LENDING CLUB CASE STUDY

PROJECT ANALYSIS REPORT

Project Team

Group Facilitator: Arjun Khanna

Group Member: Anupam Gogia

EXECUTIVE SUMMARY

Project Goal: Identify target attributes from the loan corpus data and help predict risk score for potential defaulting loan applicants.

Method:

- Use appropriate techniques to cleanse data.
- Prepare data for further analysis.
- Leverage EDA techniques (univariate, bivariate, derived metrics) to analyse data and determine influencing variables in loan processing.
- After data cleaning, identified 22 features that has high impact on determining defaulting loan applicants.
- **Recommendation(s)**: Using EDA analysis, identified following 8 features sets that determines potential defaulting loan applicants.
  - [funded_amnt (or derived column funded_amnt_bucket), int_rate (or derived column int_rate_bucket), Dti (or derived column int_rate_bucket), open_acc (or derived column open_acc_bucket), purpose, term, grade, addr_state]

DATA CLEANSING

APPROACHES USED TO REMOVE UNNECESSARY COLUMNS:

- Features with high NA values & non-business critical columns
- Features with payment details
- Data with poorly distributed values which can't be categorized for meaningful analysis

Reduced # of columns from 111 to 22 for further analysis

- ***Desc, URL, delinq_amnt, acc_now_delinq, application_type, policy_code, pymnt_plan, chargeoff_within_12_mths, collections_12_mths_ex_med, tax_liens, initial_list_status, emp_title, recoveries, collection_recovery_fee, delinq_2yrs*** *columns are dropped because it does not have any useful information, has null values in the maximum # of rows or values not well distributed.*

After dropping these features, 39 features remained for further analysis

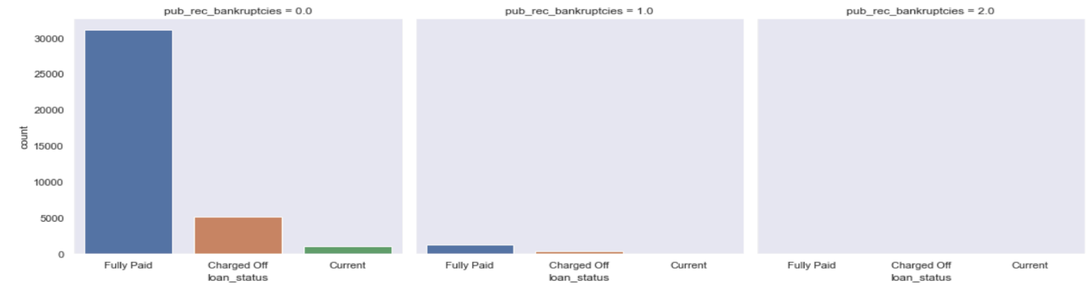
- *The "**out_prncp, out_prncp_in, total_pymnt, total_pymnt_inv, total_rec_prncp, total_rec_int, total_rec_late_fee, loan_amnt, zip_code, revol_bal, last_pymnt_d, last_pymnt_amnt, last_credit_pull_d, member_id**" features provide repetitive information and payment information of applicants, and hence the same is dropped for further analysis*

After dropping these features, 25 features remained for further analysis

DATA CLEANSING (CONTD.)

- pub_rec_bankruptcies, title, pub_rec are dropped based on below analysis:

```
catplot = sns.catplot(x='loan_status',col='pub_rec_bankruptcies',data=loan,kind='count')
plt.show()
```




```
print(loan.groupby('pub_rec')['loan_status'].count())
```

```
pub_rec
0    37601
1     2056
2       51
3        7
4        2
Name: loan_status, dtype: int64
```

Based on the above plots and exploring title and pub_rec column values, they found to be not helpful for the analysis. Also, purpose column too provides similar information to title column.

Note: 'pub_rec_bankruptcies' is removed because bankruptcies should directly map to charge-off as it indicates the history that individual has run into bankruptices but the plots shows the inverse relation with charge-off. 'pub_rec' column values are not distributed and it is skewed with '0'

```
print(loan['title'].head(20))
```

```
0                                Computer
1                                    bike
2                   real estate business
3                                personel
4                                Personal
5     My wedding loan I promise to pay back
6                                    Loan
7                        Car Downpayment
8      Expand Business & Buy Debt Portfolio
9                  Building my credit history.
10                 High intrest Consolidation
11                           Consolidation
12                                 freedom
13                           citicard fund
14                              Other Loan
15                   Debt Consolidation Loan
16                                   Home
17                                 Holiday
18                                 Medical
19       lowerratemeanseasiertogetoutofdebt!
Name: title, dtype: object
```

After dropping these features, 22 features remained for further analysis

DATA PREPARATION

Emp_length column is processed and converted to numerical dataset.

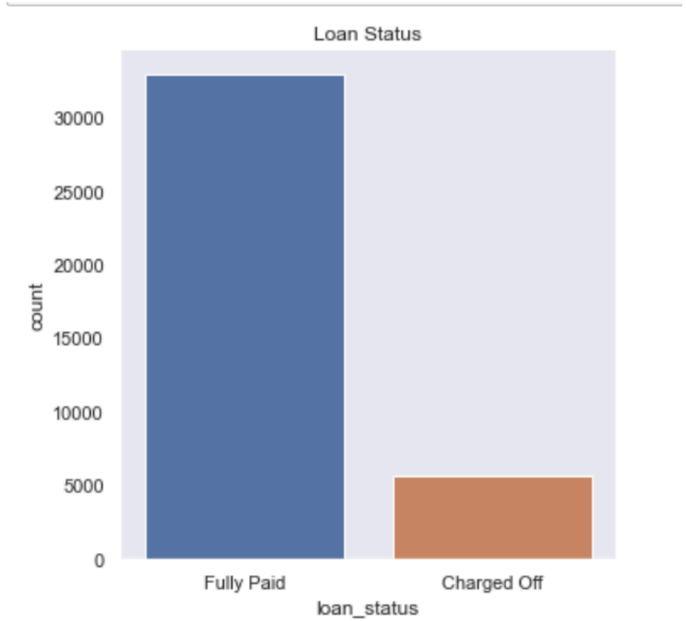
Issue_d column is split into day, month and year values for analysis

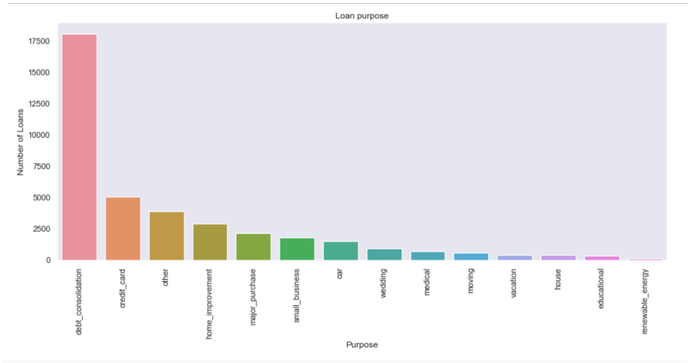
The records with ‘Current’ loan status is not considered for further processing as it does not have any information about potential defaults.

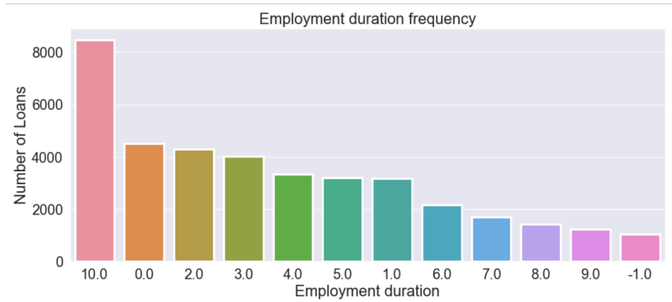
Junk characters (%) are removed from int_rate and revol_util columns

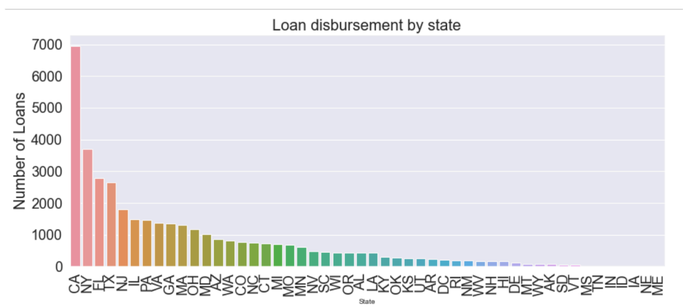
UNIVARIATE: CATEGORICAL ANALYSIS – SAMPLE

CHARTS









UNIVARIATE: CATEGORICAL ANALYSIS - INFERENCE

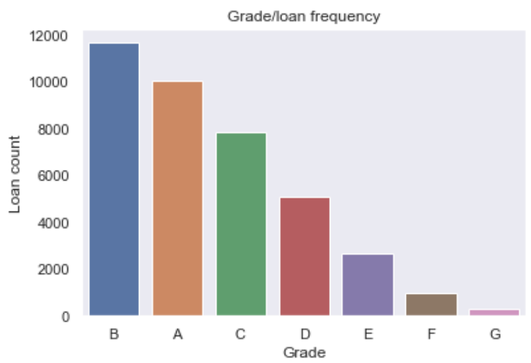
Loan Status: 85% of the total loans are fully paid while only 15% are charged off.

