Lending Club Case Study

Lending Club:

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. We will analyze data and pre-process it based on our need and build an EDA model that can identify a potential defaulter based on his/her history of transactions with Lending Club.

We will work on Following points:

1. Will analyze Data
2. Then Data Cleaning
    1. Handling Null Values.
    2. Dropping unnecessary columns
    3. Checking Numeric Columns
    4. Dropping Unnecessary numeric columns
    5. Converting Categorical columns to Numeric Columns
    6. Converting columns like rate_int to numeric columns
    7. Converting Datetime to numeric columns
3. Data Analysis: EDA
    1. We identified univariate variables
    2. Bivariate variables
    3. Segmented Variables

Problem Statement:

We are working for a **consumer finance company** which specializes in lending various types of loans to urban customers. When the company receives a loan application, the company has to make a decision for loan approval based on the applicant's profile. Two **types of risks** are associated with the bank's decision:

1. If the applicant is **likely to repay the loan**, then not approving the loan results in a **loss of business** to the company
2. If the applicant is **not likely to repay the loan,** i.e. he/she is likely to default, then approving the loan may lead to a **financial loss** for the company

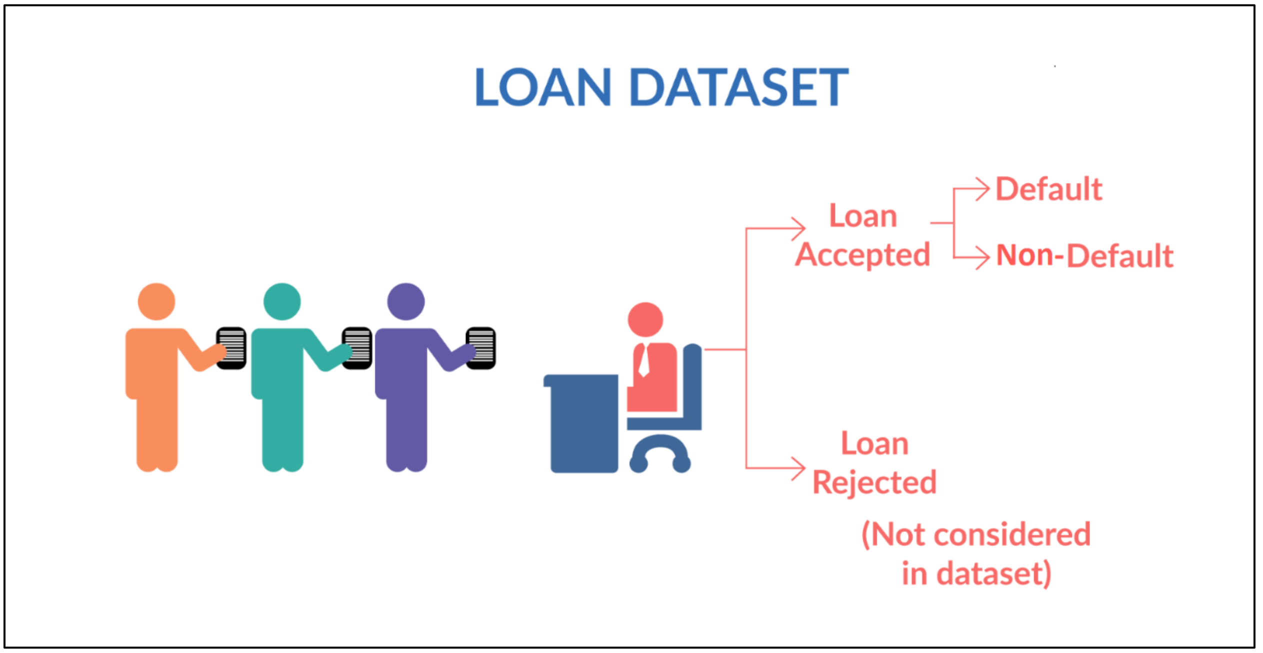
SOLUTIONS:

Data Analysis:

When a person applies for a loan, there are two types of decisions that could be taken by the company:

1. Loan accepted: If the company approves the loan, there are 3 possible scenarios described below:
    - Fully paid: Applicant has fully paid the loan (the principal and the interest rate)
    - Current: Applicant is in the process of paying the instalments, i.e. the tenure of the loan is not yet completed. These candidates are not labelled as 'defaulted'.
    - Charged-off: Applicant has not paid the instalments in due time for a long period of time, i.e. he/she has defaulted on the loan
2. Loan rejected: The company had rejected the loan (because the candidate does not meet their requirements etc.). Since the loan was rejected, there is no transactional history of

those applicants with the company and so this data is not available with the company (and thus in this dataset)



- **Data Cleaning**:

  Checking following points in data cleaning:

1. Checking duplicates on ID column: we did not find duplicate row in loan dataset.
2. How many rows are having all missing values: There are no empty rows in whole dataset.
3. How many columns have all missing values: There 54 columns have all missing values.

```
# check shape & datatype of loan data
print("Check shape of loan data:",loan.shape)

#Data Cleaning Starts
# Check for duplicate rows in dataset based on id column
dup= [loan.duplicated(['id']).sum()]
print("Check for duplicate rows in dataset based on id column:",dup)
# There are no duplicate rows in loan dataset

# sum it up to check how many rows have all missing values
print('Number of empty Rows:',loan.isnull().all(axis=1).sum())
#print(loan.isnull().all(axis=1).sum())
# Observation: There are no empty rows in whole dataset.

# sum it up to check how many columns have all missing values
print('Number of empty Columns:',loan.isnull().all(axis=0).sum())
#print(loan.isnull().all(axis=0).sum())
# Observation: There are 54 columns have all missing values.
```

```
Check shape of loan data: (39717, 111)
Check for duplicate rows in dataset based on id column: [0]
Number of empty Rows: 0
Number of empty Columns: 54
```

1. Finding which columns have all null values and dropping off those columns.

2. Finding columns which has mostly null values and dropping off those columns.
3. After dropping of unnecessary columns, we have 44 columns.
4. Now we are formatting columns like
    a. emp_length : it contains <,> and + symbols . we are removing them and keep only integer
    b. int_rate and revol_util : we are removing percentage symbol from these columns.
    c. convert amount columns into numeric data to find some correlation among important ones. Columns(loan_amnt, funded_amnt, int_rate, funded_amnt_inv, installment, annual_inc, dti, total_pymnt)

➢ EDA

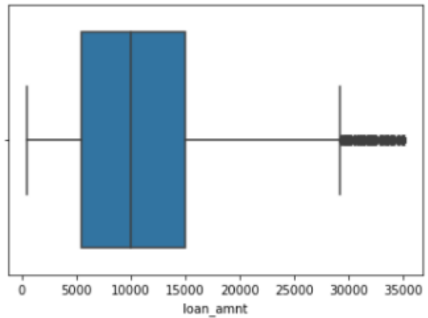
1. We are checking **some important variables** like
    a. **Loan status**:82.96 percent loans were fully paid.
    b. **Loan Purpose**: Most of the loans taken for debt consolidation (47%) and Credit card bill payment.
2. We are checking some **derived variable**s like month and year. We derived those columns from issue date.

➢ Univariate Analysis:

Univariate analysis is one variable analysis. We are finding quantitative variables loan_amt and total_pymnt.

```
Univariate Analysis of Loan_amt
```

Out[96]: <AxesSubplot:xlabel='loan_amnt'>

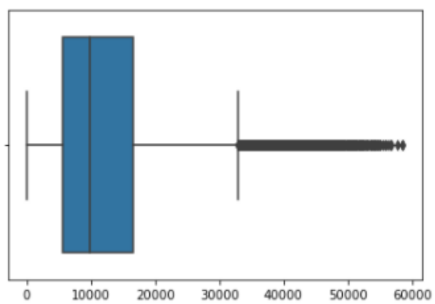


In [95]:

```python
# Basic statistics with .describe() -Quantitative Variables
print("\n\n\nUnivariate Analysis oftotal_pymnt")
loan.total_pymnt.describe()
sns.boxplot(loan.total_pymnt)
```

```
Univariate Analysis oftotal_pymnt
```

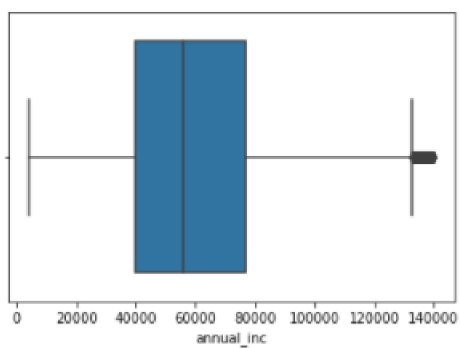
Out[95]: <AxesSubplot:xlabel='total_pymnt'>



1. **<u>Now we are finding outliers and removing outliers.</u>**

```
# Basic statistics with .describe() -Quantitative Variables

print('Before Removal of Outliers :\n')
print(loan['annual_inc'].describe())

# Data cleaning
# Remove Outliers quantile .99 from Annual Income
# it will make it easier to visualize the plots.

loan = loan[loan["annual_inc"] < loan["annual_inc"].quantile(0.99)]

print('\nAfter Removal of Outliers :\n')
print(loan["annual_inc"].describe())

# Now below data looks much better. Lets plot later and find some conclusions

sns.boxplot(loan.annual_inc)
```

```
Before Removal of Outliers :

count     37926.00
mean      61478.12
std       28370.43
min        4000.00
25%       40000.00
50%       56400.00
75%       78000.00
max      149981.00
Name: annual_inc, dtype: float64

After Removal of Outliers :

count     37529.00
mean      60620.51
std       27258.91
min        4000.00
25%       40000.00
50%       56000.00
75%       77065.00
max      139992.00
Name: annual_inc, dtype: float64

<AxesSubplot:xlabel='annual_inc'>
```



2. <u>**Now we are finding correlation between quantitative variables.**</u>
Heatmap with Dendogram(clustermap) to show closeness among numerical variables.

Observations:

a. Loan amount, investor amount, funding amount are strongly correlated.
b. Annual income with DTI(Debt-to-income ratio) is negatively correlated.
c. Debt income ratio is the percentage of a consumer's monthly gross income that goes toward paying debts.
d. That means when annual income is low DTI is high & vice versa.
e. positive correlation between annual income and employment years.

f. That means income increases with work experience.

3. Now we can identify more columns which we can drop as those are not required for further analysis. Columns are: application_type, policy_code, initial_list_status, installment, pymnt_plan.

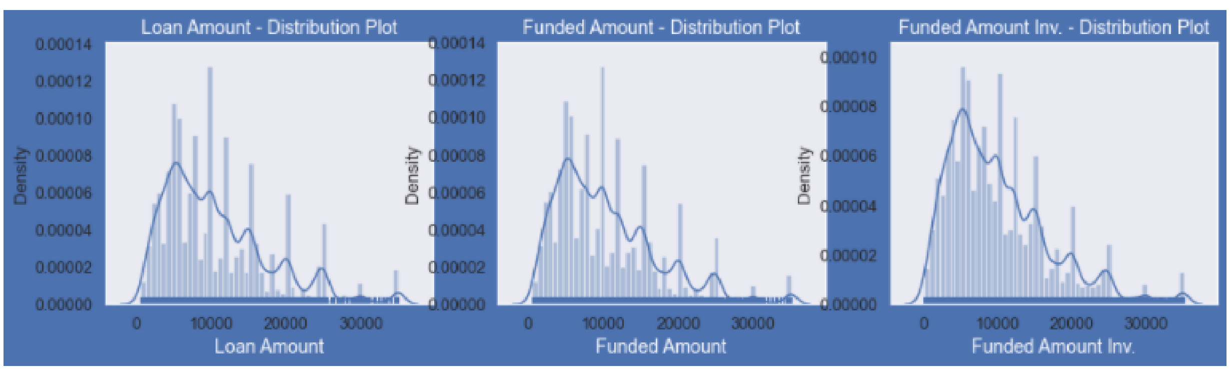
➢ Quantitative Variables

4. Now we are deriving some more columns which will help us in Bivariate analysis.
   So we are categorizing variables like loan amounts into buckets, annual incomes into buckets, interest rate and dti into buckets.

