fintech_logR

December 23, 2024

1 Problem Statement

Given a set of attributes for an individual, predict whether they qualify for a personal loan and recommend appropriate repayment terms based on their creditworthiness and risk profile. The goal is to assess borrower behavior, financial history, and risk factors to determine whether extending credit is viable, and if so, to propose optimal loan conditions such as repayment duration, interest rate, and loan amount. This decision-making process aims to balance the financial needs of the borrower with the lender's risk management strategy, enhancing both customer satisfaction and business profitability.

[3]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from scipy.stats import zscore
```

2 Exploratory Data Analysis

[4]:
```
df=pd.read_csv('data.csv')
df
```

[4]:

	loan_amnt	term	int_rate	installment	grade	sub_grade	\
0	10000.0	36 months	11.44	329.48	B	B4	
1	8000.0	36 months	11.99	265.68	B	B5	
2	15600.0	36 months	10.49	506.97	B	B3	
3	7200.0	36 months	6.49	220.65	A	A2	
4	24375.0	60 months	17.27	609.33	C	C5	
…	…	…	…	…	…	…	
396025	10000.0	60 months	10.99	217.38	B	B4	
396026	21000.0	36 months	12.29	700.42	C	C1	
396027	5000.0	36 months	9.99	161.32	B	B1	
396028	21000.0	60 months	15.31	503.02	C	C2	
396029	2000.0	36 months	13.61	67.98	C	C2	

	emp_title	emp_length	home_ownership	annual_inc	…	\
0	Marketing	10+ years	RENT	117000.0	…	
1	Credit analyst	4 years	MORTGAGE	65000.0	…	

```
2                   Statistician   < 1 year           RENT     43057.0  …
3                 Client Advocate    6 years           RENT     54000.0  …
4          Destiny Management Inc.    9 years       MORTGAGE     55000.0  …
…                             …         …             …          …  …
396025           licensed bankere    2 years           RENT     40000.0  …
396026                      Agent    5 years       MORTGAGE    110000.0  …
396027               City Carrier  10+ years           RENT     56500.0  …
396028       Gracon Services, Inc  10+ years       MORTGAGE     64000.0  …
396029  Internal Revenue Service  10+ years           RENT     42996.0  …

        open_acc pub_rec revol_bal revol_util total_acc  initial_list_status  \
0           16.0     0.0   36369.0       41.8      25.0                    w
1           17.0     0.0   20131.0       53.3      27.0                    f
2           13.0     0.0   11987.0       92.2      26.0                    f
3            6.0     0.0    5472.0       21.5      13.0                    f
4           13.0     0.0   24584.0       69.8      43.0                    f
…          …     …       …         …        …                  …
396025       6.0     0.0    1990.0       34.3      23.0                    w
396026       6.0     0.0   43263.0       95.7       8.0                    f
396027      15.0     0.0   32704.0       66.9      23.0                    f
396028       9.0     0.0   15704.0       53.8      20.0                    f
396029       3.0     0.0    4292.0       91.3      19.0                    f

        application_type  mort_acc  pub_rec_bankruptcies  \
0             INDIVIDUAL       0.0                   0.0
1             INDIVIDUAL       3.0                   0.0
2             INDIVIDUAL       0.0                   0.0
3             INDIVIDUAL       0.0                   0.0
4             INDIVIDUAL       1.0                   0.0
…                    …       …                     …
396025        INDIVIDUAL       0.0                   0.0
396026        INDIVIDUAL       1.0                   0.0
396027        INDIVIDUAL       0.0                   0.0
396028        INDIVIDUAL       5.0                   0.0
396029        INDIVIDUAL       NaN                   0.0

                                                 address
0          0174 Michelle Gateway\r\nMendozaberg, OK 22690
1        1076 Carney Fort Apt. 347\r\nLoganmouth, SD 05113
2        87025 Mark Dale Apt. 269\r\nNew Sabrina, WV 05113
3                  823 Reid Ford\r\nDelacruzside, MA 00813
4                   679 Luna Roads\r\nGreggshire, VA 11650
…                                                     …
396025   12951 Williams Crossing\r\nJohnnyville, DC 30723
396026  0114 Fowler Field Suite 028\r\nRachelborough, …
396027  953 Matthew Points Suite 414\r\nReedfort, NY 7…
396028  7843 Blake Freeway Apt. 229\r\nNew Michael, FL…
```

```
396029      787 Michelle Causeway\r\nBriannaton, AR 48052

[396030 rows x 27 columns]
```

[5]:
```
def extract_state(address):
    try:
        if ',' in address:
            return address.split(',')[1].split(' ')[1].strip()
        else:
            return address.split(' ')[1].strip()
    except IndexError:
        return None
```

[6]:
```
def extract_city(address):
    try:
        if ',' in address:
            return address.split(',')[0].strip()
        else:
            return address.split(' ')[0].strip()
    except IndexError:
        return None
```

[7]:
```
df['city'] = df.address.str.split(r'\r\n', expand=True)[1].apply(extract_city)
df['state'] = df.address.str.split(r'\r\n', expand=True)[1].apply(extract_state)
```

[8]:
```
df['earliest_cr_line_month']=df.earliest_cr_line.str.split(r'-', expand=True)[0]
df['earliest_cr_line_year']=df.earliest_cr_line.str.split(r'-', expand=True)[1].
 ↪astype(int)
```

[9]:
```
df['issue_month']=df.issue_d.str.split(r'-', expand=True)[0]
df['issue_year']=df.issue_d.str.split(r'-', expand=True)[1].astype(int)
```

[10]:
```
month_map = {
    'Jan': 1, 'Feb': 2, 'Mar': 3, 'Apr': 4, 'May': 5, 'Jun': 6,
    'Jul': 7, 'Aug': 8, 'Sep': 9, 'Oct': 10, 'Nov': 11, 'Dec': 12
}
```

[11]:
```
df['earliest_cr_line_month'] = df['earliest_cr_line_month'].map(month_map)
df['issue_month'] = df['issue_month'].map(month_map)
```

[12]:
```
df = df.drop(columns=['earliest_cr_line', 'address', 'issue_d'])
```

[13]:
```
df.head()
```

[13]:

	loan_amnt	term	int_rate	installment	grade	sub_grade	\
0	10000.0	36 months	11.44	329.48	B	B4	
1	8000.0	36 months	11.99	265.68	B	B5	
2	15600.0	36 months	10.49	506.97	B	B3	