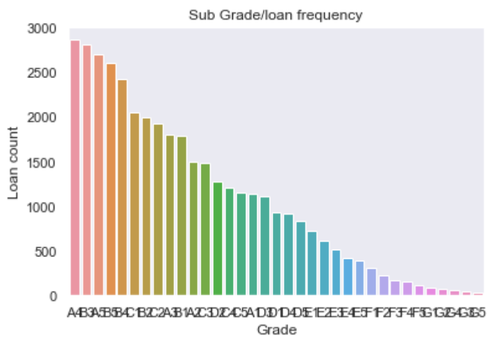
Loan Purpose: Most loans have been taken for debt consolidation.

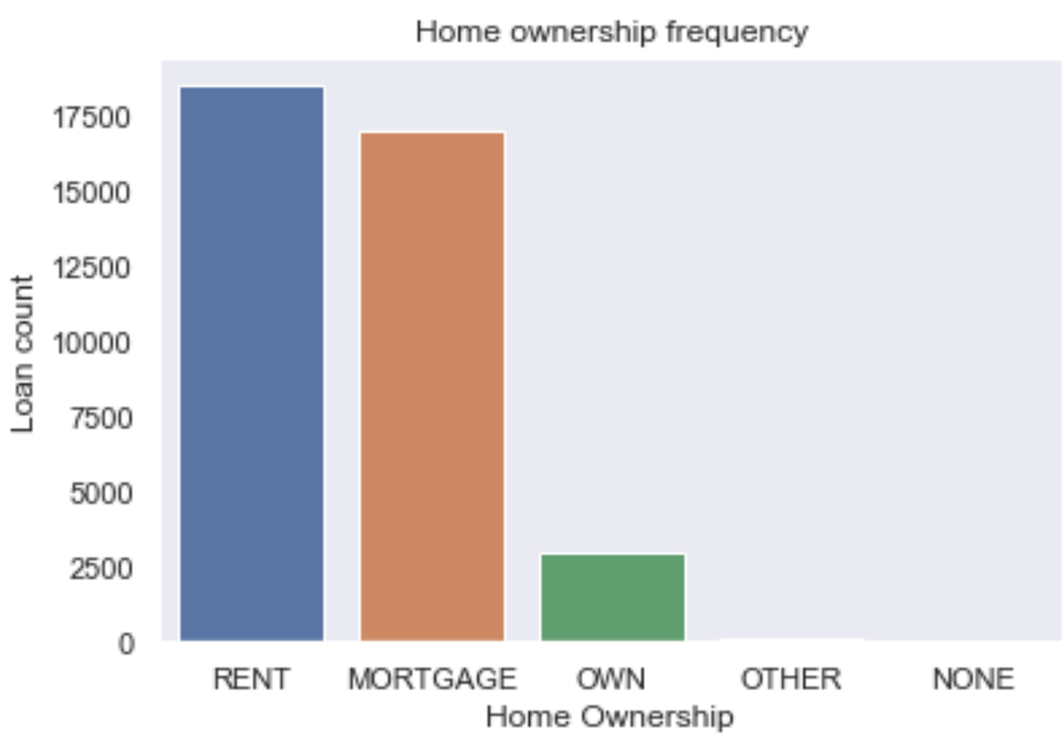
Loan by State: CA, NY, FL, TX and NJ are the top states where maximum number of loans were issued.

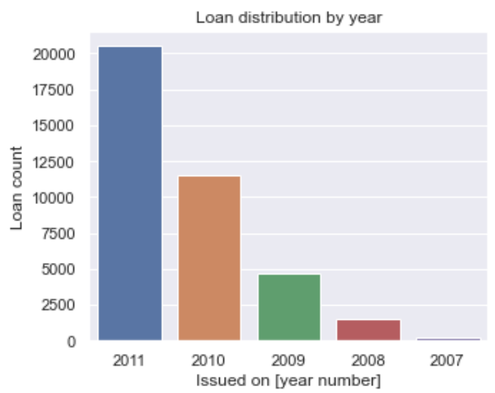
Loan by Employment Duration: People with 10 or more years of experience are the ones who have maximum number of loan accounts, followed by those who have just started their career.

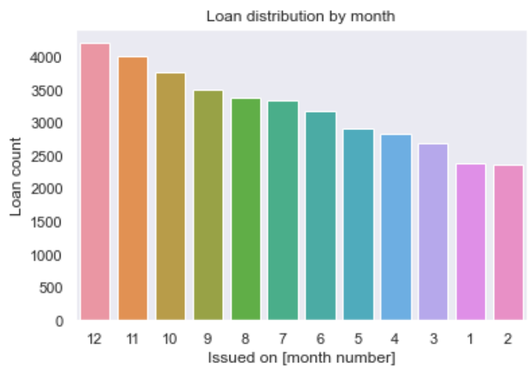
UNIVARIATE: CATEGORICAL ANALYSIS – SAMPLE CHARTS









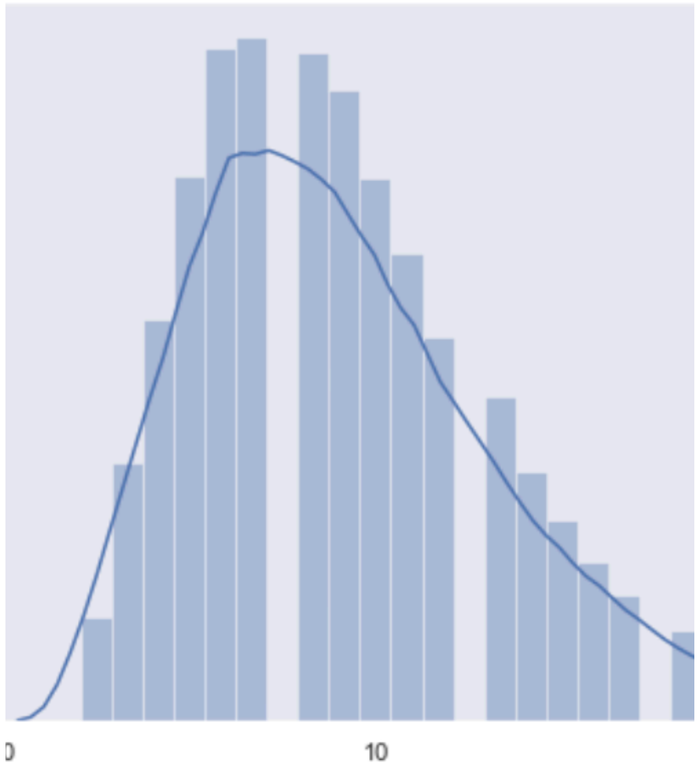


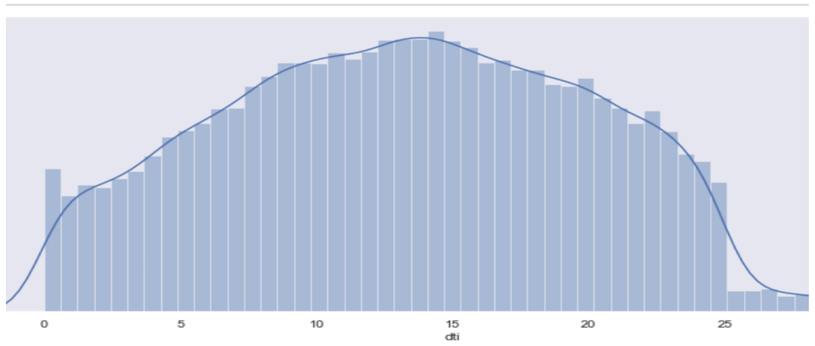
UNIVARIATE: CATEGORICAL ANALYSIS - INFERENCE