   Let's plot:

```
# Lets see distribution of three loan amount fields using distribution plot.
# Quantitative Variables
print("Distribution of three loan amount fields using distribution plot \n")
plt.figure(figsize=(15,8),facecolor='b')
sns.set_style("dark")
# subplot 1
plt.subplot(2, 3, 1)
ax = sns.distplot(loan['loan_amnt'],rug = True)
ax.set_title('Loan Amount - Distribution Plot',fontsize=14,color='w')
ax.set_xlabel('Loan Amount',fontsize=14,color='w')
# subplot 2
plt.subplot(2, 3, 2)
ax = sns.distplot(loan['funded_amnt'],rug = True)
ax.set_title('Funded Amount - Distribution Plot',fontsize=14,color='w')
ax.set_xlabel('Funded Amount',fontsize=14,color='w')
# subplot 2
plt.subplot(2, 3, 3)
ax = sns.distplot(loan['funded_amnt_inv'],rug = True)
ax.set_title('Funded Amount Inv. - Distribution Plot',fontsize=14,color='w')
ax.set_xlabel('Funded Amount Inv.',fontsize=14,color='w')
plt.show()
```

```
Distribution of three loan amount fields using distribution plot
```

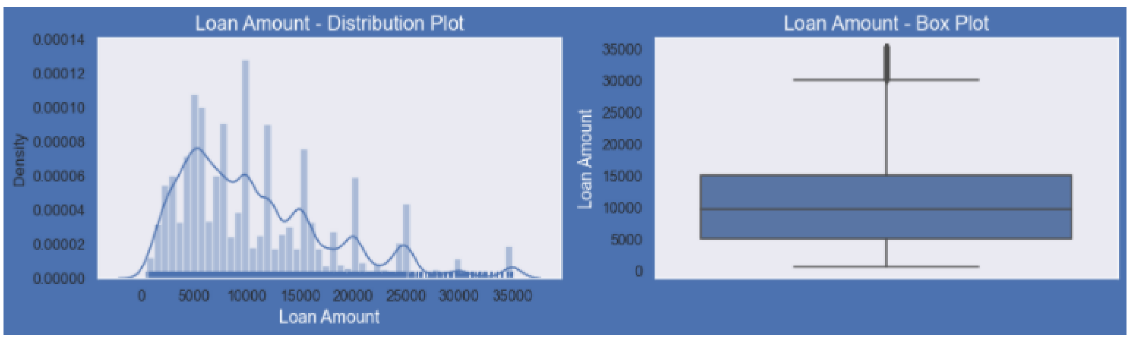


Observations: Distribution of amounts for all three looks very much similar.

Now we will see only loan amount analysis:

```
# Univariate Analysis on Loan amount-Quantitative Variables
print("Now we will see only Loan amount graph \n")

plt.figure(figsize=(15,8),facecolor='b')
sns.set_style("dark")
# subplot 1
plt.subplot(2, 2, 1)
ax = sns.distplot(loan['loan_amnt'],rug = True)
ax.set_title('Loan Amount - Distribution Plot',fontsize=16,color='w')
ax.set_xlabel('Loan Amount',fontsize=14,color='w')
# subplot 2
plt.subplot(2, 2, 2)
ax = sns.boxplot(y=loan['loan_amnt'])
ax.set_title('Loan Amount - Box Plot',fontsize=16,color='w')
ax.set_ylabel('Loan Amount',fontsize=14,color='w')
plt.show()
```

```
Now we will see only Loan amount graph
```

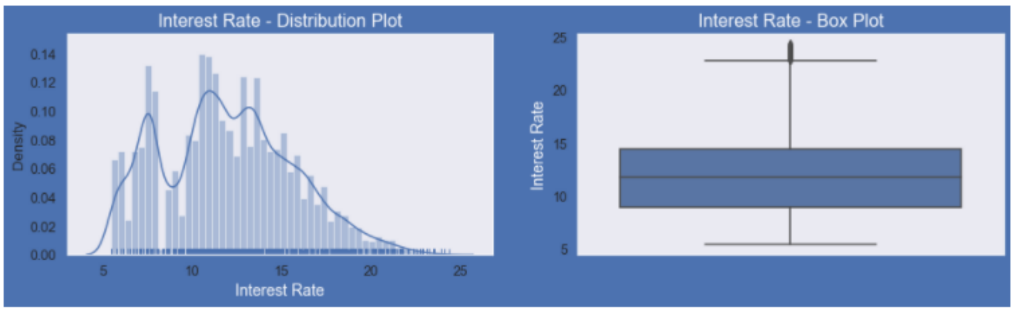


Observations: Above plots show that most of the Loan amounts are in range of 5000 – 15000

Now we will see for interest rate:

```
# Univariate Analysis on Intrest Rate-Quantitative Variables
print("Intrest Rate Analysis\n")
plt.figure(figsize=(15,8),facecolor='b')
sns.set_style("dark")
# subplot 1
plt.subplot(2, 2, 1)
ax = sns.distplot(loan['int_rate'],rug = True)
ax.set_title('Interest Rate - Distribution Plot',fontsize=16,color='w')
ax.set_xlabel('Interest Rate',fontsize=14,color='w')
# subplot 2
plt.subplot(2, 2, 2)
ax = sns.boxplot(y=loan['int_rate'])
ax.set_title('Interest Rate - Box Plot',fontsize=16,color='w')
ax.set_ylabel('Interest Rate',fontsize=14,color='w')
plt.show()
```

```
Intrest Rate Analysis
```

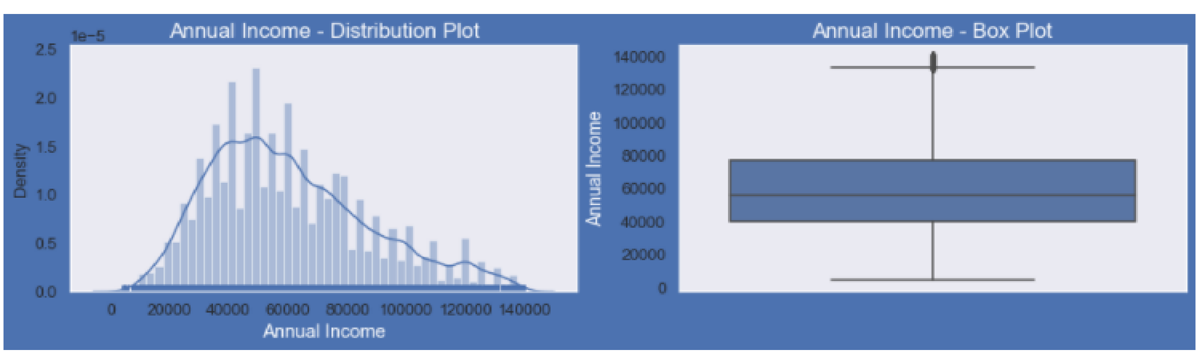


Observations: Above plots show that most of the Interest Rates on loans are in range of 10% - 15%.

Annual income analysis:

```
# Univariate Analysis on Annual Income - Quantitative Variables
print("Annual Income Analysis\n")
plt.figure(figsize=(15,8),facecolor='b')
sns.set_style("dark")
# subplot 1
plt.subplot(2, 2, 1)
ax = sns.distplot(loan['annual_inc'],rug = True)
ax.set_title('Annual Income - Distribution Plot',fontsize=16,color='w')
ax.set_xlabel('Annual Income',fontsize=14,color='w')
# subplot 2
plt.subplot(2, 2, 2)
plt.title('Annual Income - Box Plot')
ax = sns.boxplot(y=loan['annual_inc'])
ax.set_title('Annual Income - Box Plot',fontsize=16,color='w')
ax.set_ylabel('Annual Income',fontsize=14,color='w')
plt.show()

# Observations :
# Below plots show that most of the borrower's Annual incomes are in range of 40000- 80000
```

```
Annual Income Analysis
```



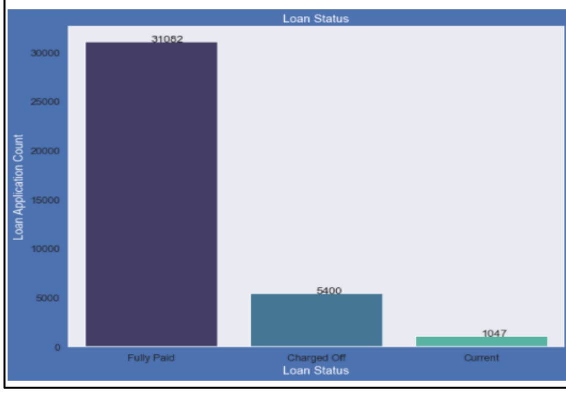
Observations: Above plots shows that the most of the borrower's annual income are in the range 40000-80000.

- ## Unordered Categorical Variables

5. Now we will see about the Loan status:

```
# Univariate Analysis - Unordered Categorical Variables - Loan Status
print("Loan Status\n")
plt.figure(figsize=(10,8),facecolor='b')
sns.set_style("dark")
ax = sns.countplot(x="loan_status",data=loan,palette='mako')
ax.set_title('Loan Status',fontsize=14,color='w')
ax.set_xlabel('Loan Status',fontsize=14,color = 'w')
ax.set_ylabel('Loan Application Count',fontsize=14,color = 'w')
# To show count of values above bars
s=loan['loan_status'].value_counts()
for i, v in s.reset_index().iterrows():
    ax.text(i, v.loan_status + 0.3 , v.loan_status, color='k')

# Observations :
# Below plot shows that close to 14% loans were charged off out of total loan issued.
```

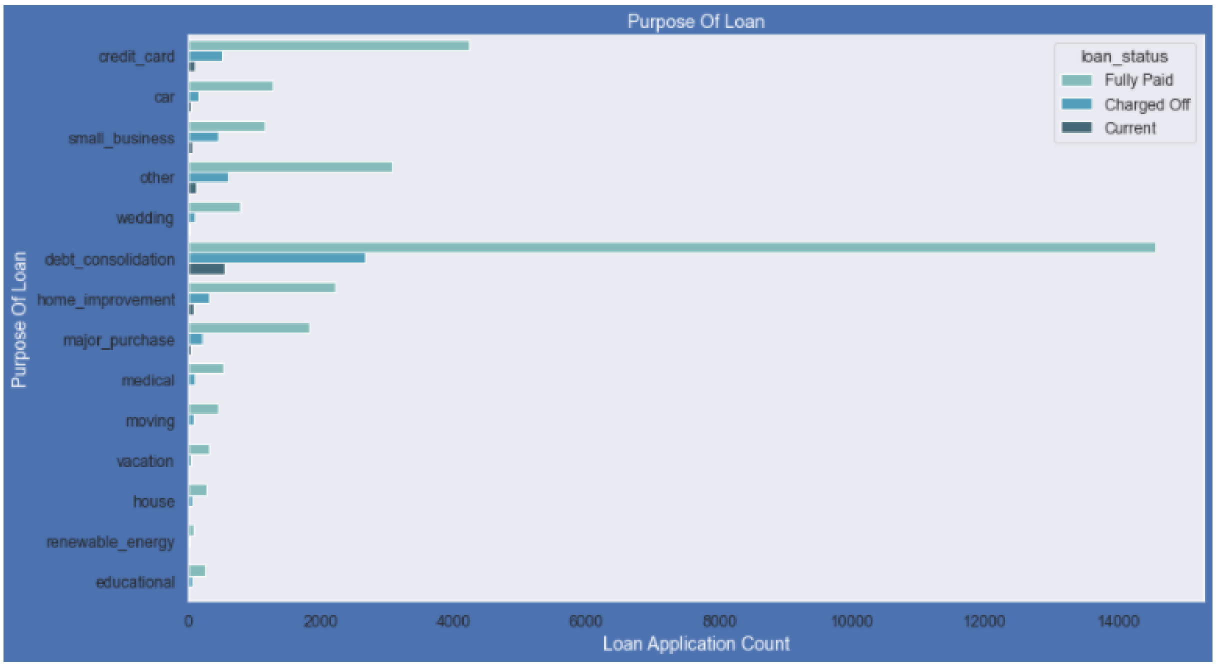
```
Loan Status
```



Observations: Above plot shows that close to 14% loans were charged out of total loan issued.
Purpose of loan:

```python
# Univariate Analysis - Unordered Categorical Variables - Purpose Of Loan
print("Purpose of Loan\n")
plt.figure(figsize=(14,8),facecolor='b')
sns.set_style("dark")
ax = sns.countplot(y="purpose",data=loan,hue='loan_status',palette='GnBu_d')
ax.set_title('Purpose Of Loan',fontsize=14,color='w')
ax.set_ylabel('Purpose Of Loan',fontsize=14,color = 'w')
ax.set_xlabel('Loan Application Count',fontsize=14,color = 'w')
plt.show()

# Observations :
# Below plot shows that most of the loans were taken for the purpose of debt consolidation & paying credit card bill.
# Number of chraged off count also high too for these loans.
```

```
Purpose of Loan
```