```
3     7200.0   36 months      6.49       220.65     A        A2
4    24375.0   60 months     17.27       609.33     C        C5

                emp_title emp_length home_ownership  annual_inc  …  \
0               Marketing  10+ years           RENT    117000.0  …
1         Credit analyst     4 years       MORTGAGE     65000.0  …
2            Statistician   < 1 year           RENT     43057.0  …
3         Client Advocate    6 years           RENT     54000.0  …
4  Destiny Management Inc.    9 years       MORTGAGE     55000.0  …

  initial_list_status application_type mort_acc pub_rec_bankruptcies  \
0                   w       INDIVIDUAL      0.0                   0.0
1                   f       INDIVIDUAL      3.0                   0.0
2                   f       INDIVIDUAL      0.0                   0.0
3                   f       INDIVIDUAL      0.0                   0.0
4                   f       INDIVIDUAL      1.0                   0.0

           city  state  earliest_cr_line_month  earliest_cr_line_year  \
0   Mendozaberg     OK                       6                   1990
1    Loganmouth     SD                       7                   2004
2   New Sabrina     WV                       8                   2007
3  Delacruzside     MA                       9                   2006
4    Greggshire     VA                       3                   1999

   issue_month  issue_year
0            1        2015
1            1        2015
2            1        2015
3           11        2014
4            4        2013

[5 rows x 30 columns]
```

2.1 Observations on Data

[14]:
```
df.shape
```

[14]:
```
(396030, 30)
```

There are 396030 rows and 29 columns

Column Details:

1. **loan_amnt**: The amount requested by the borrower for the loan.
2. **term**: The loan duration, either 36 or 60 months.
3. **int_rate**: The annual interest rate applied to the loan.
4. **installment**: The fixed monthly repayment amount for the loan.
5. **grade**: LoanTap's assigned loan grade, indicating the risk level (Risk rating by LoanTap).

6. **sub_grade**: A more granular risk rating assigned by LoanTap to the borrower.
7. **emp_title**: The job title of the borrower.
8. **emp_length**: The number of years the borrower has been employed (0-10 years).
9. **home_ownership**: The borrower's housing status (e.g., own, rent, mortgage).
10. **annual_inc**: The borrower's annual income.
11. **verification_status**: Whether the borrower's income has been verified.
12. **issue_d**: The date when the loan was issued.
13. **loan_status**: The current status of the loan (e.g., fully paid, charged off).
14. **purpose**: The reason the borrower is requesting the loan.
15. **title**: The title given to the loan by the borrower.
16. **dti (Debt-to-Income ratio)**: The ratio of the borrower's monthly debt payments to their monthly income.
17. **earliest_cr_line**: The date when the borrower's oldest credit account was opened.
18. **open_acc**: The number of active credit accounts the borrower has.
19. **pub_rec**: The number of negative records on the borrower's public credit profile.
20. **revol_bal**: The total balance on the borrower's revolving credit accounts (e.g., credit cards).
21. **revol_util**: The percentage of available credit used on revolving accounts.
22. **total_acc**: The total number of credit lines the borrower has.
23. **initial_list_status**: The loan's initial category, either 'W' (waiting) or 'F' (funded).
24. **application_type**: Whether the application was made individually or jointly with another borrower.
25. **mort_acc**: The number of mortgages the borrower holds.
26. **pub_rec_bankruptcies**: The number of bankruptcy records associated with the borrower.
27. **address**: The location or geographical area of the borrower.

[13]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 396030 entries, 0 to 396029
Data columns (total 30 columns):
 #   Column                Non-Null Count   Dtype  
---  ------                --------------   -----  
 0   loan_amnt             396030 non-null  float64
 1   term                  396030 non-null  object 
 2   int_rate              396030 non-null  float64
 3   installment           396030 non-null  float64
 4   grade                 396030 non-null  object 
 5   sub_grade             396030 non-null  object 
 6   emp_title             373103 non-null  object 
 7   emp_length            377729 non-null  object 
 8   home_ownership        396030 non-null  object 
 9   annual_inc            396030 non-null  float64
 10  verification_status   396030 non-null  object 
 11  loan_status           396030 non-null  object 
 12  purpose               396030 non-null  object 
 13  title                 394274 non-null  object 
 14  dti                   396030 non-null  float64
 15  open_acc              396030 non-null  float64
```

```
 16  pub_rec                 396030 non-null  float64
 17  revol_bal               396030 non-null  float64
 18  revol_util              395754 non-null  float64
 19  total_acc               396030 non-null  float64
 20  initial_list_status     396030 non-null  object
 21  application_type        396030 non-null  object
 22  mort_acc                358235 non-null  float64
 23  pub_rec_bankruptcies    395495 non-null  float64
 24  city                    396030 non-null  object
 25  state                   396030 non-null  object
 26  earliest_cr_line_month  396030 non-null  int64
 27  earliest_cr_line_year   396030 non-null  int64
 28  issue_month             396030 non-null  int64
 29  issue_year              396030 non-null  int64
dtypes: float64(12), int64(4), object(14)
memory usage: 90.6+ MB
```

Data Types:

- **Numerical columns:** 12 columns are of type float64, representing financial amounts, rates, and numerical measurements (e.g., loan_amnt, int_rate, installment, etc.).
- **Categorical columns:** 15 columns are of type object, representing categorical variables such as term, grade, emp_title, home_ownership, loan_status, etc. These need to be properly encoded for machine learning tasks.
- **List of Columns with null values:** emp_title, emp_length, title, revol_util, mort_acc, pub_rec_bankruptcies