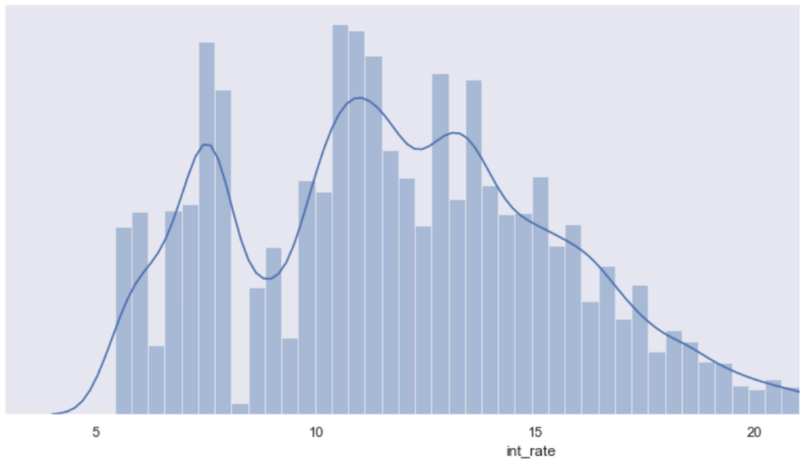
Loans by Grade and subgrade: It is observed that there are more loans in grade B followed by A. However, subgrades loan frequency is slightly different.

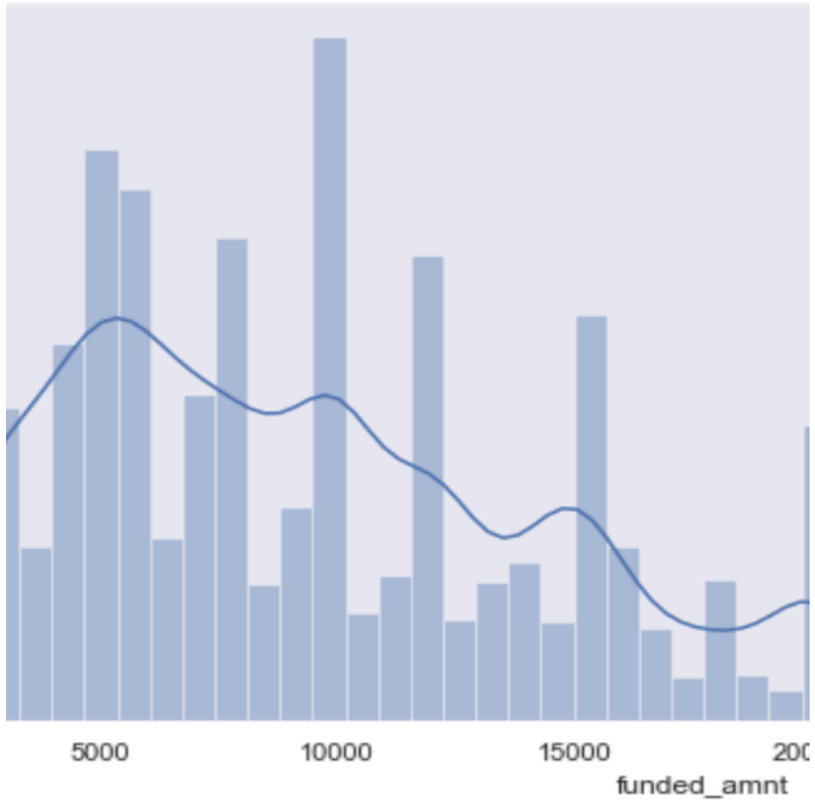
Loan by Home Ownership: There are more loans from those who don't own a home.

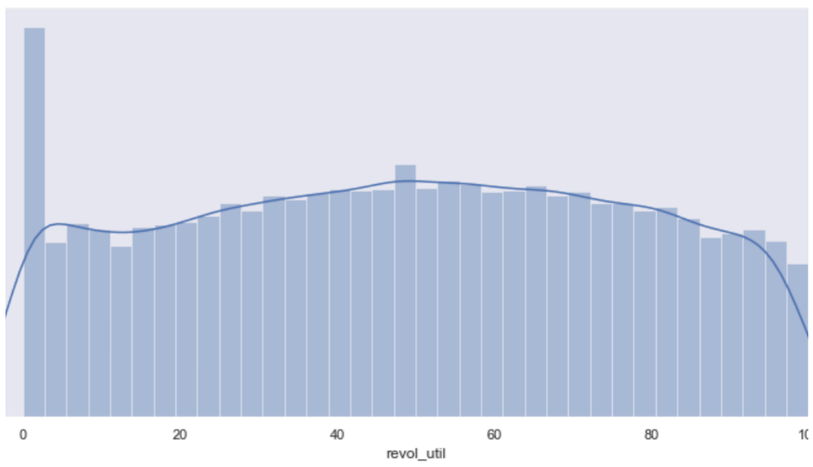
Loan by Year and Month: There is an increasing trend in the number of loans disbursed each year. Most loans are taken around the month of December.