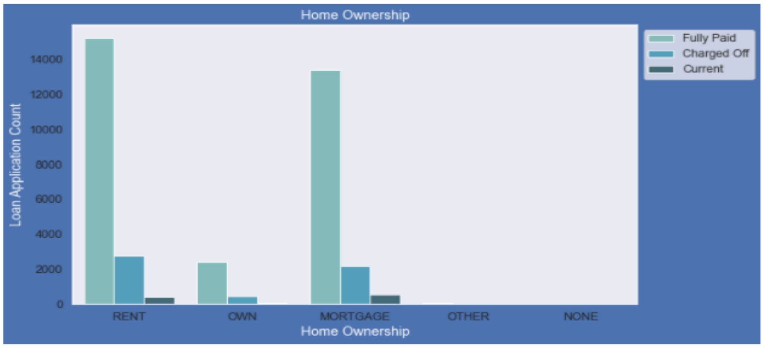


Observations: Above plot shows that most of the loans were taken for the purpose of debt consolidation & paying credit card bill. And number of charged off count also high too for these loans.

Homeownerships:

```
In [140]: # Univariate Analysis - Unordered Categorical Variables - Home Ownership
print("Home Ownership\n")
plt.figure(figsize=(10,6),facecolor='b')
ax = sns.countplot(x="home_ownership",data=loan,hue='loan_status',palette='GnBu_d')
ax.legend(bbox_to_anchor=(1, 1))
ax.set_title('Home Ownership',fontsize=14,color='w')
ax.set_xlabel('Home Ownership',fontsize=14,color = 'w')
ax.set_ylabel('Loan Application Count',fontsize=14,color = 'w')
plt.show()
# Observations :
# Below plot shows that most of them living in rented home or mortgazed their home.
# Applicant numbers are high from these categories so charged off is high too.
```

```
Home Ownership
```



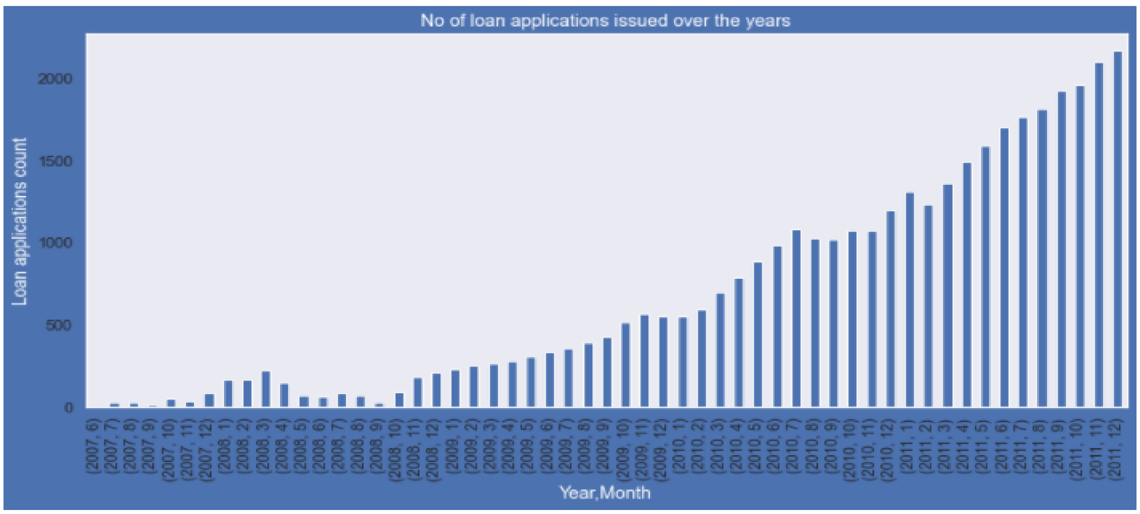
Observations: Above plot shows that most of them living in rented home or mortgazed their home. And applicant numbers are high from these categories so charged off is high too.

➢ Ordered Categorical Variables:

To show ordered categorical variables we can plot histogram or a bar chart and we will observe any unexpected trend in it.

```
# Derived Column - Ordered Categorical Variables
# Let us look into number of loans which were approved every year/month
# Lets use derived column year to check pattern of loan issuing over the years.
print("Ordered categorical variable")
plt.figure(figsize=(14,6),facecolor='b')
loan.groupby(['year','month']).id.count().plot(kind='bar')
plt.ylabel('Loan applications count',fontsize=14,color='w')
plt.xlabel('Year,Month',fontsize=14,color = 'w')
plt.title("No of loan applications issued over the years",fontsize=14,color='w')
plt.show()

# Observation is that count of loan application is increasing every passing year.
# so increase in number of loan applications are adding more to number of charged off applications.
# number of loans issued in 2008( May-October) got dipped, may be due to Recession.
```

```
Ordered categorical variable
```

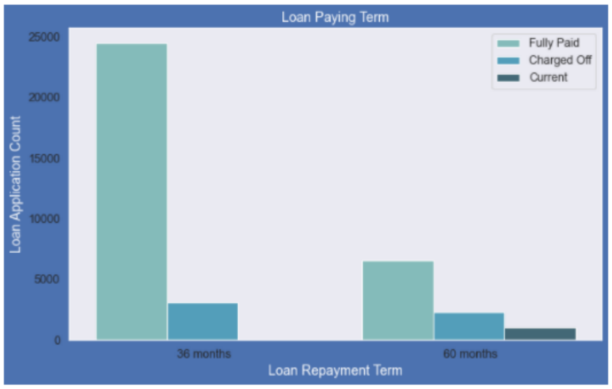


Observations: Above chart shows us count of loan application is increasing every passing year.so increase in number of loan applications are adding more to number of charged off applications. And number of loans issued in 2008(May-October) got dipped, may be due to Recession.

Loan Repayment:

```
# Univariate Analysis - Ordered Categorical Variables- Loan Paying Term
print("Ordered categorical variable")
plt.figure(figsize=(10,6),facecolor='b')
ax = sns.countplot(x="term",data=loan,hue='loan_status',palette='GnBu_d')
ax.set_title('Loan Paying Term',fontsize=14,color='w')
ax.set_xlabel('Loan Repayment Term',fontsize=14,color = 'w')
ax.set_ylabel('Loan Application Count',fontsize=14,color = 'w')
ax.legend(bbox_to_anchor=(1, 1))
plt.show()

# Observations :
# Below plot shows that those who had taken loan to repay in 60 months had more % of number of applicants getting
# charged off as compared to applicants who had taken loan for 36 months.
```

```
Ordered categorical variable
```



<u>Observations:</u> Above plot shows that those who had taken loan to repay in 60 months had more % of number of applicants getting. And charged off as compared to applicants who had taken loan for 36 months.

➢ <u>Segmented Univariate Analysis:</u>

1. Taking subset of data.
2. Identifying groups and subgroups.
3. We need to analyze and choose the right data which is best for our business.
4. We need to summarize using mean, median etc.. and compare aggregated metric across the groups.

➢ <u>Bivariate Analysis:</u>

we are analyzing loan status against some important columns which might have played important role in charged off of loans. Let's try to find proportion of charged offs in some categories.

To calculate the proportion of charged off loans, will do these below steps:

1. Group loans by different variables and loan status, get the count,use .unstack() to return a DataFrame.
2. Since we're going to use this for further analysis, it will be helpful to use. reset_index() to clean up the index.
3. Assign the new DataFrame to a variable.
4. Create a simple derived column that sums charged off,current and fully paid loans for each category.
5. Divide the number of charged off loans by the total number of loans to get the proportion of charged off loans.
6. Store this as another derived column.

7. Finally, output the whole table, sorted by proportion of charged off in each category issued

1. **Annual Income Vs Charged off proportion:**

```
#Bivariate Analysis : Part 1
#Bivariate Analysis :
# In this part of analysis, lets try to analyze the loan status against some important columns which might have played
# important role in charged off of loans. Lets try to find proportion of charged offs in some categories.
# To calculate the proportion of charged off loans, will do these below steps:
# Group loans by different variables and loan status, get the count,use .unstack() to return a DataFrame.
# Since we're going to use this for further analysis, it will be helpful to use .reset_index() to clean up the index.
# Assign the new DataFrame to a variable.
# Create a simple derived column that sums charged off,current and fully paid loans for each category.
# Divide the number of charged off loans by the total number of loans to get the proportion of charged off loans.
# Store this as another derived column. # Finally, output the whole table, sorted by proportion of charged off in each category
#issued loans.

# Bivariate Analysis on annual income against Chargedoff_Proportion
print("Income aginst Chargedoff_Propotion\n")

inc_range_vs_loan = loan.groupby(['annual_inc_cats', 'loan_status']).loan_status.count().unstack().fillna(0).reset_index()
inc_range_vs_loan['Total'] = inc_range_vs_loan['Charged Off'] + inc_range_vs_loan['Current'] + inc_range_vs_loan['Fully Paid']
inc_range_vs_loan['Chargedoff_Proportion'] = inc_range_vs_loan['Charged Off'] / inc_range_vs_loan['Total']
inc_range_vs_loan.sort_values('Chargedoff_Proportion', ascending=False)

# Observations:
# Income range 80000+  has less chances of charged off.
# Income range 0-20000 has high chances of charged off.
# Notice that with increase in annual income charged off proportion got decreased.
```

```
Income aginst Chargedoff_Propotion
```

loan_status	annual_inc_cats	Charged Off	Current	Fully Paid	Total	Chargedoff_Proportion
0	0-20000	237	9	943	1189	0.20
1	20000-40000	1514	170	7004	8688	0.17
2	40000-60000	1729	345	9534	11608	0.15
3	60000-80000	1024	240	6597	7861	0.13
4	80000 +	1122	376	8859	10357	0.11