[14]:
```
print('Statiscal description of categorical columns')
df.describe(include=['object'])
```

```
Statiscal description of categorical columns
```

[14]:

	term	grade	sub_grade	emp_title	emp_length	home_ownership	\
count	396030	396030	396030	373103	377729	396030	
unique	2	7	35	173105	11	6	
top	36 months	B	B3	Teacher	10+ years	MORTGAGE	
freq	302005	116018	26655	4389	126041	198348	

	verification_status	loan_status	purpose	\
count	396030	396030	396030	
unique	3	2	14	
top	Verified	Fully Paid	debt_consolidation	
freq	139563	318357	234507	

	title	initial_list_status	application_type	city	\
count	394274	396030	396030	396030	
unique	48816	2	3	67513	
top	Debt consolidation	f	INDIVIDUAL	DPO	

```
freq                152472              238066           395319    14289
```

```
          state
count    396030
unique       54
top          AP
freq      14308
```

Statistical Summary (Categorical Columns):

- **loan_status:** The most frequent loan status is "Fully Paid", with a frequency of 318,357, indicating that the majority of the loans in the dataset have been paid off.
- **home_ownership:** Most borrowers own homes, with MORTGAGE being the most frequent value (198,348 occurrences).
- **purpose:** The most frequent loan purpose is debt_consolidation, showing that a significant number of loans are for consolidating existing debt.
- **emp_title:** A diverse range of job titles, but with many missing values, suggesting that many records do not specify the job title of the borrower.
- **emp_length:** The majority of borrowers have 10+ years of employment history, but some records have missing or unspecified employment lengths.
- **issue_d:** The most frequent issue date is October 2014, indicating that the dataset spans multiple loan issue dates, with 115 unique dates.
- **city:** The most frequent city is DPO, appearing 14,289 times. The dataset covers a wide range of locations with 67,513 unique cities.
- **state:** The most frequent state is AP, appearing 14,308 times. The dataset includes loans from 54 unique states, showing broad geographical distribution.
- **term:** The most frequent term is 36 months, appearing 302,005 times. There are 2 unique terms in the dataset.
- **grade:** The most frequent grade is B, appearing 116,018 times. There are 7 unique grades.
- **sub_grade:** The most frequent sub-grade is B3, appearing 26,655 times. There are 35 unique sub-grades.
- **verification_status:** The most frequent verification status is Verified, appearing 139,563 times. There are 3 unique verification statuses.
- **title:** The most frequent loan title is Debt consolidation, appearing 152,472 times. There are 48,816 unique titles, with some missing values (1,756 rows).
- **initial_list_status:** The most frequent initial list status is f, appearing 238,066 times. There are 2 unique statuses.
- **application_type:** The most frequent application type is INDIVIDUAL, appearing 395,319 times. There are 3 unique application types.

[15]:
```
print('Statiscal description of numerical columns')
df.describe()
```

```
Statiscal description of numerical columns
```

[15]:

	loan_amnt	int_rate	installment	annual_inc	\
count	396030.000000	396030.000000	396030.000000	3.960300e+05	
mean	14113.888089	13.639400	431.849698	7.420318e+04	

```
std      8357.441341       4.472157     250.727790  6.163762e+04
min       500.000000       5.320000      16.080000  0.000000e+00
25%      8000.000000      10.490000     250.330000  4.500000e+04
50%     12000.000000      13.330000     375.430000  6.400000e+04
75%     20000.000000      16.490000     567.300000  9.000000e+04
max     40000.000000      30.990000    1533.810000  8.706582e+06

                dti       open_acc        pub_rec     revol_bal  \
count  396030.000000  396030.000000  396030.000000  3.960300e+05
mean       17.379514      11.311153       0.178191  1.584454e+04
std        18.019092       5.137649       0.530671  2.059184e+04
min         0.000000       0.000000       0.000000  0.000000e+00
25%        11.280000       8.000000       0.000000  6.025000e+03
50%        16.910000      10.000000       0.000000  1.118100e+04
75%        22.980000      14.000000       0.000000  1.962000e+04
max      9999.000000      90.000000      86.000000  1.743266e+06

          revol_util      total_acc       mort_acc  pub_rec_bankruptcies  \
count  395754.000000  396030.000000  358235.000000         395495.000000
mean       53.791749      25.414744       1.813991              0.121648
std        24.452193      11.886991       2.147930              0.356174
min         0.000000       2.000000       0.000000              0.000000
25%        35.800000      17.000000       0.000000              0.000000
50%        54.800000      24.000000       1.000000              0.000000
75%        72.900000      32.000000       3.000000              0.000000
max       892.300000     151.000000      34.000000              8.000000

       earliest_cr_line_month  earliest_cr_line_year    issue_month  \
count           396030.000000          396030.000000  396030.000000
mean                 6.756231            1997.857667       6.553188
std                  3.435011               7.198387       3.426622
min                  1.000000            1944.000000       1.000000
25%                  4.000000            1994.000000       4.000000
50%                  7.000000            1999.000000       7.000000
75%                 10.000000            2003.000000      10.000000
max                 12.000000            2013.000000      12.000000

          issue_year
count  396030.000000
mean     2013.629074
std         1.481725
min      2007.000000
25%      2013.000000
50%      2014.000000
75%      2015.000000
max      2016.000000
```

Statistical Summary (Numerical Columns):