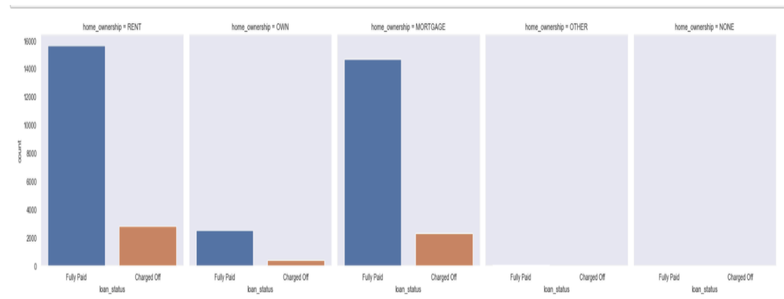


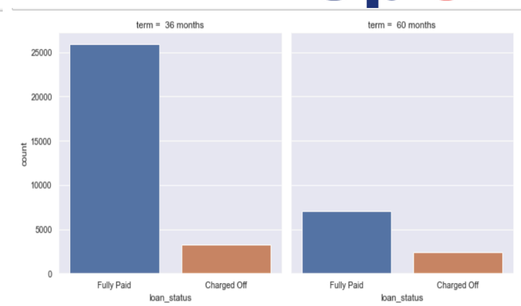


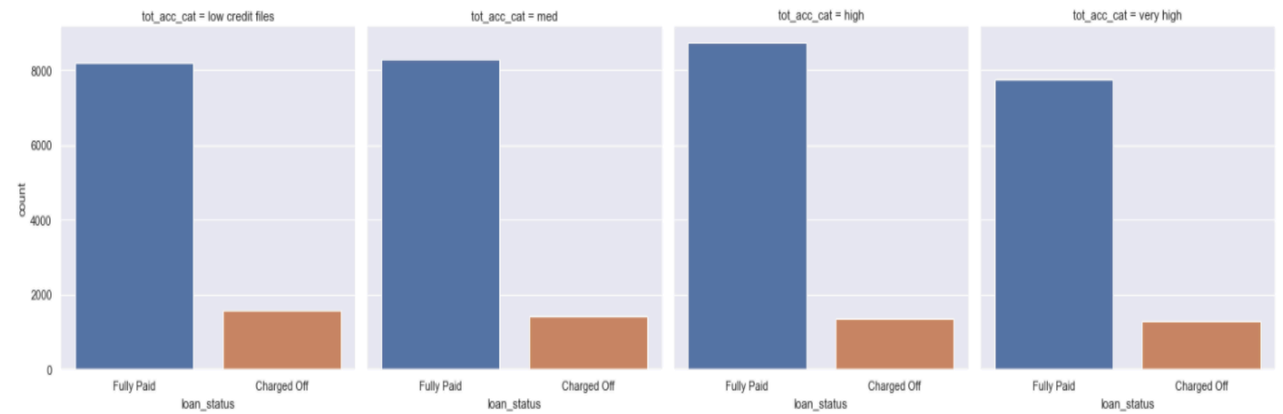


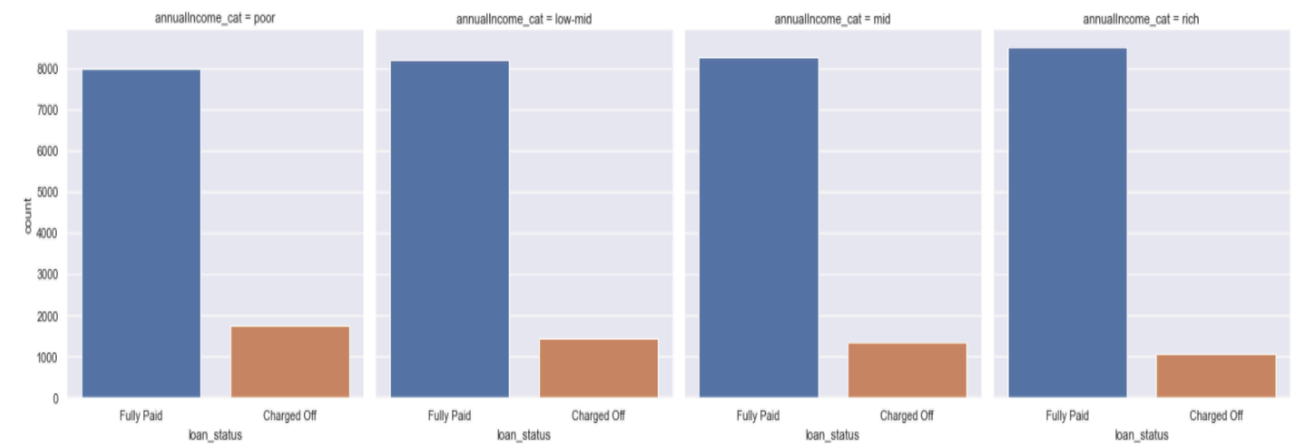
UNIVARIATE: CONTINOUS ANALYSIS – SAMPLE CHARTS

UNIVARIATE: SEGMENTED ANALYSIS – SAMPLE CHARTS









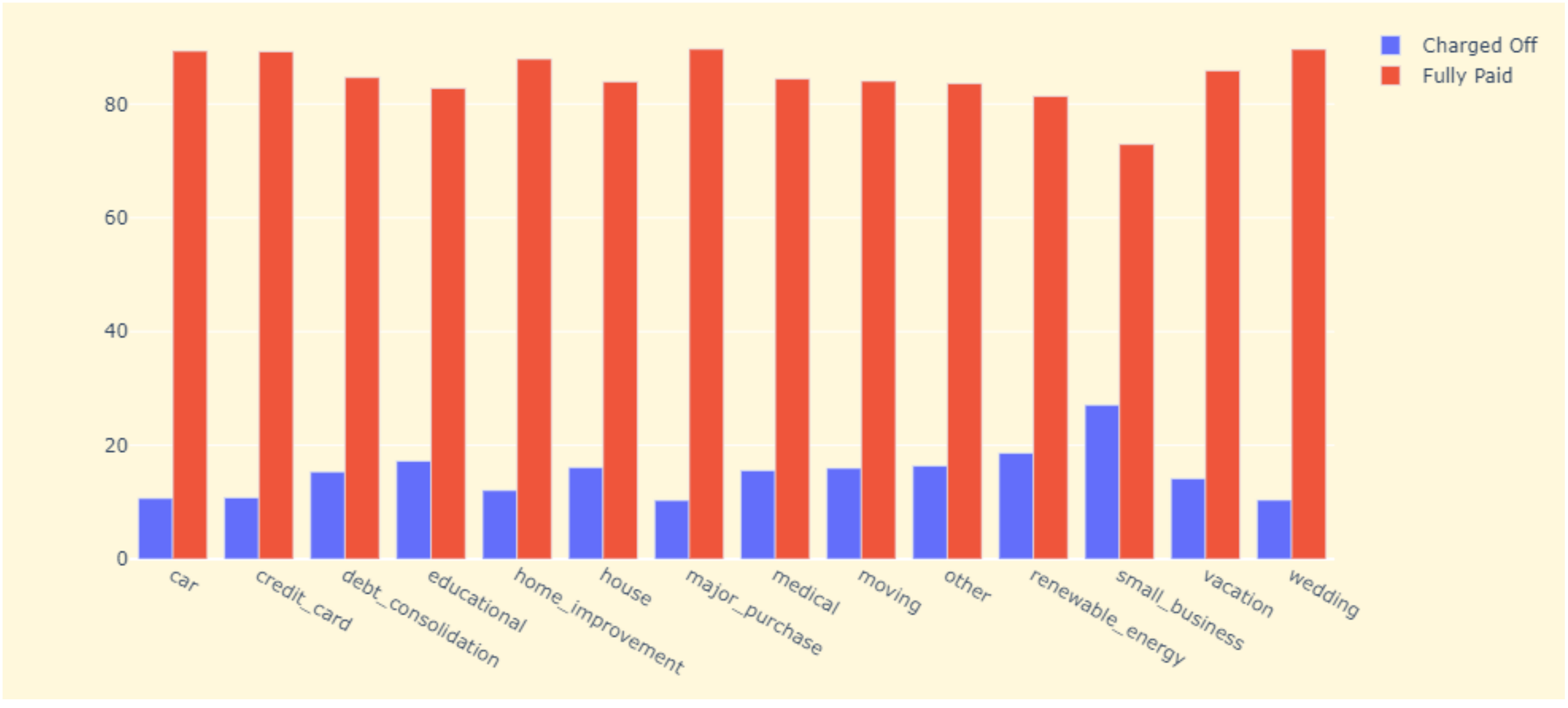
UNIVARIATE: SEGMENTED ANALYSIS - INFERENCE

Home Ownership vs Loan Status: Those who don't own a home are more likely to default.

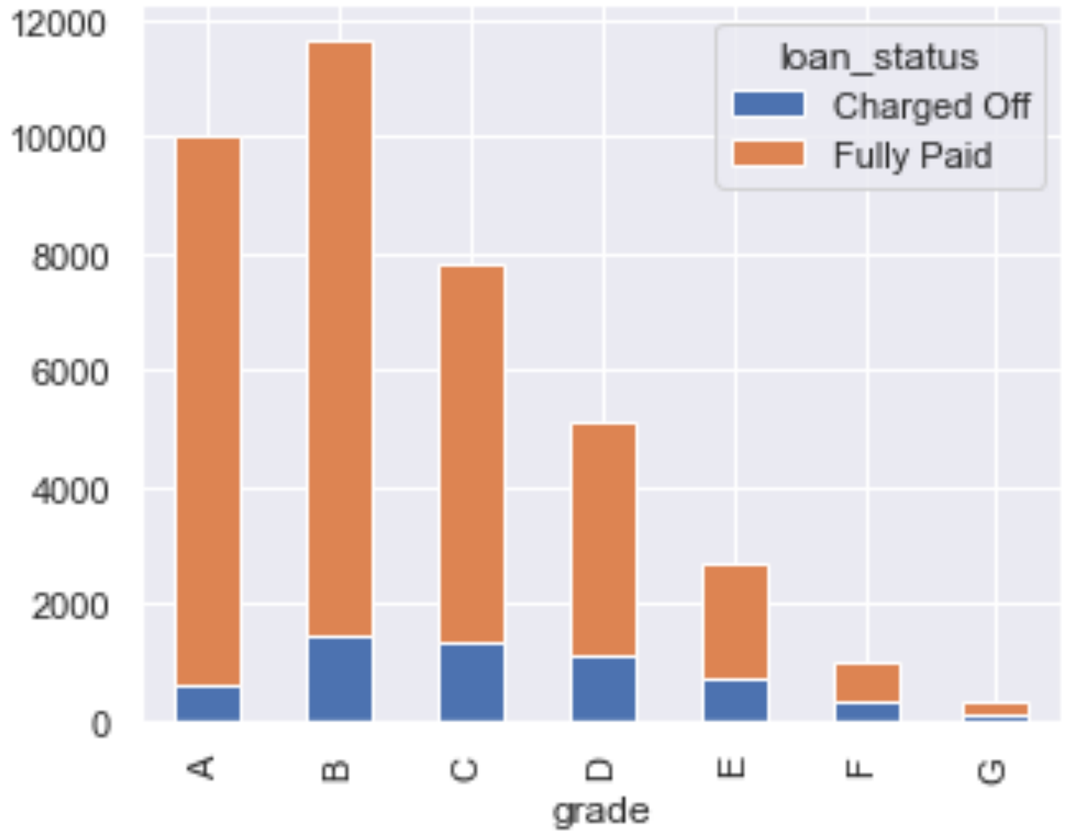
Term vs Loan Status: Long term loans are more likely to be charged off.