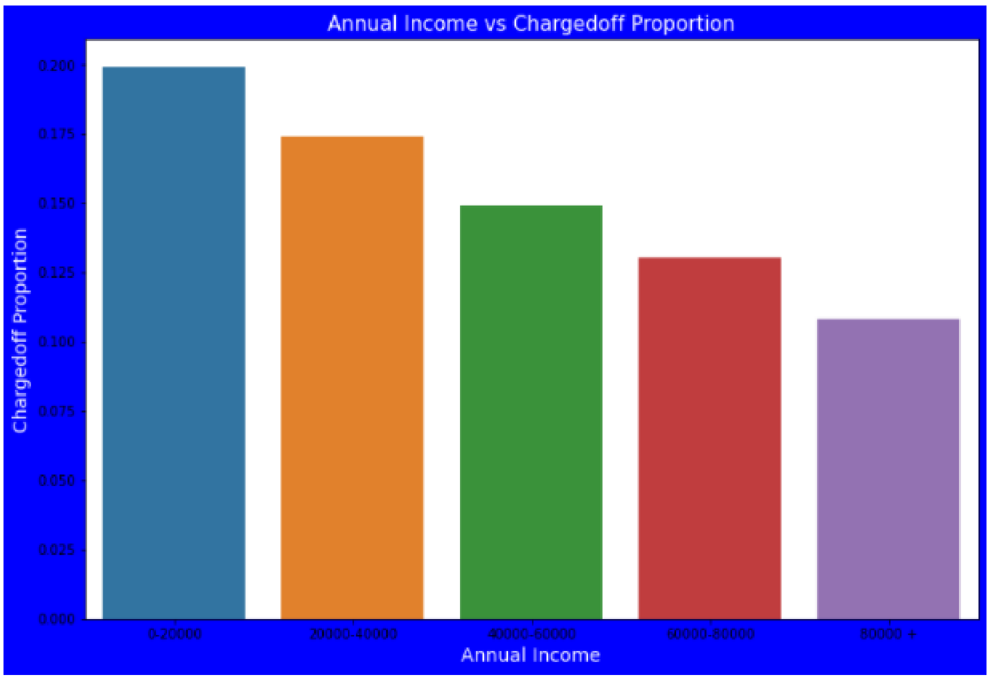
Observations:

1. Income range 80000+ has less chances of charged off.
2. Income range 0-20000 has high chances of charged off.
3. Notice that with increase in annual income charged off proportion got decreased.

Let's draw bar plots on data calculated above, to visualize the pattern to understand the data better.

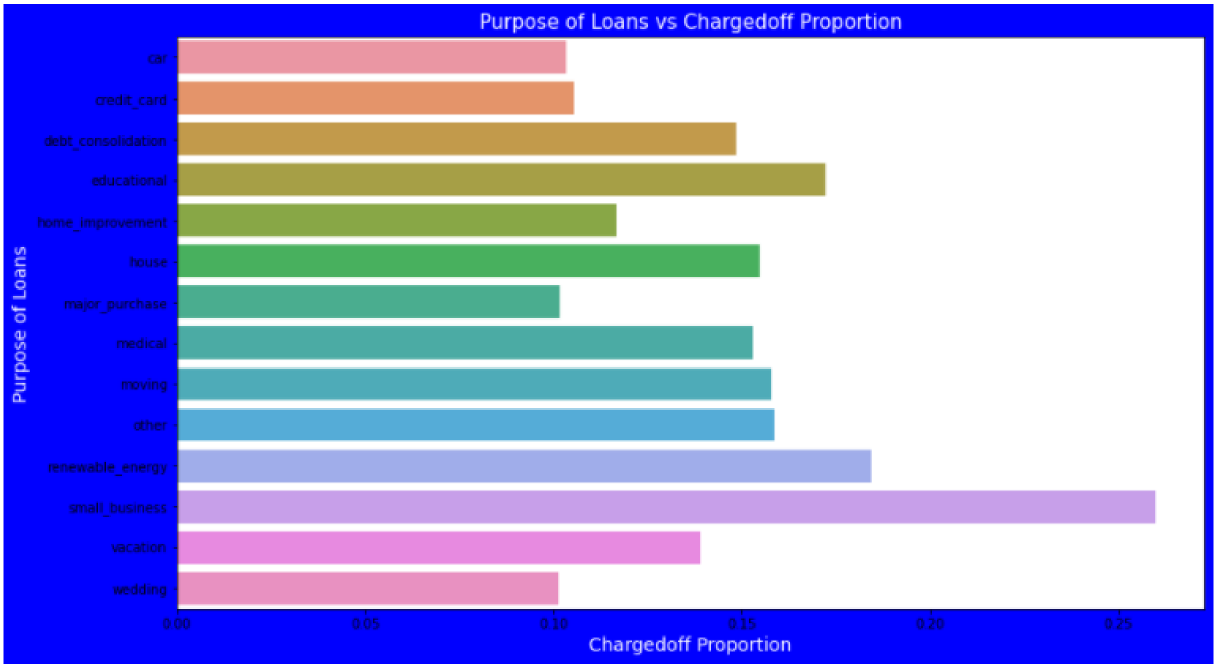
```python
# Lets draw bar plots on data calculated above. Try to visualize the pattern to understand the data better.
print ("Annual income VS Charged Off Propotion\n")
fig, ax1 = plt.subplots(figsize=(12, 8),facecolor='b')
ax1.set_title('Annual Income vs Chargedoff Proportion',fontsize=15,color = 'w')
ax1=sns.barplot(x='annual_inc_cats', y='Chargedoff_Proportion', data=inc_range_vs_loan)
ax1.set_ylabel('Chargedoff Proportion',fontsize=14,color = 'w')
ax1.set_xlabel('Annual Income',fontsize=14,color='w')
plt.show()

# Observations:
# Income range 80000+  has less chances of charged off.
# Income range 0-20000 has high chances of charged off.
# Notice that with increase in annual income charged off proportion got decreased.
```

```
Annual income VS Charged Off Propotion
```



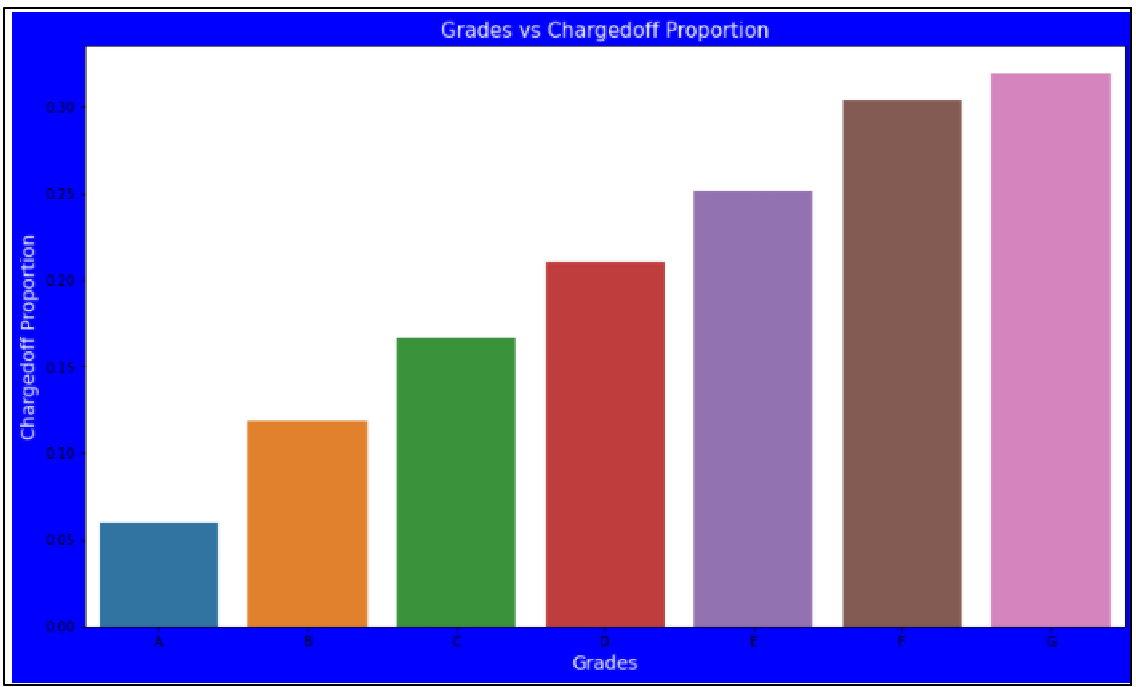
2. <u>Purpose of Loan VS Charge off Proportion:</u>

Purpose of loan VS chargedoff Propotion



Observations: Small Business applicants have high chances of getting charged off. Renewable_energy was charged off proportion is better as compare to other categories.

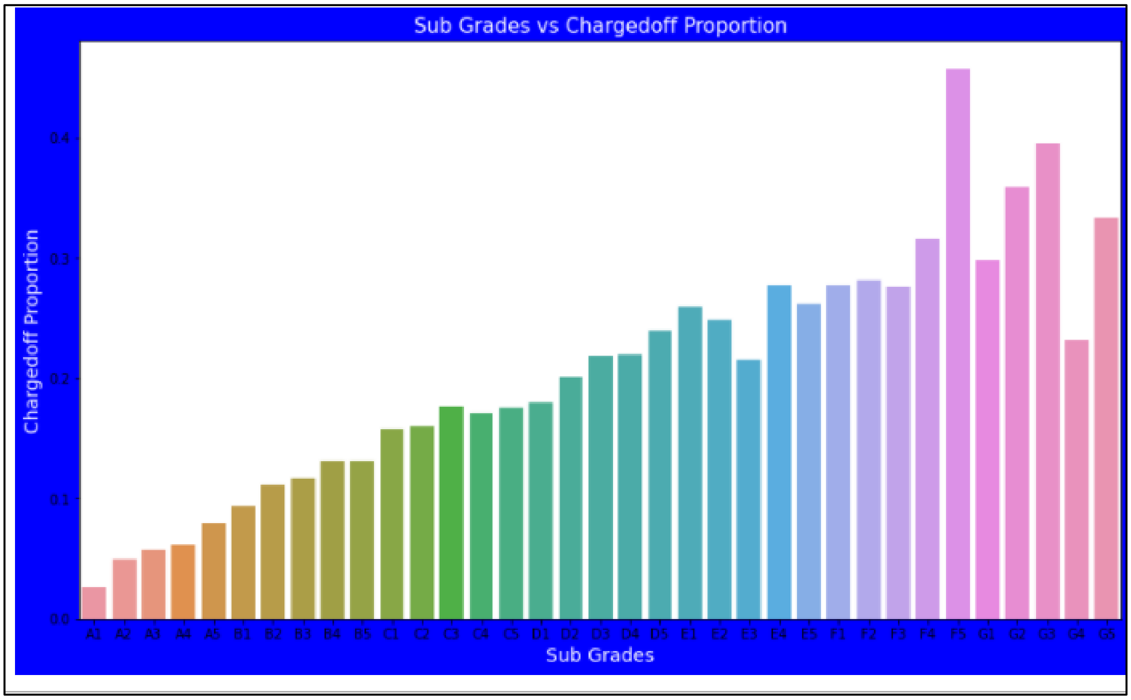
3. **Grade against Charged off Proportion:**



Observations:

- Grade "A" has very less chances of charged off.
- Grade "F" and "G" have very high chances of charged off.
- Chances of charged of is increasing with grade moving from "A" towards "G"