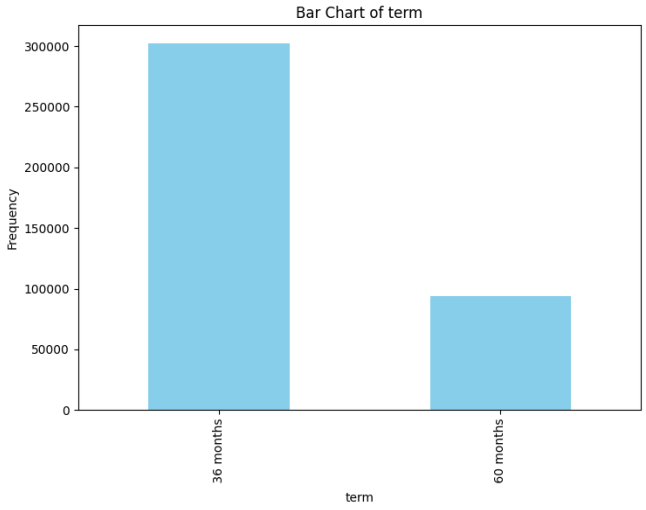
- **Loan Amount (loan_amnt):** The average loan amount is approximately 14,114, with a range between 500 and 40,000. This suggests that most loans are relatively smaller (around 8,000 - 12,000), with a few large loans driving the upper bound.
- **Interest Rate (int_rate):** The mean interest rate is 13.64%, with a standard deviation of 4.47%. This implies a somewhat uniform interest rate distribution with some outliers.
- **Installment (installment):** The average installment is 431.85, with a high standard deviation (250.73). This suggests that installment amounts can vary significantly depending on the loan amount.
- **Annual Income (annual_inc):** The average annual income is about 74,203, with a high standard deviation (61,637), indicating a large income disparity in the dataset.
- **Debt-to-Income Ratio (dti):** The average dti is 17.38, indicating a moderate level of debt compared to income on average.
- **open_acc (Open Accounts):** Average number of open accounts: 11.31. Range: 0 to 90. The majority have between 8 (25th percentile) and 14 (75th percentile) open accounts.
- **pub_rec (Public Records):** Average number of public records: 0.18. Range: 0 to 86. Most borrowers have 0 public records.
- **Revolving Balance (revol_bal):** The mean revolving balance is around 15,844, which seems reasonable given the range of credit balances.
- **Revolving Utilization (revol_util):** The mean is 53.79, showing that on average, individuals are using more than half of their available credit, with a std deviation of 24.45 suggesting high variation.
- **total_acc (Total Accounts):** Average total accounts: 25.41. Range: 2 to 151. Typically, borrowers have between 17 (25th percentile) and 32 (75th percentile) total accounts.
- **Mortgage Accounts (mort_acc):** The mort_acc column has a mean of 1.81 but also significant missing data (9.54%). It might be worth investigating how it impacts the dataset and whether imputation or dropping is better.
- **pub_rec_bankruptcies:** Average bankruptcies: 0.12. Range: 0 to 8. Most borrowers have 0 bankruptcies.
- **earliest_cr_line_month:** Average month: June (6.76). Range: January (1) to December (12).
- **earliest_cr_line_year:** Average year: 1997.86. Range: 1944 to 2013. The 25th percentile year is 1994, and the 75th percentile is 2003.

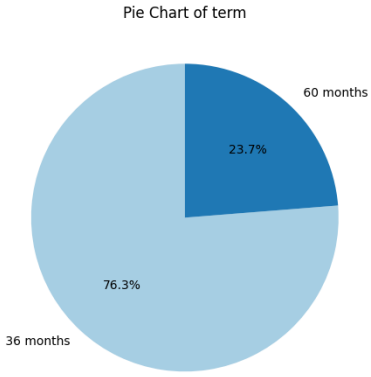
2.2 Univariate Analysis

[16]:
```
for i in df.select_dtypes(include=['object']).columns.tolist():
    print(f'Column name: {i} and it contains \n {df[i].value_counts()}')
    if len(df[i].value_counts()) > 50:
        print(f"\033[1mToo many values to plot bar and pie chart for column␣
 ↪'{i}'\033[0m \n\n")
        continue
    fig, axes = plt.subplots(1, 2, figsize=(14, 6))
    df[i].value_counts().plot(kind='bar', ax=axes[0], color='skyblue')
    axes[0].set_title(f'Bar Chart of {i}')
    axes[0].set_xlabel(i)
    axes[0].set_ylabel('Frequency')
```

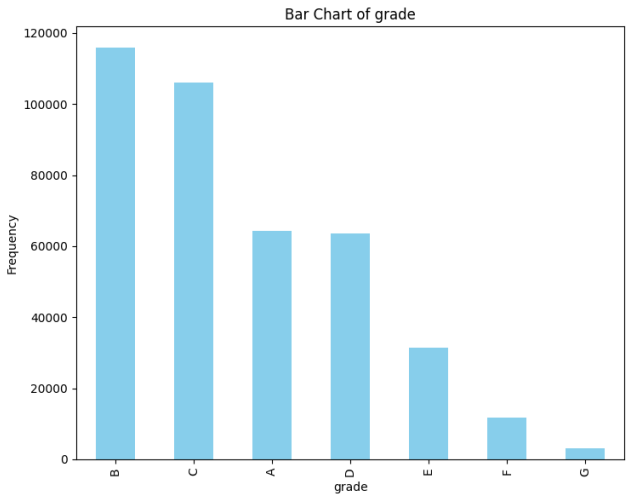
```
    pie_data = df[i].value_counts()
    axes[1].pie(pie_data, labels=pie_data.index, autopct='%1.1f%%',␣
↪startangle=90, colors=plt.cm.Paired.colors)
    axes[1].set_title(f'Pie Chart of {i}')
    plt.tight_layout()
    plt.show()
```

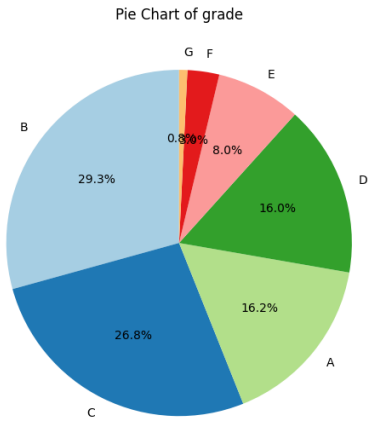
```
Column name: term and it contains
 term
36 months    302005
60 months     94025
Name: count, dtype: int64
```






```
Column name: grade and it contains
 grade
B    116018
C    105987
A     64187
D     63524
E     31488
F     11772
G      3054
Name: count, dtype: int64
```