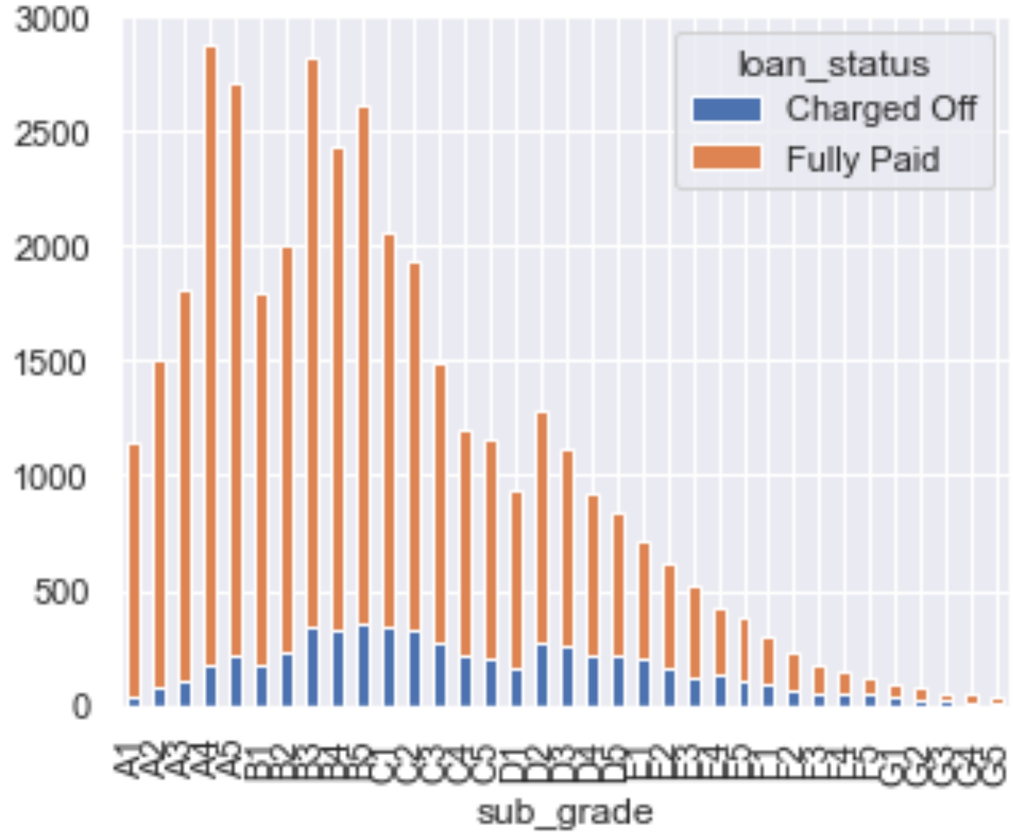
Low-income group is more likely to not fully pay the loan.

UNIVARIATE: SEGMENTED ANALYSIS SAMPLE CHARTS



UNIVARIATE: SEGMENTED ANALYSIS SAMPLE CHARTS





UNIVARIATE: SEGMENTED ANALYSIS - INFERENCE

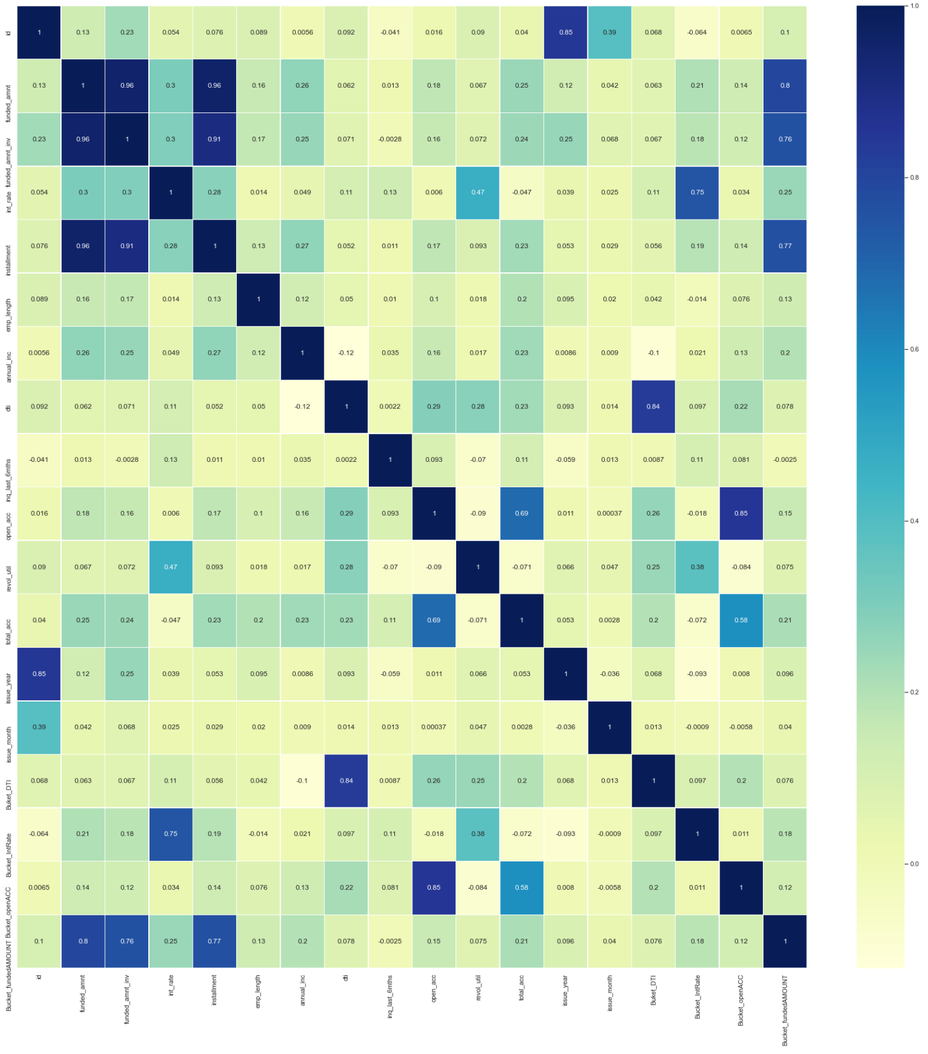
Purpose vs Loan Status: Small businesses are more likely to default.