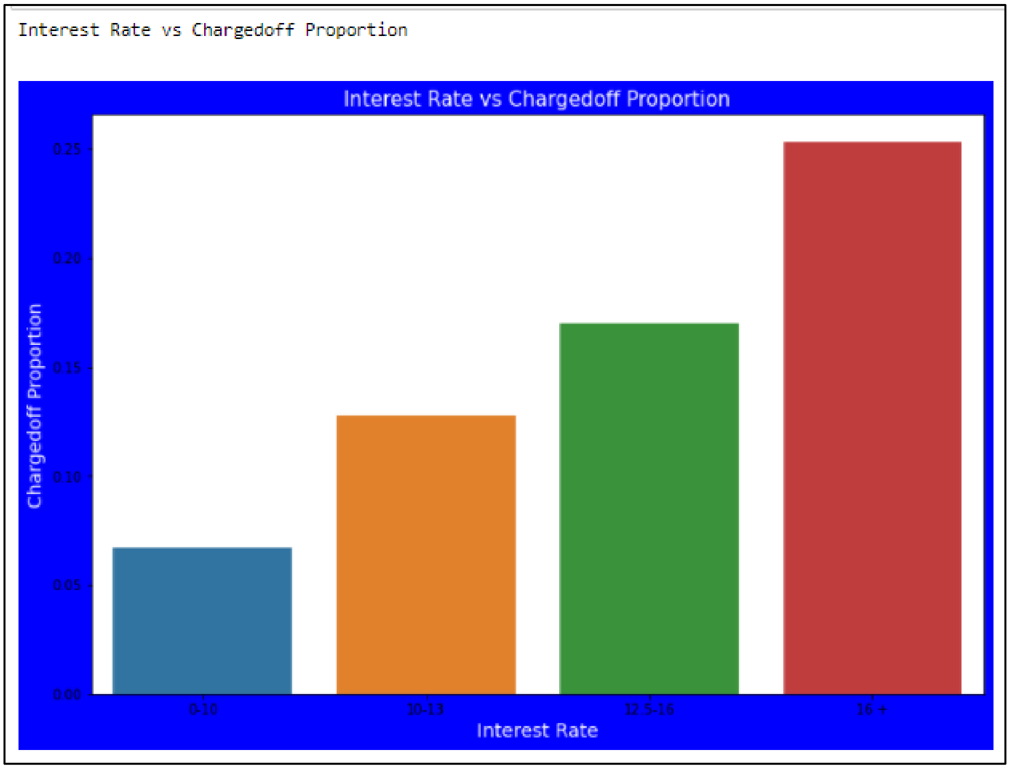
Subgrades Vs Charged off proportion



Observations:

1. Sub-Grades of "A" has very less chances of charged off.
2. Sub Grades of "F" and "G" have very high chances of charged off.
3. Proportion of charged off is increasing with sub grades moving from sub grades of "A" towards sub grades of "G"

Interest Rate vs Charged off Proportion



Observations:

- Interest rate less than 10% has very less chances of charged off. Interest rates are starting from minimum 5 %.
- interest rate more than 16% has good chances of charged off as compared to other category interest rates.
- Charged off proportion is increasing with higher interest rates.

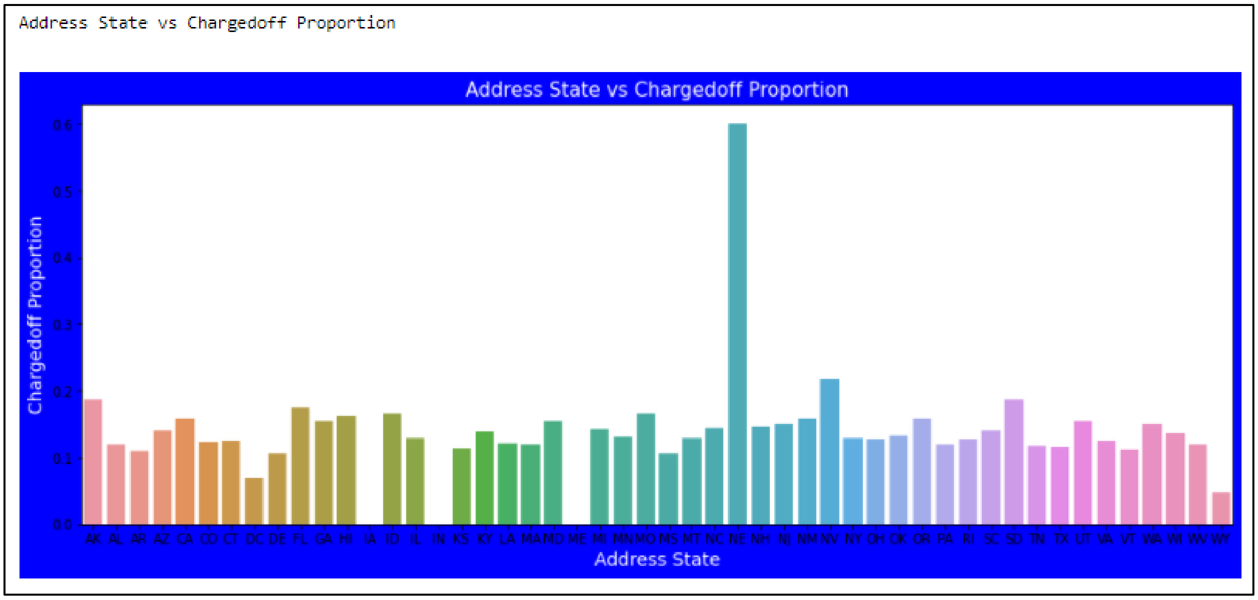
EmpoymnetLength VS Chargedoff Proportion:

EmpoymnetLength VS Chargedoff Proportion

Observations:

- Those who are not working or have less than 1 year of work experience have high chances of getting charged off.
- It makes sense as with less or no experience they don't have source of income to repay loan.
- Rest of the applicants have more or less same chances of getting charged off.

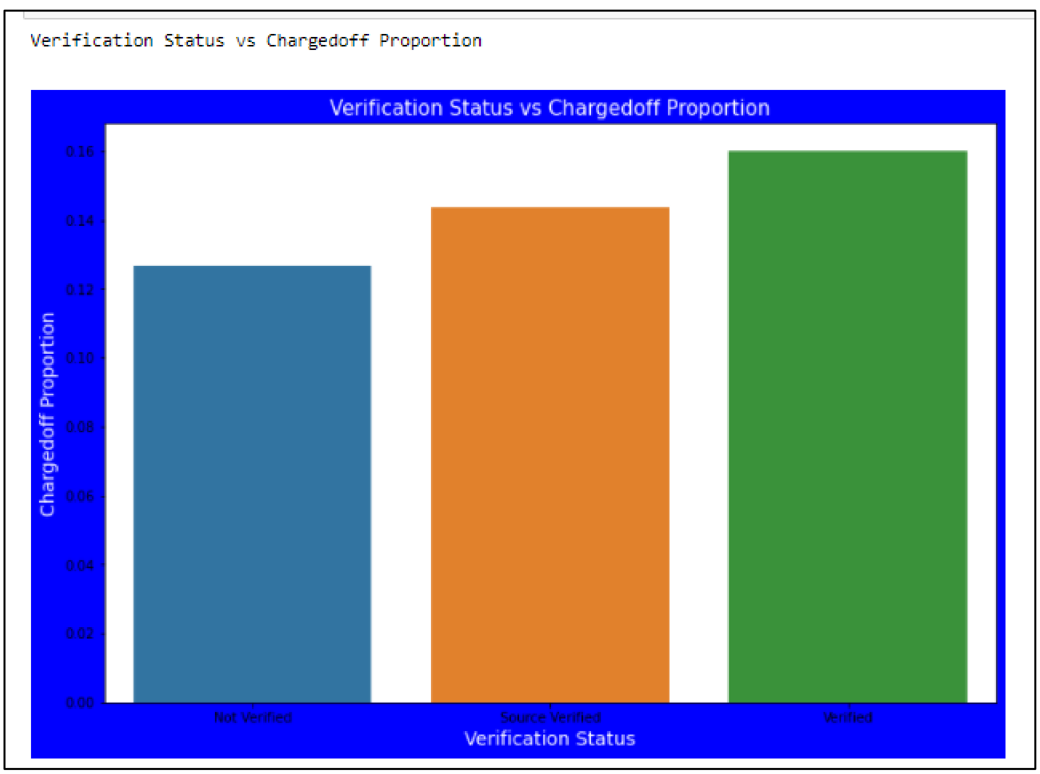
Address State VS Charged off Proportion:



Observations:

- states NE has very high chances of charged off but number of applications are too low to make any decisions.
- NV,CA and FL states shows good number of charged offs in good number of applications.

Verification Status vs Chargedoff Proportion:

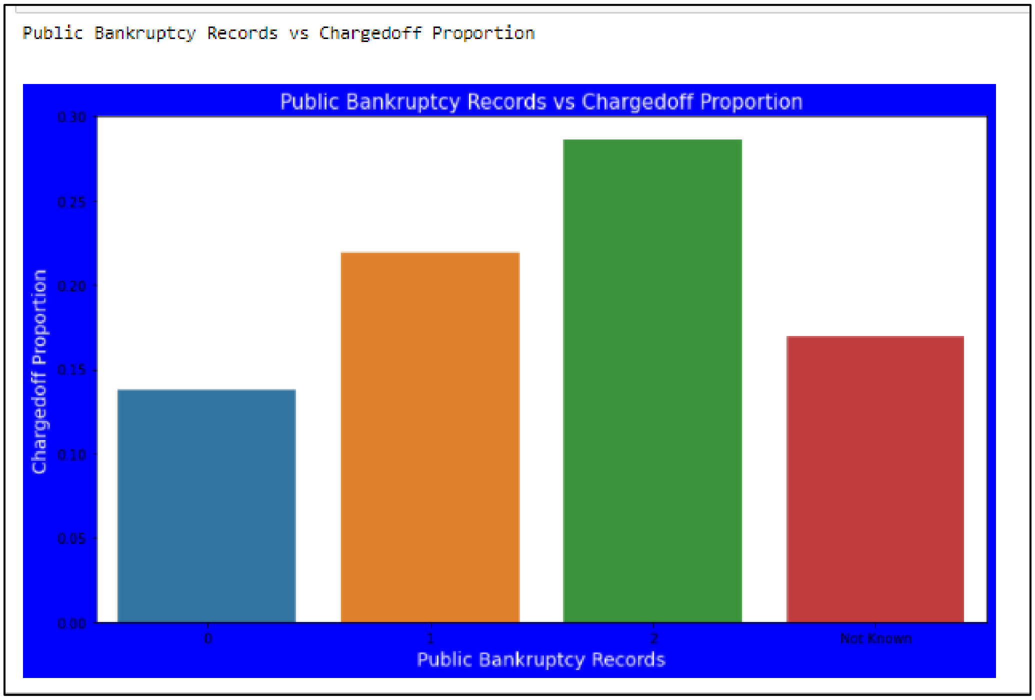


Observations:

- There is not much difference in charged off proportion.

- This variable doesn't provide any insights for charged off.

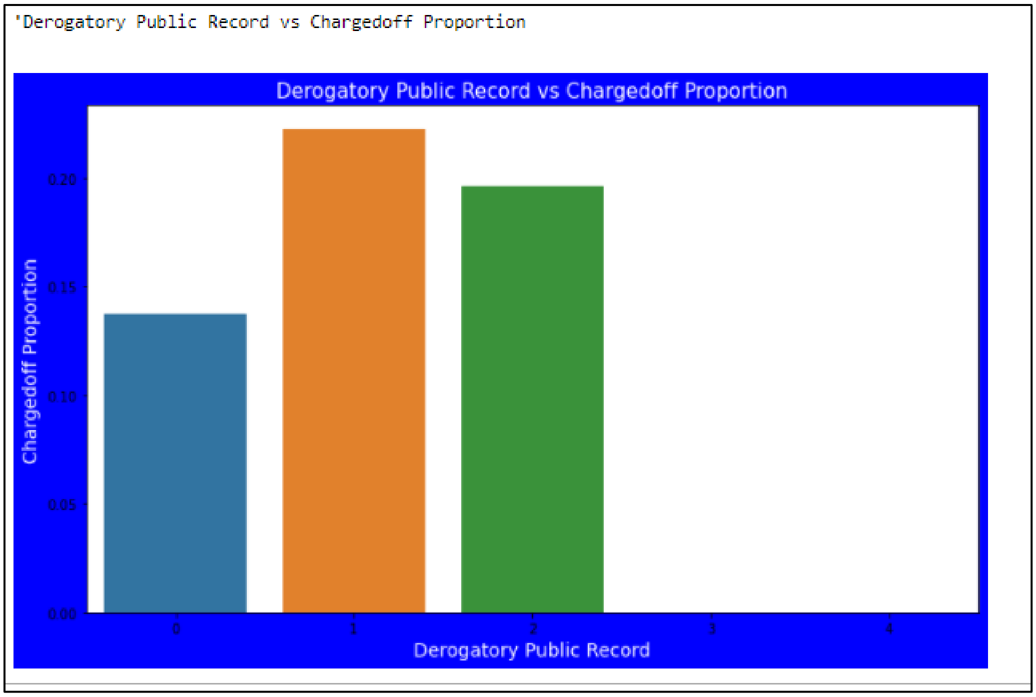
Public Bankruptcy Records vs Chargedoff Proportion:



Observations:

- Those who already have pub_rec_bankruptcies value 1, have charged off proportion higher than who have no pub_rec_bankruptcies.
- pub_rec_bankruptcies count 2 has even higher charged off proportion but those numbers are not significant to decide.
- Not known is the column for which we don't have any information about borrower.
- This also makes sense that who has defaulted before has more chances of defaulting in future as well.

