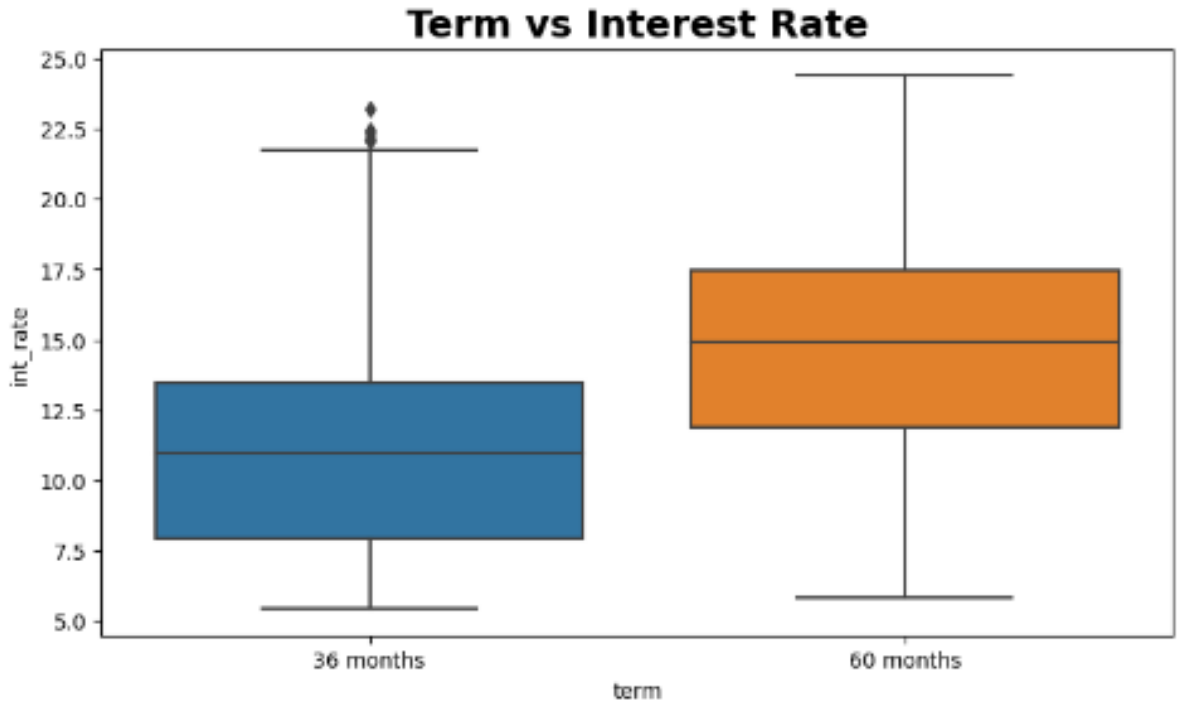
- Borrowers having more employee length are more in number
- High number of borrowers are from CA California.