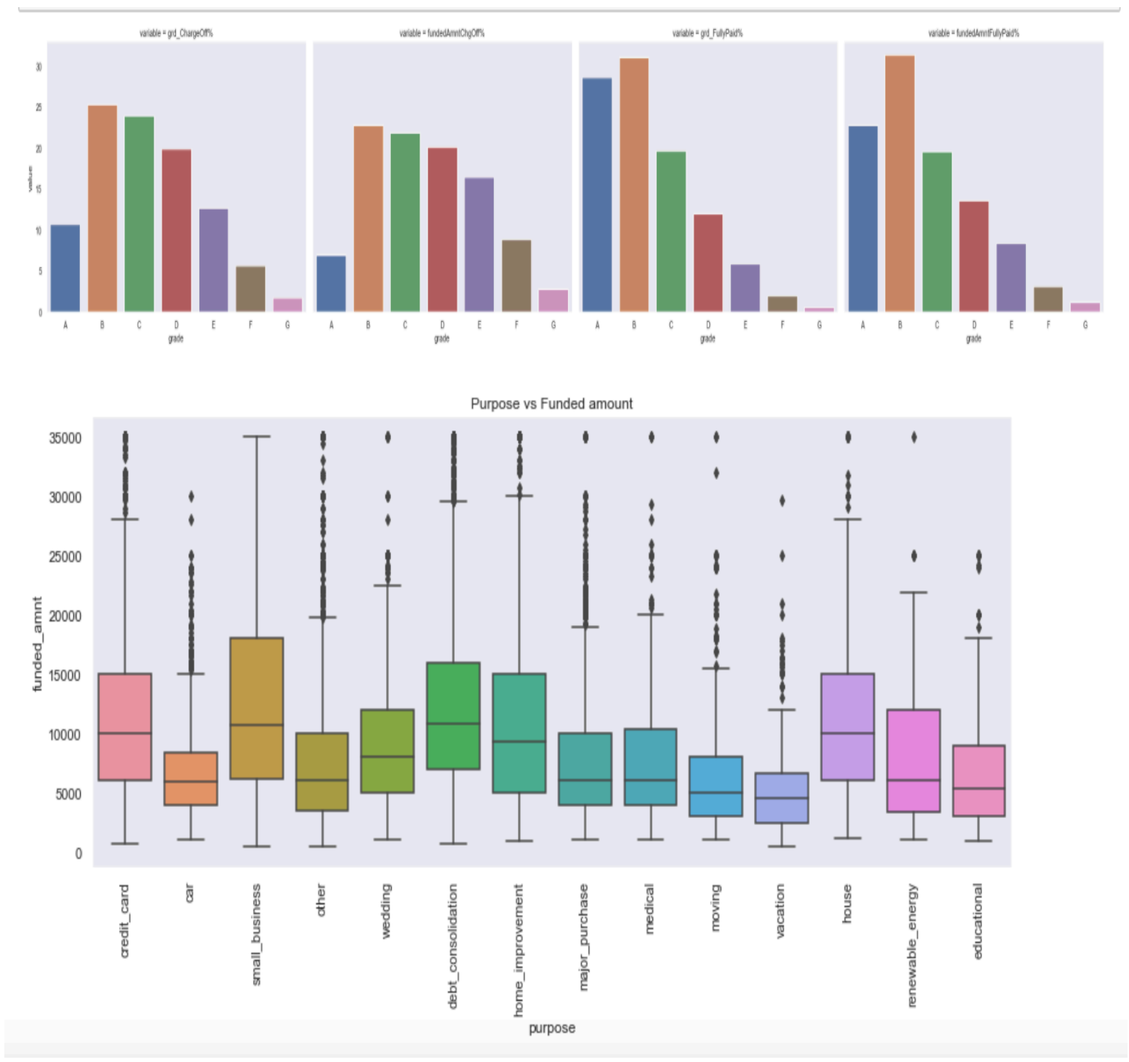
Grade vs Loan Status: Grade C is more likely to not fully pay the loan. While B5 seems to have the highest charged off rate.

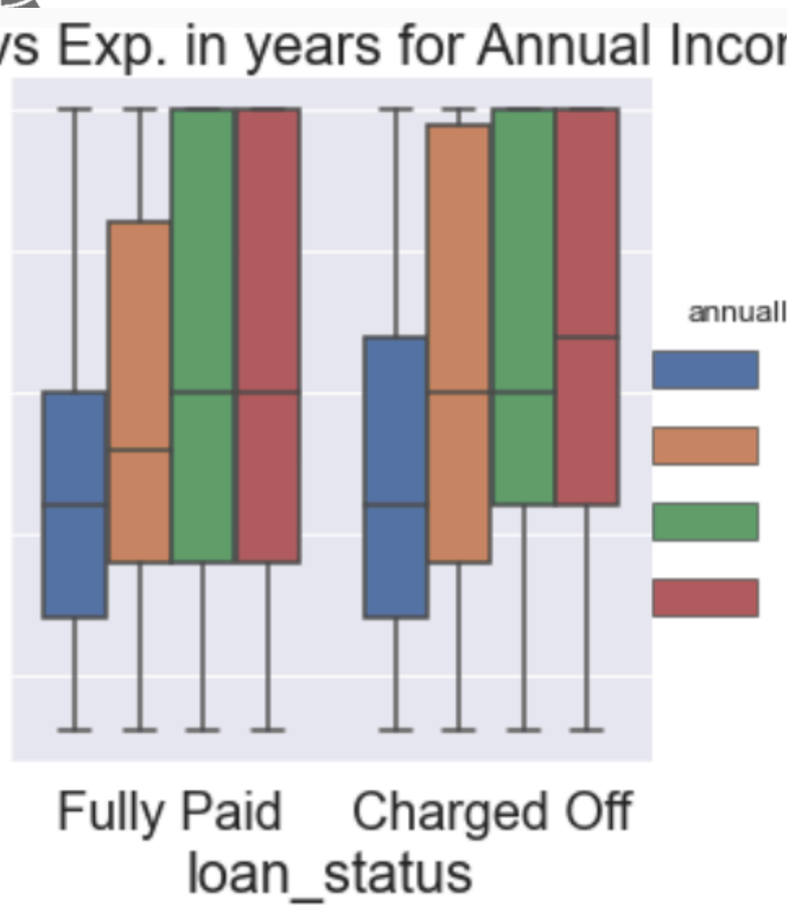
BIVARIATE ANALYSIS

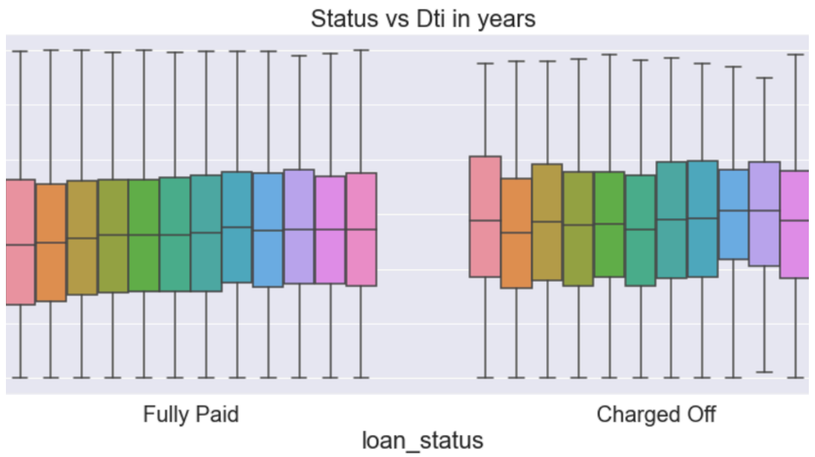
- Heatmaps, Pivot tables are used to analyse Continuous variables and categorical variables respectively.
- Observed and calculated percentage values are plotted using barplot.
- Constructing a pivot table to calculate the percentage of funded amount for each state considering the loan statuses
- Constructing a pivot table limiting to top 5 states and calculating the charge off and full paid percentage for each purpose of loans
- BAR charts are used to infer that **Funded_amnt_inv, installment, emp_length, total_acc, annual_inc** are excluded for further analysis