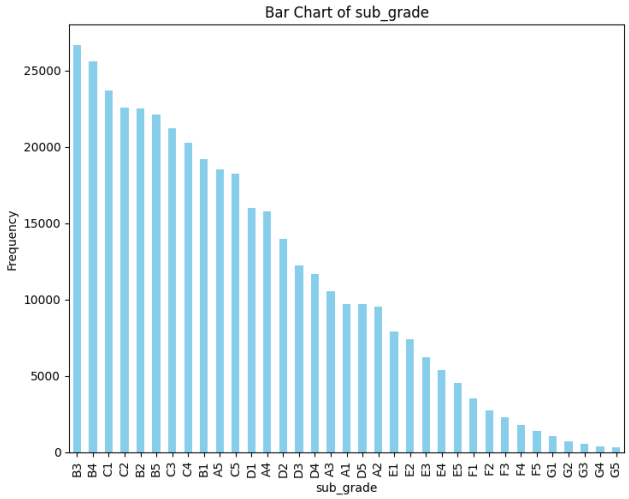


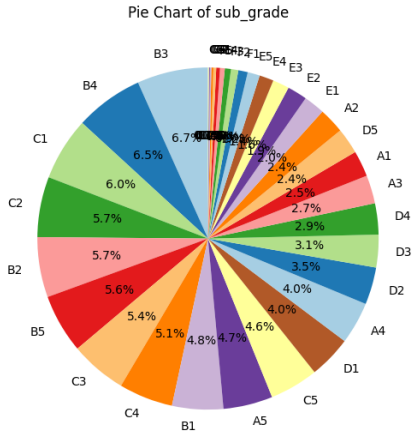


Column name: sub_grade and it contains

sub_grade	
B3	26655
B4	25601
C1	23662
C2	22580
B2	22495
B5	22085
C3	21221
C4	20280
B1	19182
A5	18526
C5	18244
D1	15993
A4	15789
D2	13951
D3	12223
D4	11657
A3	10576
A1	9729
D5	9700
A2	9567
E1	7917
E2	7431
E3	6207
E4	5361
E5	4572
F1	3536
F2	2766
F3	2286

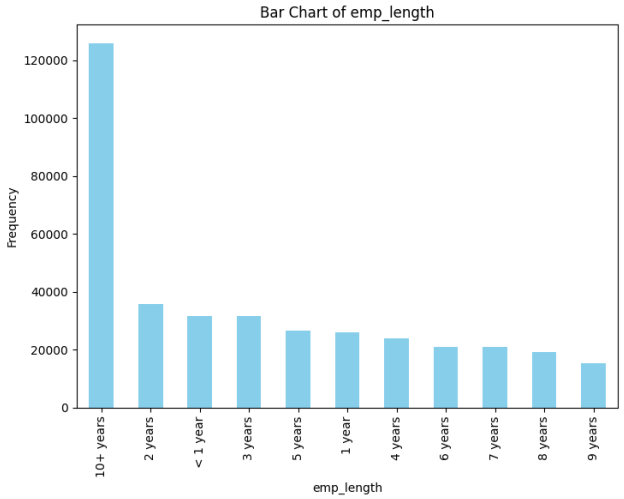
```
F4     1787
F5     1397
G1     1058
G2      754
G3      552
G4      374
G5      316
Name: count, dtype: int64
```

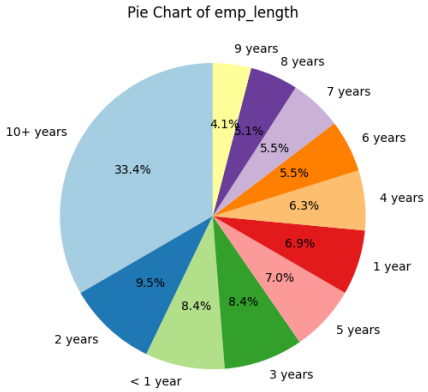




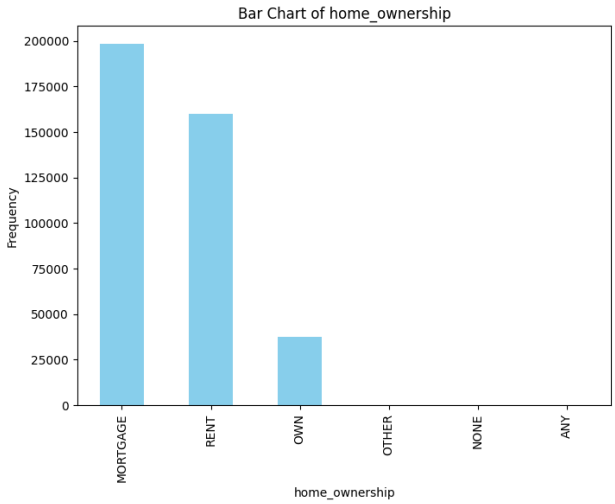

```
Column name: emp_title and it contains
 emp_title
Teacher                          4389
Manager                          4250
Registered Nurse                 1856
RN                               1846
Supervisor                       1830
                                 ...
Social Work/Care Manager            1
Regional Counsel                    1
Nor-Com Inc                         1
Director of the Bach Society        1
SPO II                              1
Name: count, Length: 173105, dtype: int64
Too many values to plot bar and pie chart for column 'emp_title'


Column name: emp_length and it contains
 emp_length
10+ years    126041
2 years       35827
< 1 year      31725
```