Derogatory Public Record vs Chargedoff Proportion:



Observations:

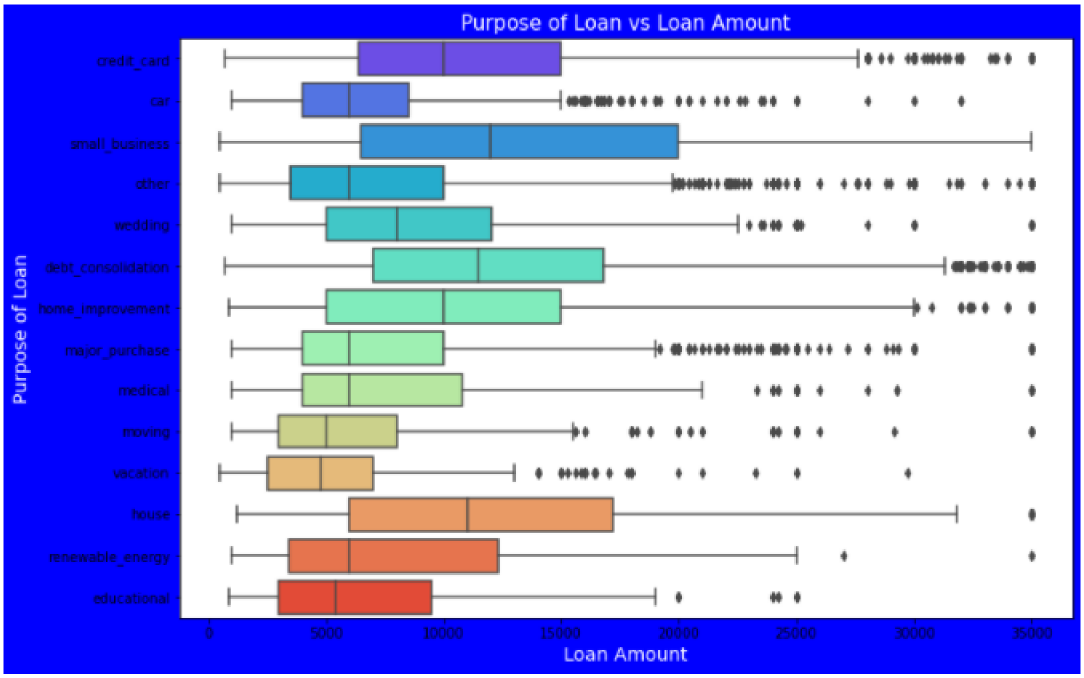
- A derogatory item is an entry that may be considered negative by lenders because it indicates risk and hurts
- your ability to qualify for credit or other services. Public records and collections are derogatory items
- Because they reflect financial obligations that were not paid as agreed.
- Those who already have pub_rec value 1 or 2 have charged off chances higher than who have no Derogatory Public Record.
- pub_rec count 3-4 has less numbers so cannot reach on any conclusions.

➢ **Bivariate Analysis - Part 2**

Puprose of loan vs Loan amount:

Box Plot:

Puprose of loan vs Loan amount



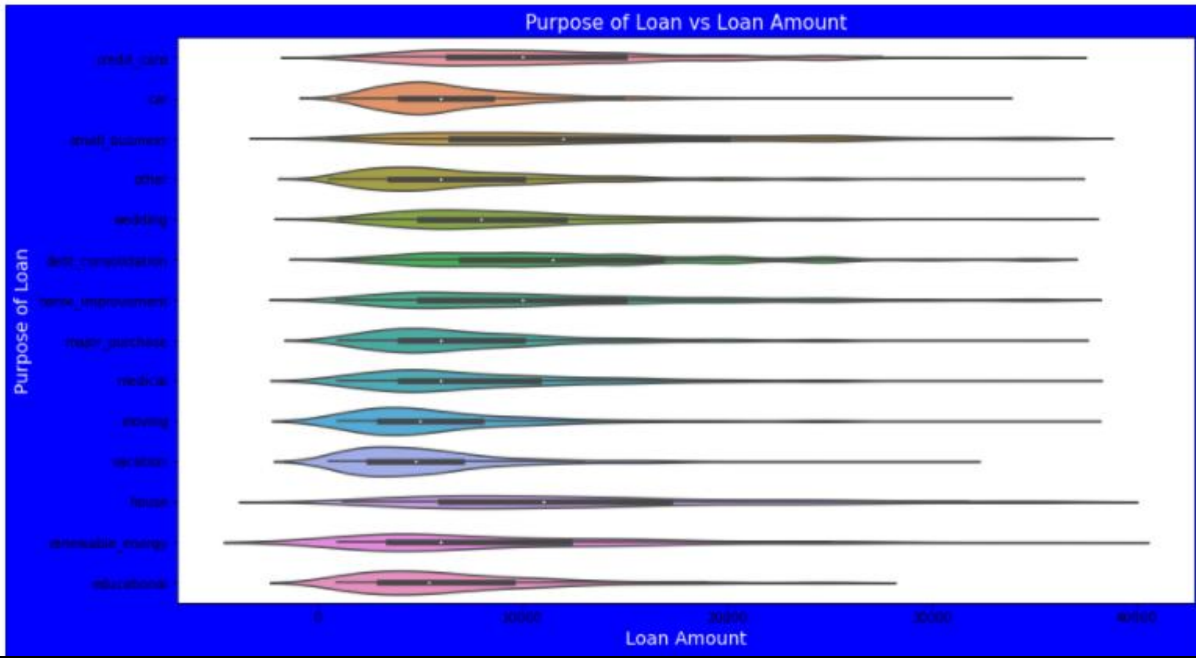
Observations:

- Median,95th percentile,75th percentile of loan amount is highest for loan taken for small business purpose among all purposes.
- Debt consolidation is second and Credit card comes 3rd.

violin Plot –

It shows the distribution of quantitative data across several levels of one (or more) categorical variables such that those distributions can be compared.

Puprose of loan vs Loan amount

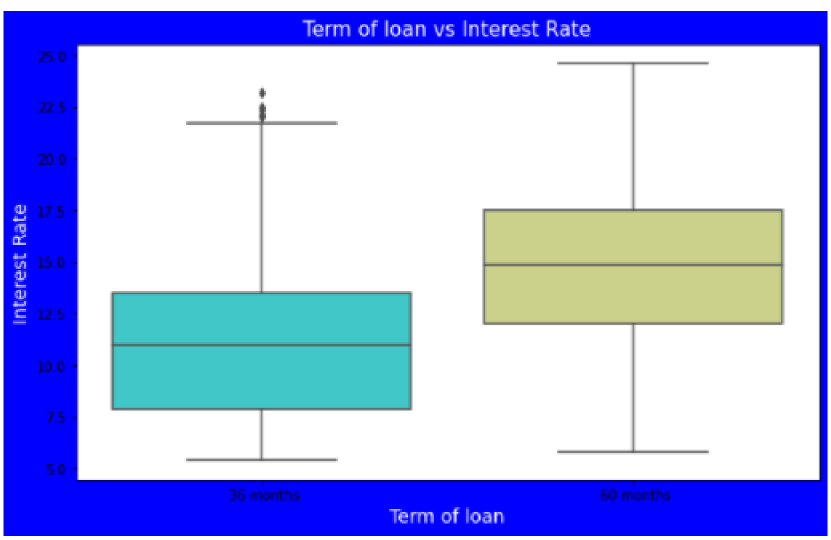


Observations:

- Loan taken for small business purpose, Debt consolidation and Credit card are somewhat evenly distributed
- As compare to loan taken for other purposes.

Term of loan vs Interest Rate:

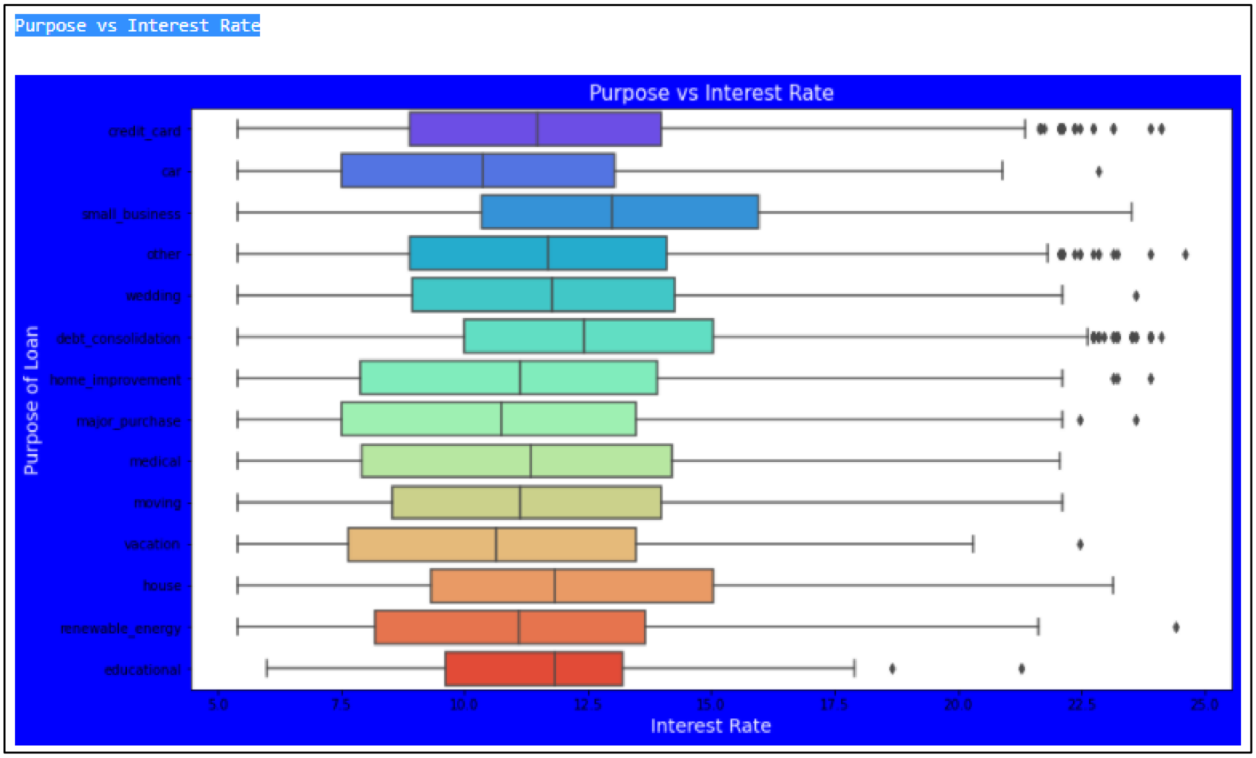
Term of loan vs Interest Rate



Observations:

- It is clear that average interest rate is higher for 60 months loan term.
- Most of the loans issued for longer term had higher interest rates for repayment.