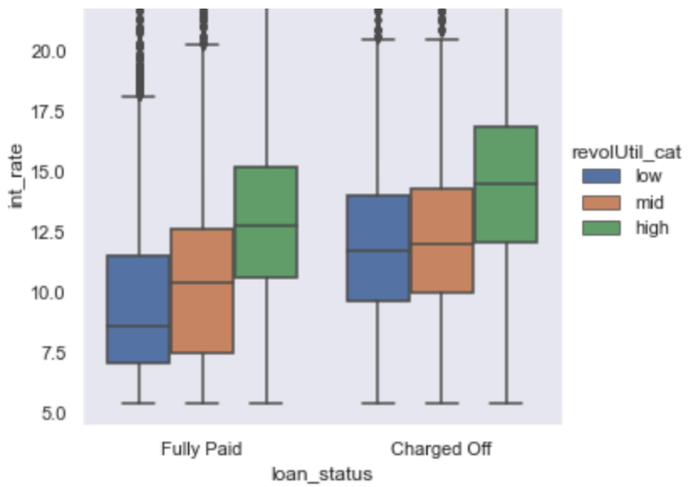
BIVARIATE ANALYSIS: SAMPLE CHARTS

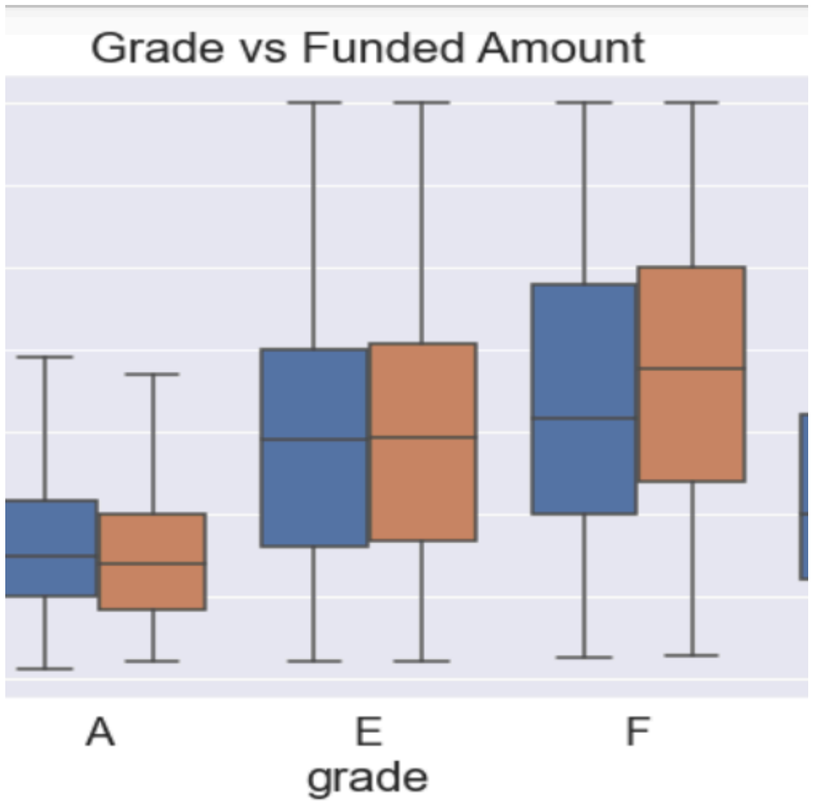


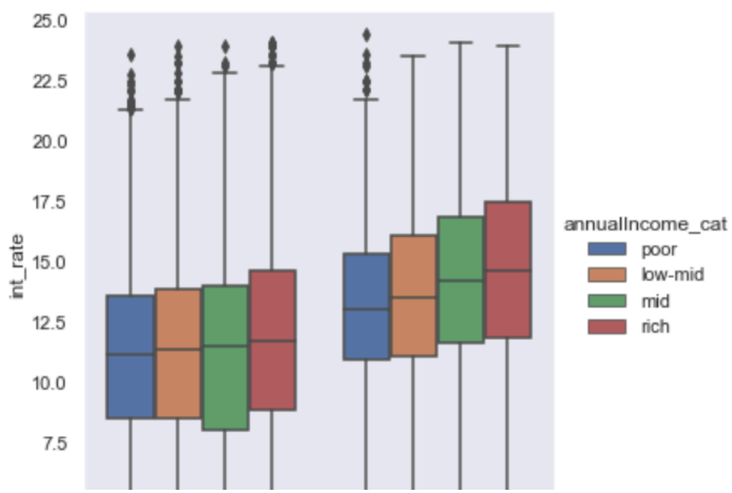












BIVARIATE ANALYSIS: SAMPLE CHARTS

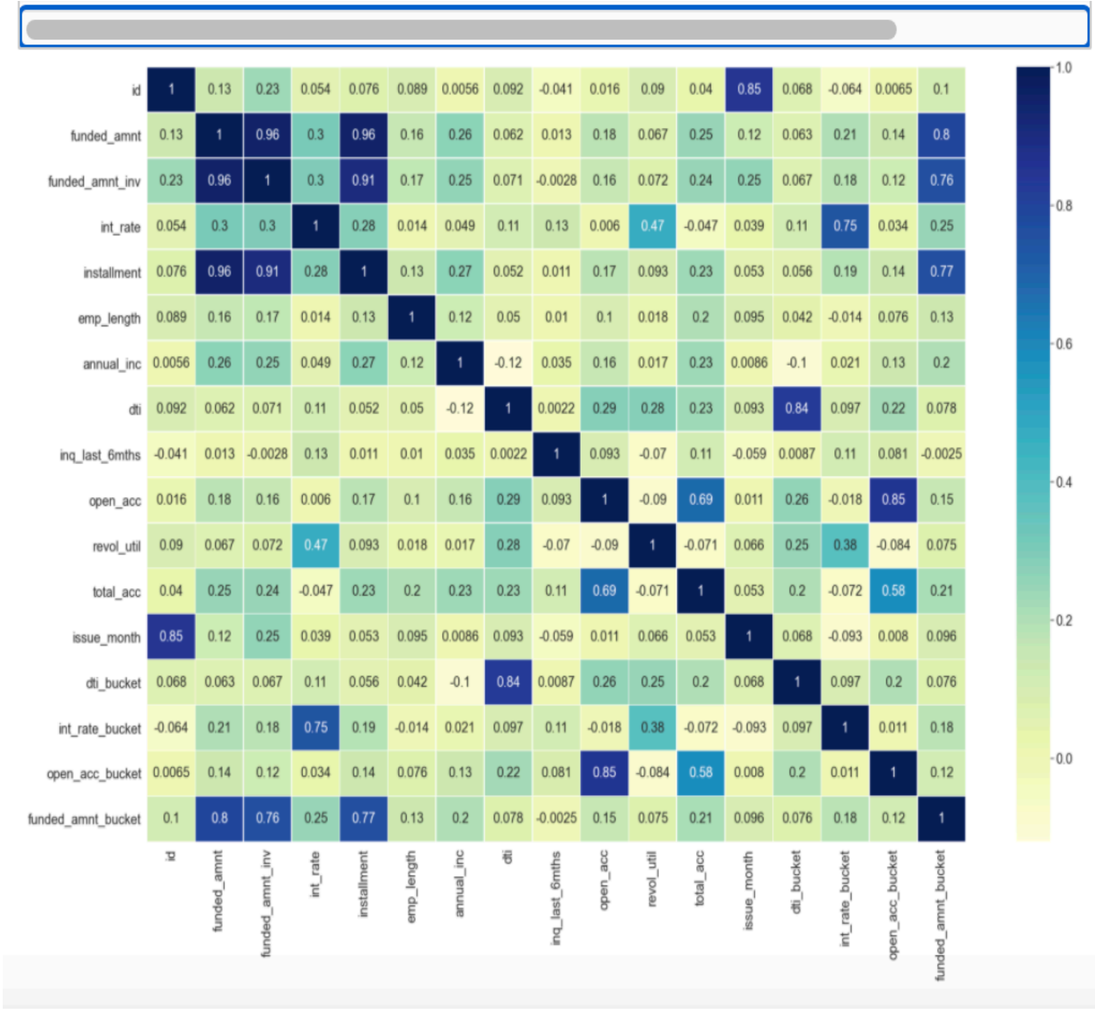
DERIVED METRICS - ANALYSIS

The following derived columns are determined:

- Dti_bucket: has new grouping based on dtiranges
- Int_rate_bucket: has new grouping based on int_ratecolumn
- Open_acc_bucket: has new grouping based on open_acccolumns
- Funded_amt_bucket: has new grouping based on funded amount

Correlation Chart - Derived Metrics

- Funded_amnt is highly correlated with funded_amnt_inv & installment.
- The Derived metrics column funded_amnt_bucket is highly correlated to funded_amnt_inv & installment & Funded_amnt .
- Int_rate and int_rate_bucket has high correlation value of .75.
- Dti & dti_bucket are highly correlated too with value of 0.84 and hence we can use derived column for analysis.
- Open_acc and open_acc_bucket are highly correlated with value of .85