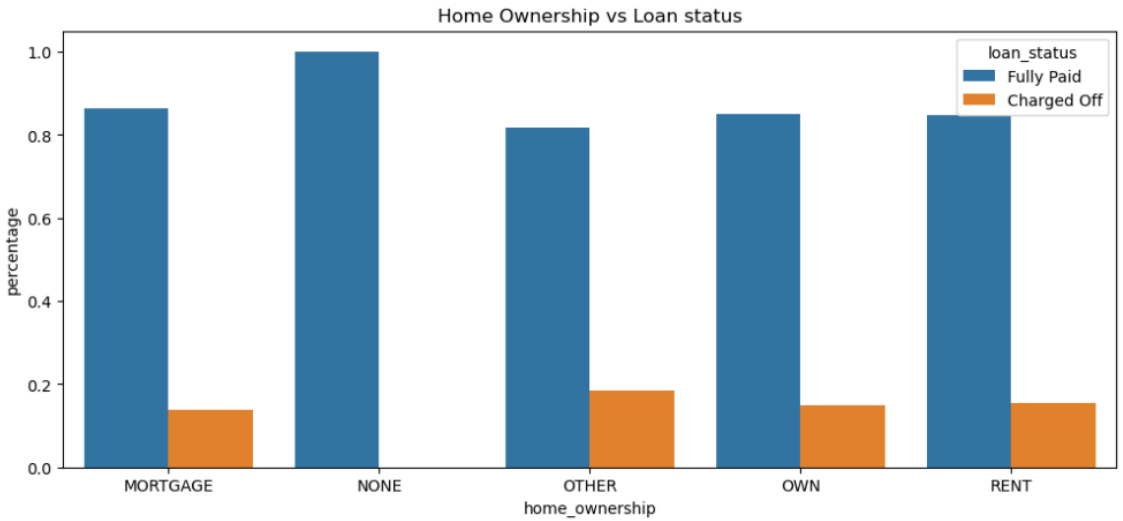
Segmented Univariate Analysis

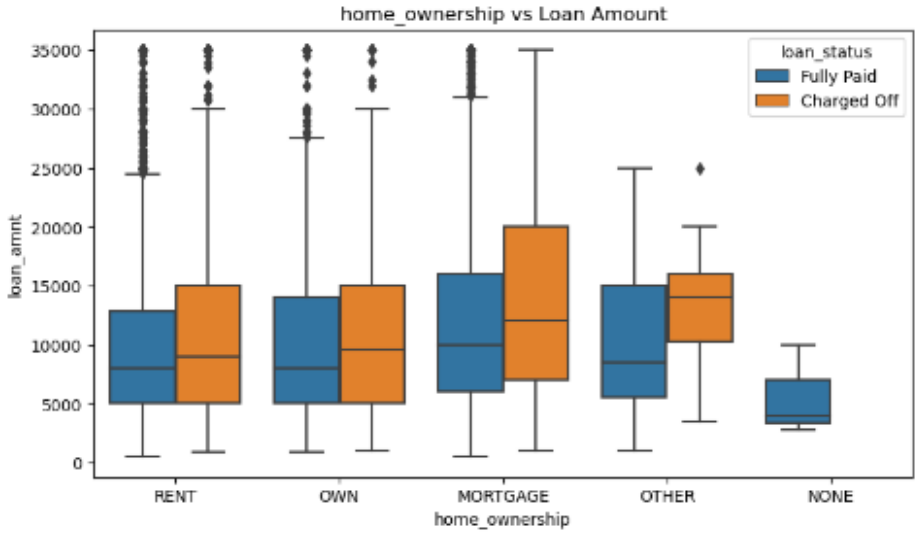
- High interest rate loan, there is high chance of loan getting defaulted



Bivariate Analysis

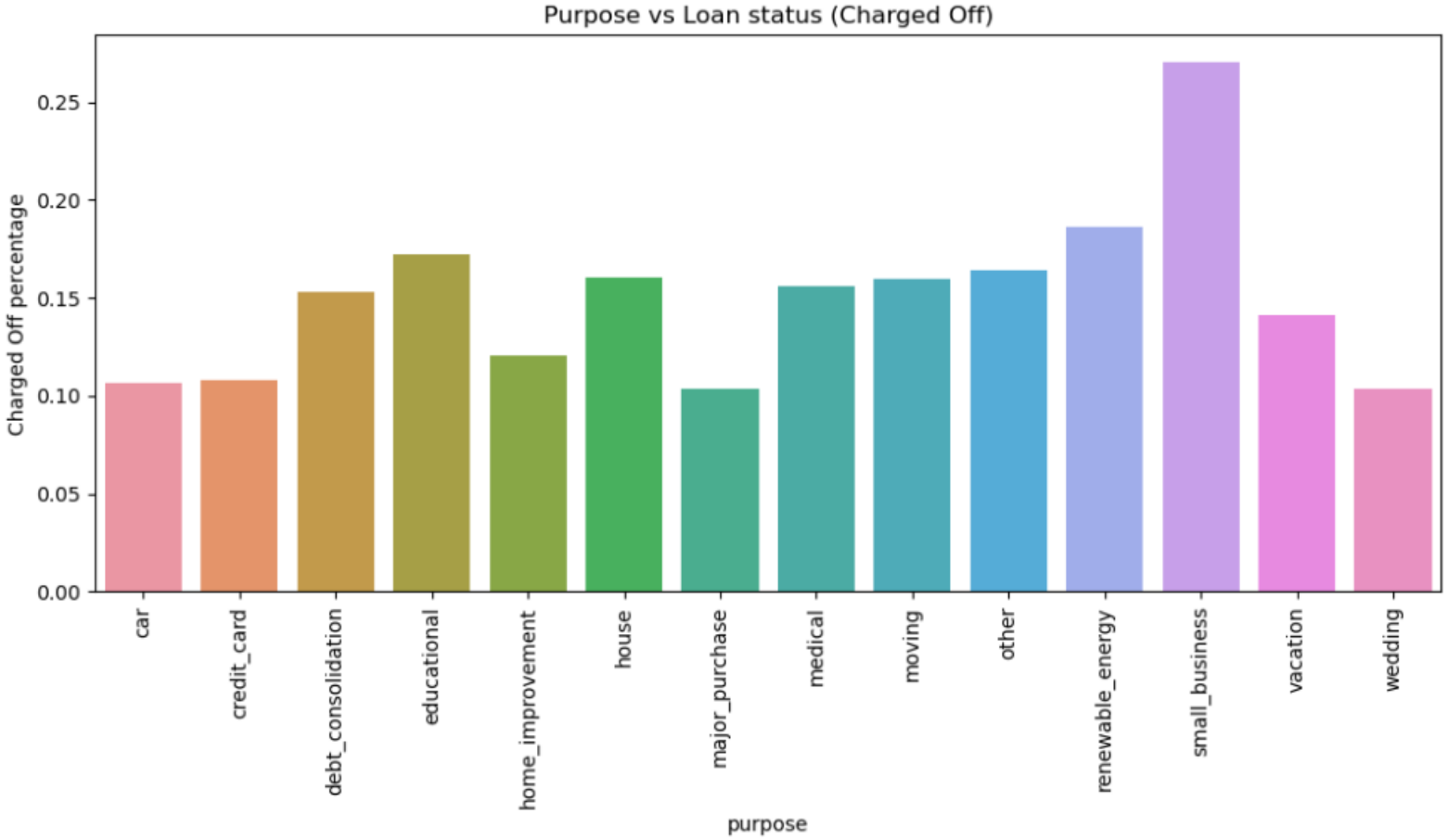
- 20% chance of loan default in each home ownership category.
- The people with higher loan amounts in **mortgage** home ownership has high default rate than others.





Bivariate Analysis

- Small business loans are charged off highly



Conclusion

- Lending club should reduce the high interest loans for 60 months tenure, they are prone to loan default.
- Lending Club should control their number of loan issues to borrowers who are from CA, FL and NY to make profits.
- Borrowers with mortgage home ownership are taking higher loans and defaulting the approved loans. Lending club should stop giving loans to this category when loan amount requested is more than 12000.
- Small business loans are defaulted more. Lending club should stop/reduce issuing the loans to them.

Thank You…