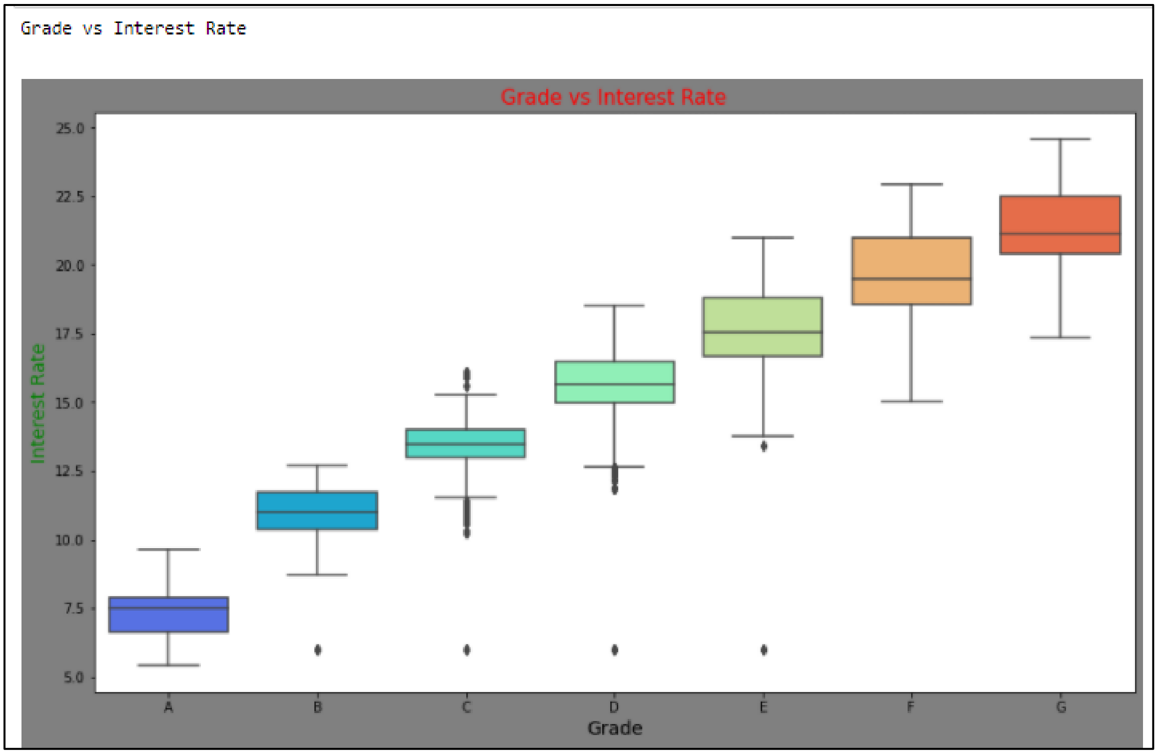
Purpose vs Interest Rate:



Observations:

- It is clear that average interest rate is highest for small business purpose.
- Loans taken for small business purposes had to repay the loan with more interest rate as compared to other.
- Debt consolidation is 2nd where borrowers had to pay more interest rate.

Grade vs Interest Rate



Observations:

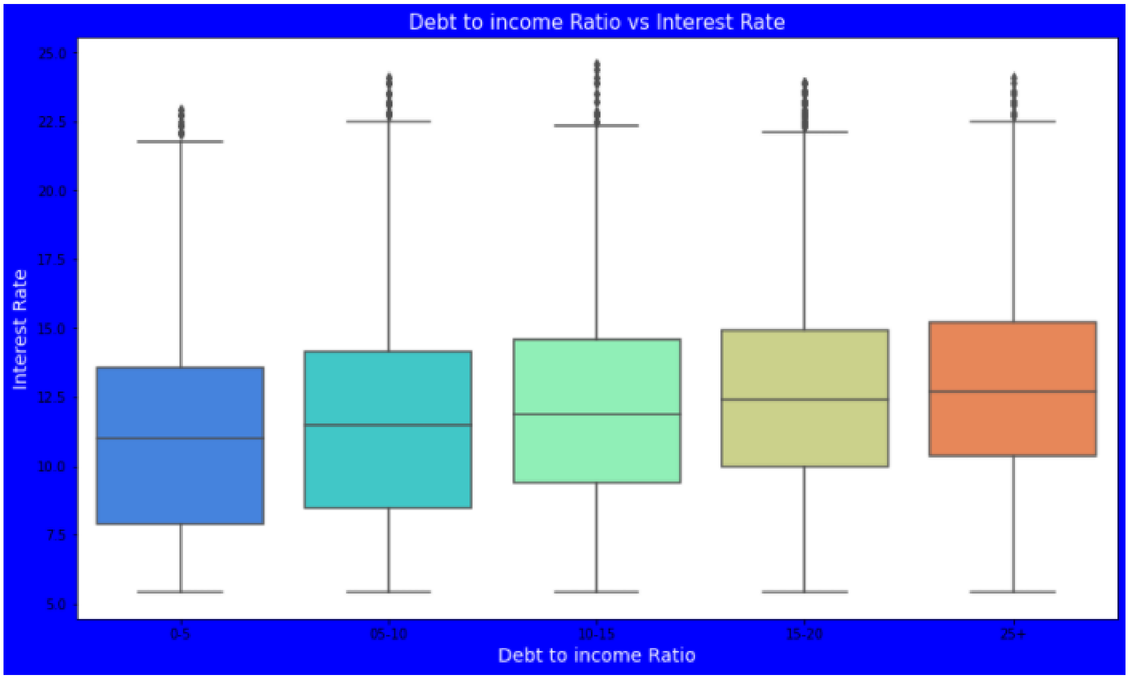
- A-grade is a top letter grade for a lender to assign to a borrower.
- The higher the borrower's credit grade, the lower the interest rate offered to that borrower on a loan.
- It is clear that interest rate is increasing with grades moving from A to F.

dti vs Interest Rate:

Interest rate is increasing with loan amount increase.

Probably when loan amount is more it is taken for longer loan term, we saw earlier that longer the loan term more the interest rate.

dti vs Interest Rate

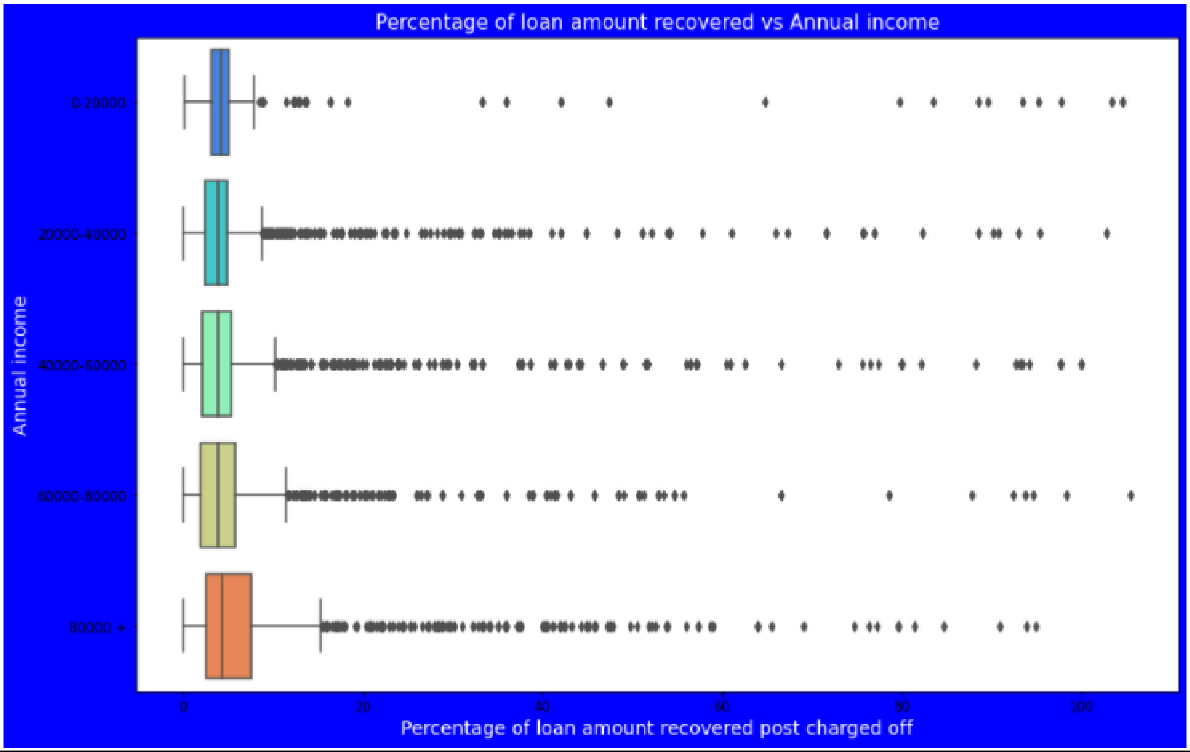


Observations:

- If your DTI is low enough you may get a lower interest rate.
- Plot shows no significant variation but there is slight increase in interest rate with increase in DTI.

Recoveries vs Annual income

Recoveries vs Annual income

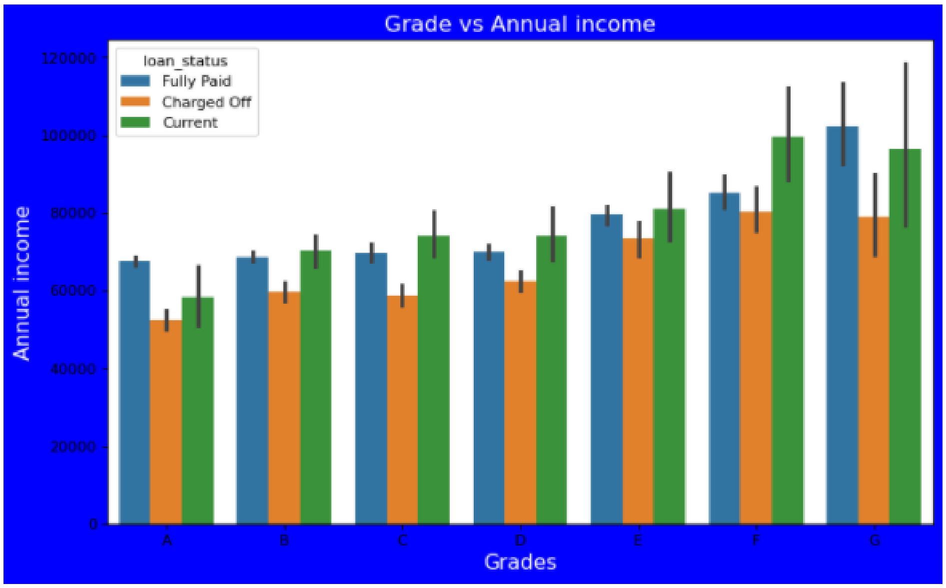


Observations:

- Higher percentage of loan amount is recovered when annual income is high.
- Plot shows no significant variation but there is slight increase in recovery percentage with increase in annual income.