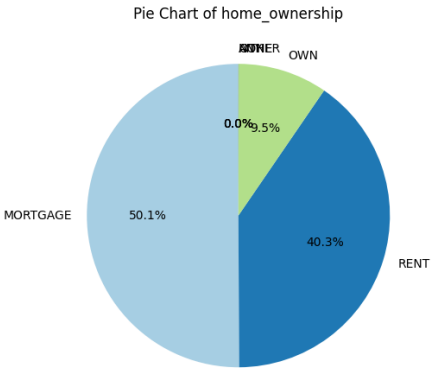
```
3 years       31665
5 years       26495
1 year        25882
4 years       23952
6 years       20841
7 years       20819
8 years       19168
9 years       15314
Name: count, dtype: int64
```



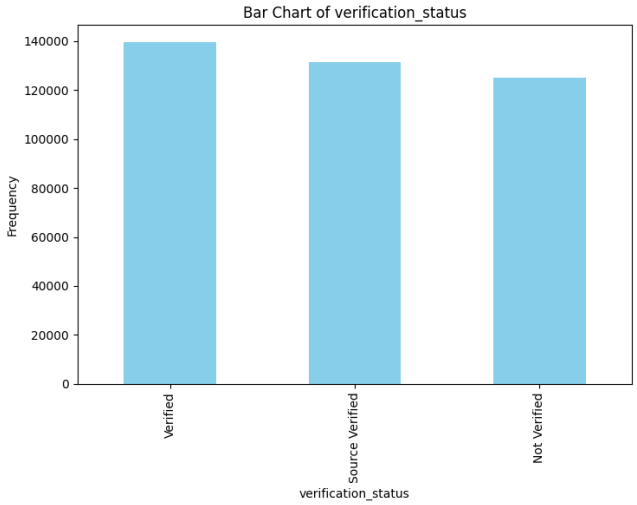


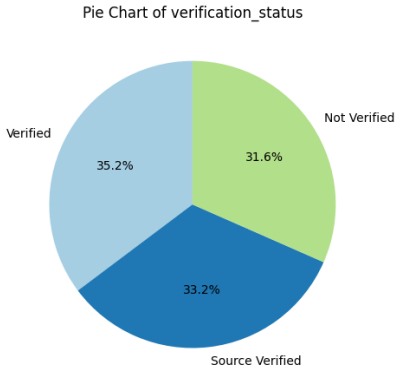

```
Column name: home_ownership and it contains
 home_ownership
MORTGAGE    198348
RENT        159790
OWN          37746
OTHER          112
NONE            31
ANY              3
Name: count, dtype: int64
```



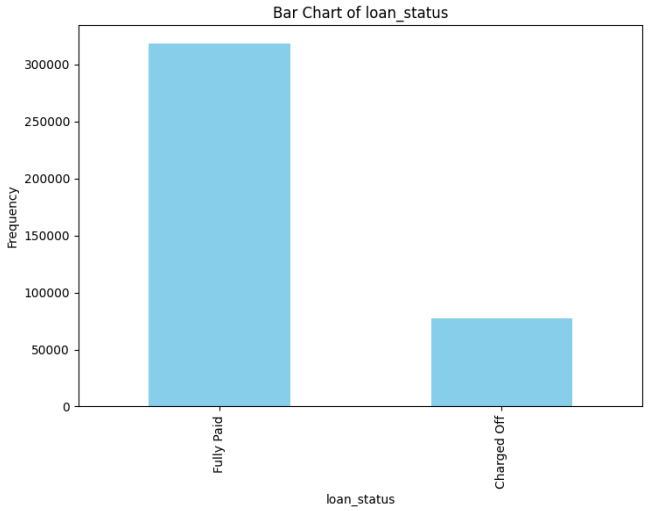


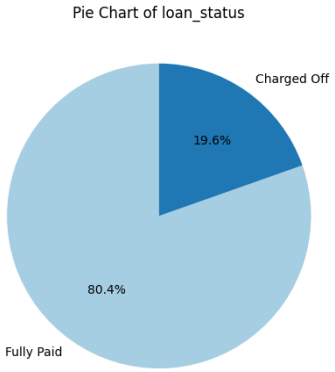

```
Column name: verification_status and it contains
 verification_status
Verified           139563
Source Verified    131385
Not Verified       125082
Name: count, dtype: int64
```



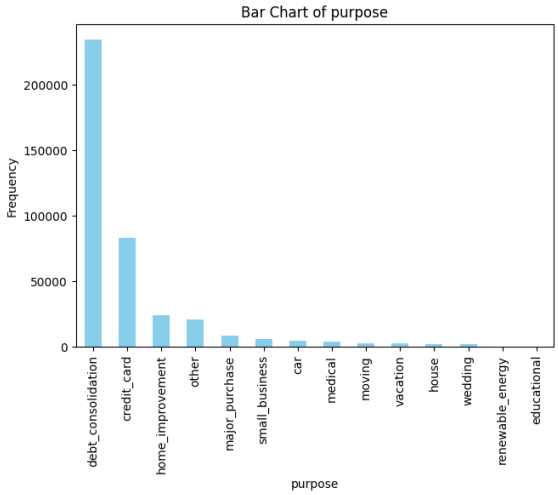


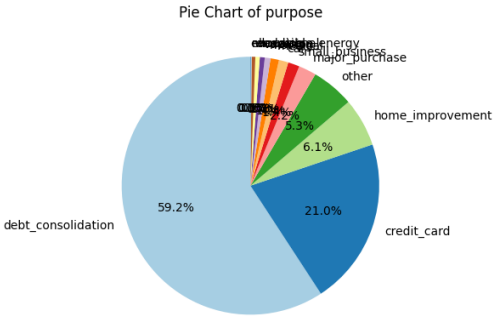

```
Column name: loan_status and it contains
 loan_status
Fully Paid     318357
Charged Off     77673
Name: count, dtype: int64
```