After dropping these features, 22 features remained for further analysis

DRIVER VARIABLES - ANALYSIS

```
#FundedAmount_PC
FundedAmount_PC = pd.pivot_table(loan,values=['id'],columns=['loan_status'],index=['Bucket_fundedAMOUNT'],aggfunc={'id'
FundedAmount_PC = FundedAmount_PC.reset_index()
FundedAmount_PC['Total'] = FundedAmount_PC['id']['Charged Off'] + FundedAmount_PC['id']['Fully Paid']
FundedAmount_PC['%'] = (FundedAmount_PC['id']['Charged Off'] * 100) / (FundedAmount_PC['id']['Charged Off'] + FundedAmc
FundedAmount_PC = FundedAmount_PC.reset_index()
FundedAmount_PC
```

	index	Bucket_fundedAMOUNT	id		Total	%
loan_status			Charged Off	Fully Paid		
0	0	10000	2995	19557	22552	13.280419
1	1	20000	1870	10340	12210	15.315315
2	2	20001	762	3053	3815	19.973788

```
#InterestRate_PC
InterestRate_PC = pd.pivot_table(loan,values=['id'],columns=['loan_status'],index=['Bucket_IntRate'],aggfunc={'id':len}
InterestRate_PC = InterestRate_PC.reset_index()
InterestRate_PC['Total'] = InterestRate_PC['id']['Charged Off'] + InterestRate_PC['id']['Fully Paid']
InterestRate_PC['%'] = (InterestRate_PC['id']['Charged Off'] * 100) / (InterestRate_PC['id']['Charged Off'] + InterestF
InterestRate_PC = InterestRate_PC.reset_index()
InterestRate_PC
```

	index	Bucket_IntRate	id		Total	%
loan_status			Charged Off	Fully Paid		
0	0	8	440	7778	8218	5.354101
1	1	16	3625	21326	24951	14.528476
2	2	17	1562	3846	5408	28.883136

```
#DTI_PC
DTI_PC = pd.pivot_table(loan,values=['id'],columns=['loan_status'],index=['Buket_DTI'],aggfunc={'id':len})
DTI_PC = DTI_PC.reset_index()
DTI_PC['Total'] = DTI_PC['id']['Charged Off'] + DTI_PC['id']['Fully Paid']
DTI_PC['%'] = (DTI_PC['id']['Charged Off'] * 100) / (DTI_PC['id']['Charged Off'] + DTI_PC['id']['Fully Paid'])
DTI_PC = DTI_PC.reset_index()
DTI_PC
```

	index	Buket_DTI	id		Total	%
loan_status			Charged Off	Fully Paid		
0	0	10	1631	11304	12935	12.609200
1	1	20	2791	15650	18441	15.134754
2	2	21	1205	5996	7201	16.733787

```
#OPENACC_PC
OPENACC_PC = pd.pivot_table(loan,values=['id'],columns=['loan_status'],index=['Bucket_openACC'],aggfunc={'id':len})
OPENACC_PC['Total'] = OPENACC_PC['id']['Charged Off'] + OPENACC_PC['id']['Fully Paid']
OPENACC_PC['%'] = (OPENACC_PC['id']['Charged Off'] * 100) / (OPENACC_PC['id']['Charged Off'] + OPENACC_PC['id']['Fully
OPENACC_PC = OPENACC_PC.reset_index()
OPENACC_PC
```

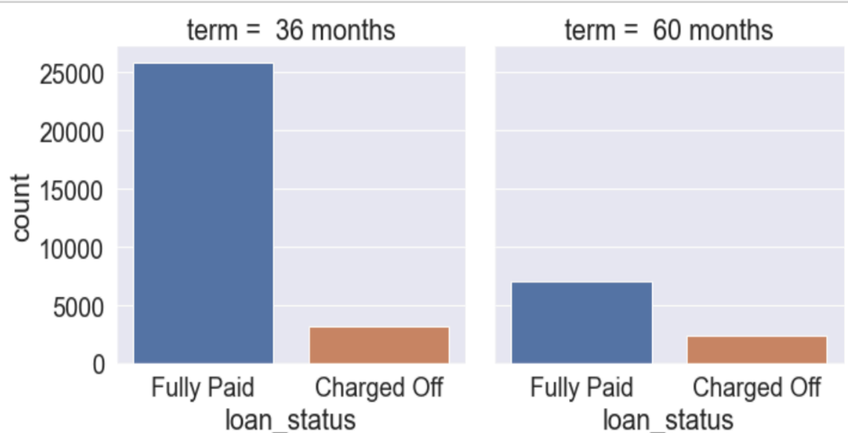
	Bucket_openACC	id		Total	%
loan_status		Charged Off	Fully Paid		
0	10	3803	21947	25750	14.768932
1	20	1707	10390	12097	14.110937
2	30	109	592	701	15.549215
3	31	8	21	29	27.586207

DRIVER VARIABLES – ANALYSIS (contd)

```
purpose_pc = pd.pivot_table(loan,values=['id'],columns=['loan_status'],index=['purpose'],aggfunc={'id':(lambda x:x.cour
purpose_pc = purpose_pc.reset_index()
purpose_pc['purposeChargeOff%'] = round(100*(purpose_pc['id']['Charged Off']/ purpose_pc['id']['Charged Off'].sum()),2)
purpose_pc['purposeFullyPaid%'] = round(100*(purpose_pc['id']['Fully Paid']/ purpose_pc['id']['Fully Paid'].sum()),2)
Purpose_mstr = purpose_pc[['purpose','purposeChargeOff%','purposeFullyPaid%']]
pd.DataFrame(Purpose_mstr).sort_values(['purposeChargeOff%'],ascending=False)
```

loan_status	purpose	purposeChargeOff%	purposeFullyPaid%
2	debt_consolidation	49.17	46.40
9	other	11.25	9.81
1	credit_card	9.63	13.61
11	small_business	8.44	3.88
4	home_improvement	6.17	7.67
6	major_purchase	3.95	5.85
0	car	2.84	4.06
7	medical	1.88	1.75
13	wedding	1.71	2.52
8	moving	1.63	1.47
5	house	1.05	0.93
3	educational	1.00	0.82
12	vacation	0.94	0.98
10	renewable_energy	0.34	0.25

```
catplot = sns.catplot(x='loan_status',col='term',data=loan,kind='count')
plt.show()
```




```
Grade_percent = pd.pivot_table(loan, values=['funded_amnt', 'id'],columns=['loan_status'],index=['grade'],aggfunc={'id'
Grade_percent = Grade_percent.reset_index()
Grade_percent
```

	grade	funded_amnt		id	
loan_status		Charged Off	Fully Paid	Charged Off	Fully Paid
0	A	4590650	79740800	602	9443
1	B	15054300	109722950	1425	10250
2	C	14498050	68505200	1347	6487
3	D	13320575	47377375	1118	3967
4	E	10918125	29315750	715	1948
5	F	5908050	11129075	319	657
6	G	1846625	4089100	101	198

```
AddressState_pc = pd.pivot_table(loan,values=['funded_amnt'],columns=['loan_status'],index=['addr_state'],aggfunc={'fur
AddressState_pc = AddressState_pc.reset_index()
AddressState_pc['addrStateChargeOff%'] = round(100*(AddressState_pc['funded_amnt']['Charged Off']/ AddressState_pc['fur
AddressState_pc['addrStateFullyPaid%'] = round(100*(AddressState_pc['funded_amnt']['Fully Paid']/ AddressState_pc['func
AddressState_Mstr = AddressState_pc[['addr_state','addrStateChargeOff%','addrStateFullyPaid%']]
pd.DataFrame(AddressState_Mstr).sort_values(['addrStateChargeOff%'],ascending=False).head(7)
```

loan_status	addr_state	addrStateChargeOff%	addrStateFullyPaid%
4	CA	20.19	17.91
33	NY	8.88	9.83
9	FL	8.60	6.56
42	TX	5.89	7.41
30	NJ	5.12	4.84
10	GA	3.60	3.50
14	IL	3.55	4.00

CONCLUSION

- Via EDA analysis, it is concluded the following feature sets has maximum impact in determining potential defaulting applicant.

- **Shortlisted quantitative variables/measures:**

  - funded_amnt (or derived column funded_amnt_bucket), int_rate (or derived column int_rate_bucket), Dti (or derived column int_rate_bucket), open_acc (or derived column open_acc_bucket)

- **Shortlisted categorical driver variables:**

  - Purpose, term, grade, addr_state