Variation of annual income across grade for every loan status:

Variation of annual income across grade for every loan status

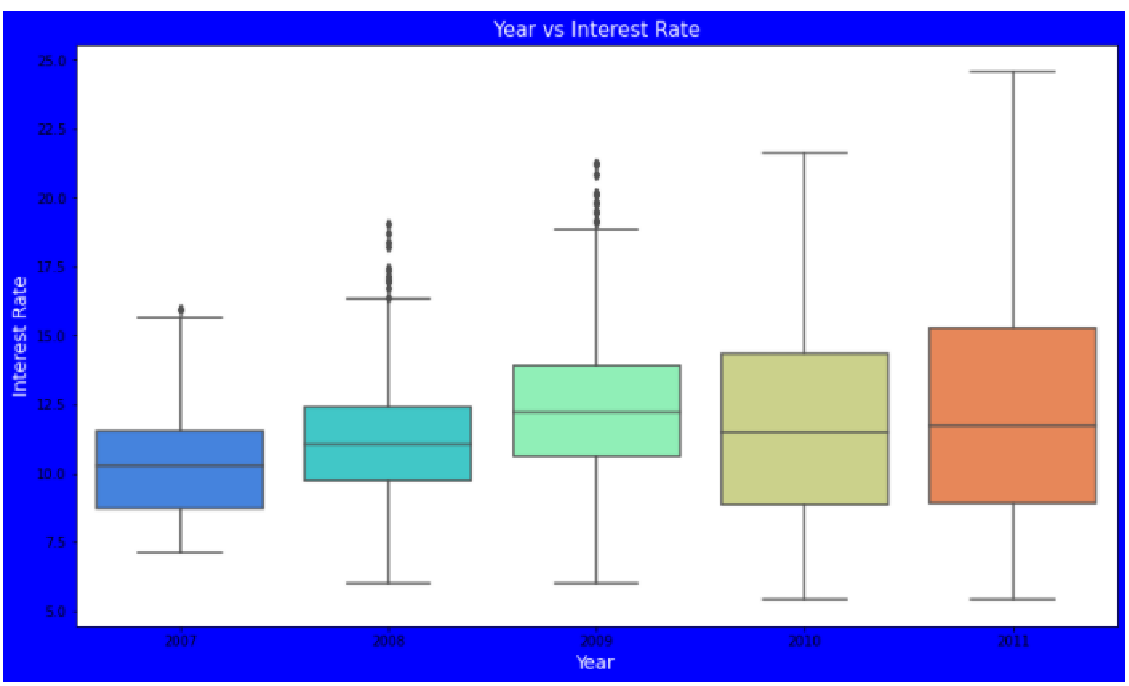


Observations:

- From this we can conclude that the ones getting 'charged off' have lower annual incomes than the ones who 'paid fully' for each and every grade (i.e. at same interest range)

year vs Interest Rate:

Analysis - year vs Interest Rate



Observations: Plot shows interest rate is increasing slowly with increase in year.

➢ Multivariate Analysis

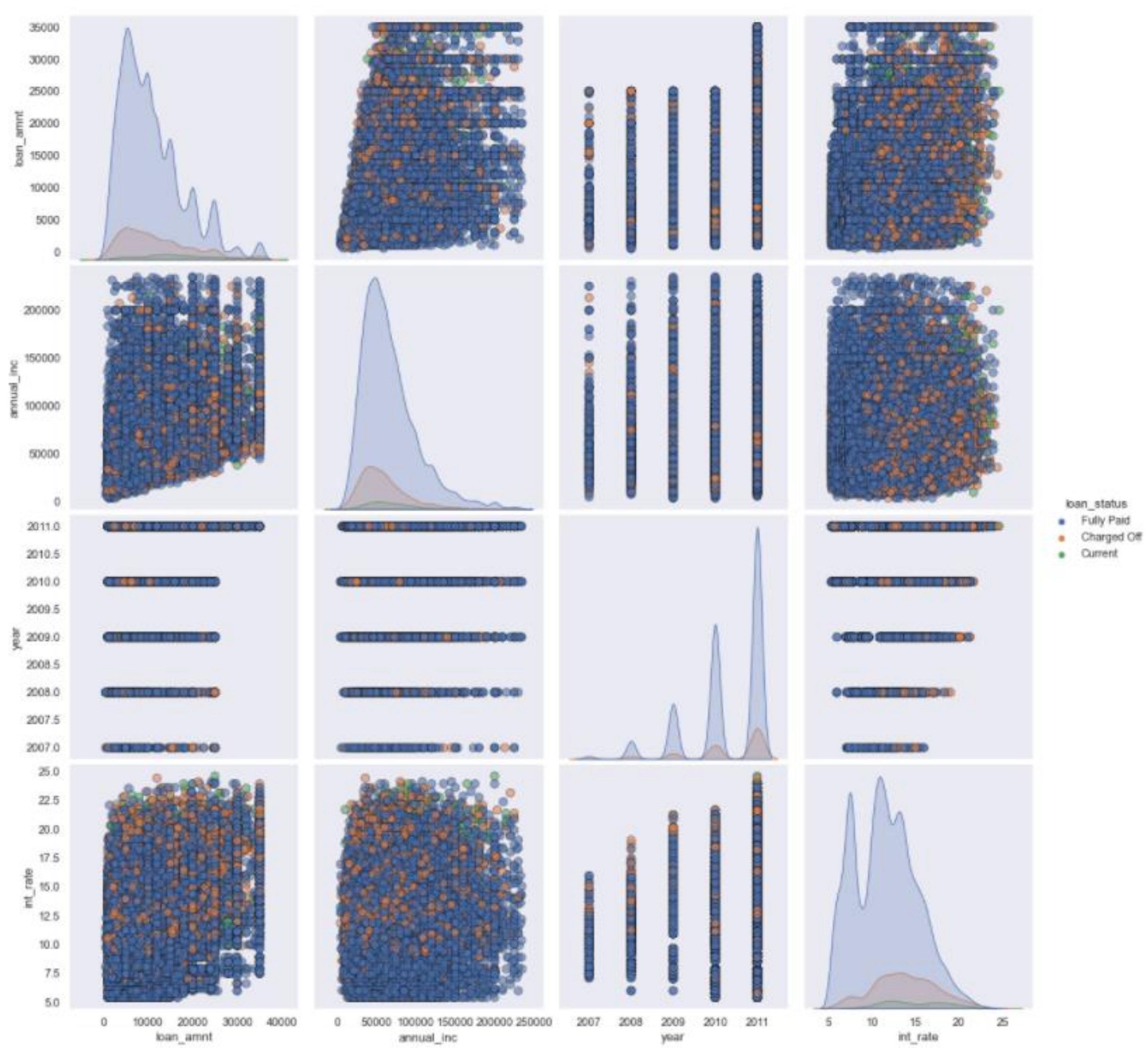
Pair Plots: Lets draw a pair plot for multivariate analysis and pick few important columns from dataset separately.

Lets draw a pair plot for multivariate analysis

	loan_amnt	annual_inc	year	int_rate	loan_status
0	5000	24000.00	2011	10.65	Fully Paid
1	2500	30000.00	2011	15.27	Charged Off
2	2400	12252.00	2011	15.96	Fully Paid
3	10000	49200.00	2011	13.49	Fully Paid
4	3000	80000.00	2011	12.69	Current

Loan Status

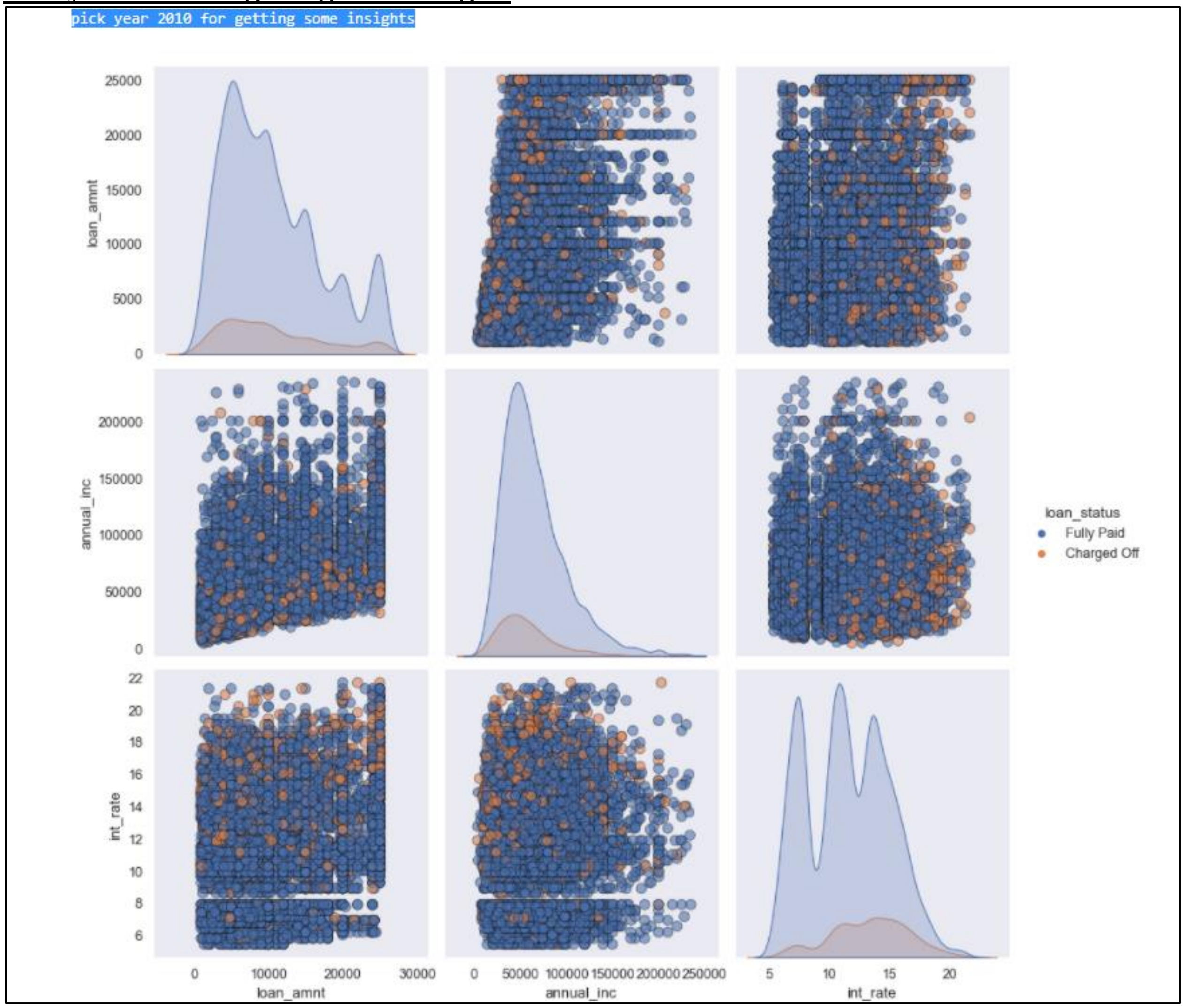
loan Status



Observations:

- Higher the interest rate higher charged off ratio
- Higher the annual income higher the loan amount slightly.
- Increase in number of charged off with increase in year.
- Interest rate is increasing with loan amount increase

<u>Pick year 2010 for getting some insight:</u>



<u>Observations:</u>

- Higher the interest rate higher charged off ratio
- Higher the annual income higher the loan amount slightly
- Interest rate is increasing with loan amount increase this results in high charged off.

➢ <u>Summary:</u>

<u>Based on the Analysis done on the Variables, we conclude the below mentioned points:-</u>

➢ Small Business Applicants have high chances of getting charged off.

- Charged off proportion increases with grades moving from “A” towards “G”.
- Charged off proportion increases as Interest Rate Increases.
- Higher the public bankruptcy record greater the charged-off proportion.
- The loan amounts are bigger on average for small business purpose among all purposes of Loan.
- Those who already have Derogatory Public Records have higher charged off chances than others.
- Average interest rate is considerably higher for 60 months loan term than 36 months.
- Ones getting charged off have lower annual incomes than the ones who has fully paid for each and every grade.

Suggestions

- Loans for Small Business Applicants should be checked properly.
- Loan approval should be avoided for those who already have Derogatory Public Records.
- Loan approval should be avoided for those who already have Public Bankruptcy Records.
- Loan approval for Low quality loans should be avoided or given for smaller loan repayment term.
- Lower annual income applicants should be avoided for big loan amounts with higher interest Rates.
- Loan approval should be avoided for applicants who doesn’t have a source of income.