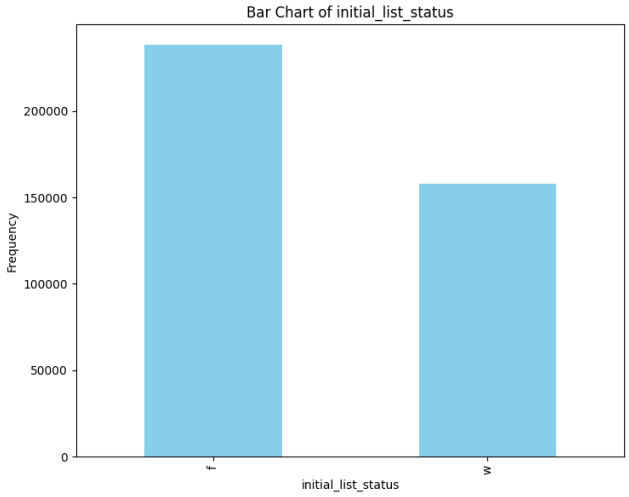


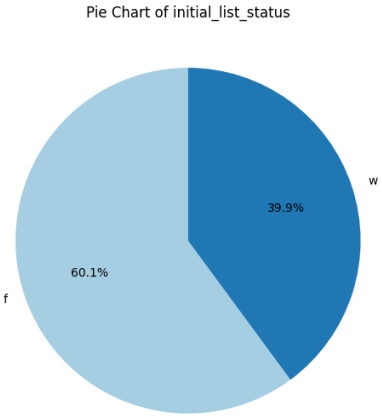

```
Column name: purpose and it contains
 purpose
debt_consolidation    234507
credit_card            83019
home_improvement       24030
other                  21185
major_purchase          8790
small_business          5701
car                     4697
medical                 4196
moving                  2854
vacation                2452
house                   2201
wedding                 1812
renewable_energy         329
educational              257
Name: count, dtype: int64
```



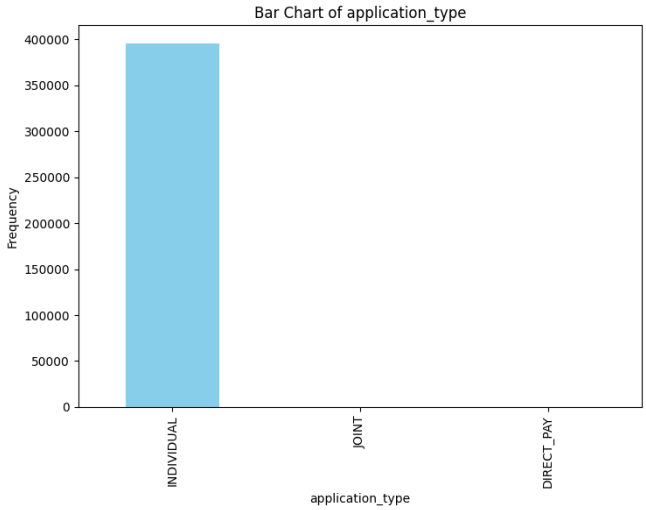


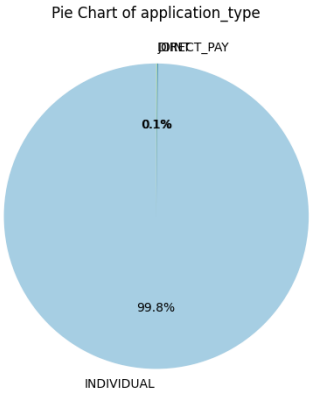

```
Column name: title and it contains
 title
Debt consolidation                          152472
Credit card refinancing                      51487
Home improvement                             15264
Other                                        12930
Debt Consolidation                           11608
                                             ...
Outboard Motor Repower Loan                      1
2011 Insurance and Debt Consolidation            1
Credit buster                                    1
Loanforpayoff                                    1
Toxic Debt Payoff                                1
Name: count, Length: 48816, dtype: int64
Too many values to plot bar and pie chart for column 'title'


Column name: initial_list_status and it contains
 initial_list_status
f    238066
w    157964
Name: count, dtype: int64
```






```
Column name: application_type and it contains
 application_type
INDIVIDUAL    395319
JOINT            425
DIRECT_PAY       286
Name: count, dtype: int64
```






```
Column name: city and it contains
 city
DPO                    14289
APO                    14060
FPO                    14035
East Michael             311
```

```
Port Michael              305
                          …
South Noahshire             1
Lake Jeremyfurt             1
East Dillonfurt             1
North Nicolasville          1
South Natashaborough        1
Name: count, Length: 67513, dtype: int64
```
Too many values to plot bar and pie chart for column 'city'

```
Column name: state and it contains
 state
AP    14308
AE    14157
AA    13919
NJ     7091
WI     7081
LA     7068
NV     7038
AK     7034
MA     7022
VA     7022
VT     7005
NY     7004
MS     7003
TX     7000
SC     6973
ME     6972
AR     6969
OH     6969
GA     6967
IN     6958
ID     6958
KS     6945
WV     6944
RI     6940
MO     6939
IL     6934
WY     6933
HI     6927
NE     6927
IA     6926
FL     6921
AZ     6918
CO     6914
OK     6911
MN     6904
```

```
CT     6904
NC     6901
OR     6898
CA     6898
AL     6898
MD     6896
WA     6895
SD     6887
UT     6887
MT     6883
DE     6874
TN     6869
ND     6858
MI     6854
NM     6842
DC     6842
PA     6825
NH     6818
KY     6800
Name: count, dtype: int64
Too many values to plot bar and pie chart for column 'state'
```

Analysis and Insights (Categorical Columns)

- **Term:**
  - The majority of loans have a term of 36 months (about 76%) compared to 60 months (around 24%).
  - Insight: Most borrowers prefer shorter-term loans, possibly due to quicker repayment cycles. This could indicate a more financially stable borrower group.
- **Grade:**
  - The most common grades are B (29%) and C (27%), followed by A (16%) and D (16%).
  - Insight: The grade distribution shows that the loans are largely concentrated in the middle-range credit grades. This suggests that most borrowers are not in the highest or lowest risk categories.
- **Sub-Grade:**
  - The sub-grades B3, B4, and C1 are the most frequent, with B3 being the highest.
  - Insight: Sub-grades are more granular, and this distribution shows a heavy skew toward B and C grades. Loans for these sub-grades likely represent moderate-risk borrowers.
- **Employment Title:**
  - There is a wide range of unique job titles (173,105 in total), with "Teacher", "Manager", "Registered Nurse", and "Supervisor" being the most frequent.
  - Insight: The variety in employment titles suggests a diverse borrower base. Certain job titles, like "Teacher" and "Manager", could represent stable income sources, which may correlate with loan repayment ability.
- **Employment Length:**
  - The majority of borrowers have 10+ years of employment (about 31%), followed by 2

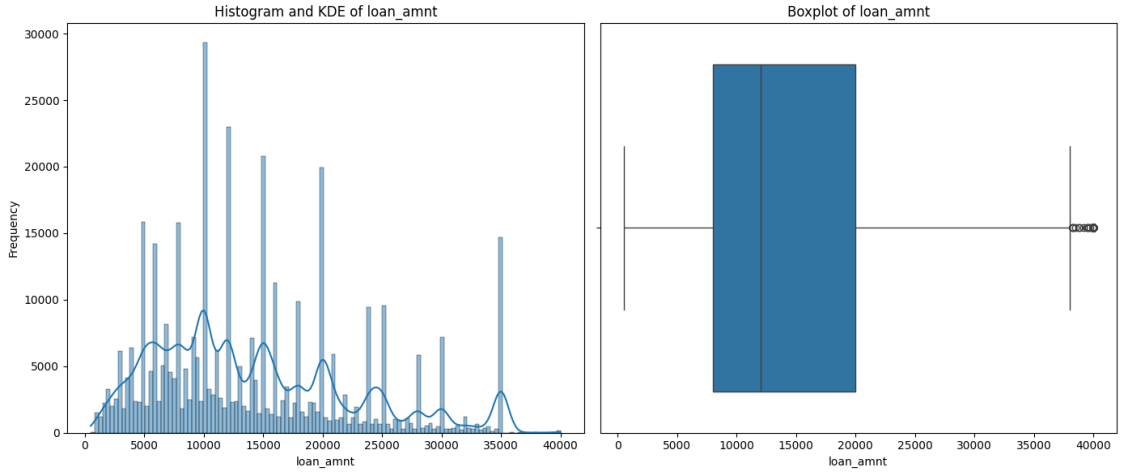
years (9%), and < 1 year (8%).
  - Insight: Most borrowers have a relatively long and stable employment history, which is a positive indicator of financial stability and the ability to repay loans.
- **Home Ownership:**
  - The majority of borrowers have a mortgage (50%), followed by renters (40%), and a smaller percentage of owners (9%).
  - Insight: A large number of borrowers have mortgages, which might suggest that many of them have some level of financial stability but are also leveraging credit for additional financial needs.
- **Verification Status:**
  - The majority of borrowers are "Verified" (35%), followed by "Source Verified" (33%) and "Not Verified" (32%).
  - Insight: A large proportion of borrowers have verified income or other details, suggesting that the dataset may represent a relatively trustworthy sample of applicants.
- **Issue Date:**
  - The issue dates are spread across multiple years, with Oct 2014 being the most common issue month (approximately 3.75% of the data).
  - Insight: The dataset covers a broad range of loan periods, which could provide valuable insights into trends over time, such as changes in loan types, amounts, and interest rates.
- **Loan Status:**
  - Most loans have been "Fully Paid" (80%), while a smaller percentage are "Charged Off" (20%).
  - Insight: The high percentage of fully paid loans suggests that the dataset could primarily consist of borrowers who successfully repaid their loans, which could positively skew risk assessments.
- **Purpose:**
  - The majority of loans are taken for debt consolidation (59%), followed by credit card refinancing (21%).
  - Insight: Debt consolidation is a major reason for taking out loans, indicating that many borrowers may be trying to simplify their finances by consolidating multiple debts into one.
- **Title:**
  - Titles like "Debt Consolidation" and "Credit Card Refinancing" are the most common, reflecting the purposes of the loans.
  - Insight: The similarity in loan titles and purposes (such as debt consolidation) further confirms the idea that borrowers are using these loans to manage or restructure their debts.
- **Initial List Status:**
  - The majority of loans were listed as "f" (fully funded), followed by "w" (waiting for funding).
  - Insight: The preponderance of fully funded loans (80%) indicates that most loans in the dataset were successfully issued.
- **Application Type:**
  - Most applicants applied as "INDIVIDUAL" (99%), with a very small percentage applying as "JOINT" or "DIRECT_PAY".
  - Insight: The high percentage of individual applicants may indicate that personal loans are more common than joint loans in this dataset.
- **City:**

  - The city column has 67,513 unique values, with DPO (14,289), APO (14,060), and FPO (14,035) dominating, indicating a high number of military or government addresses. The distribution is skewed, with most cities appearing only once or a few times.
- **State:**
  - The state column has 54 unique values, with AP (14,308), AE (14,157), and AA (13,919) being the most frequent, indicating a significant number of military or diplomatic addresses. The distribution shows other states appearing in smaller, relatively even proportions.

[17]:
```
for i in df.select_dtypes(include=['number']).columns.tolist():
    print(f"\n{'-'*50}")
    print(f"Analysis for Column: {i}")
    print(f"{'-'*50}")
    skew_value = df[i].skew()
    print(f"Skewness: {skew_value:.4f}")
    kurt_value = df[i].kurt()
    print(f"Kurtosis: {kurt_value:.4f}")
    print("\nTop 10 value counts:")
    print(df[i].value_counts().head(10))
    print("\nBottom 10 value counts:")
    print(df[i].value_counts().tail(10))
    z_scores = zscore(df[i].dropna())
    print(f"\nZ-scores (First 10 values):")
    print(z_scores[:10])
    IQR = df[i].quantile(0.75)-df[i].quantile(0.25)
    print(f"\nIQR value is {IQR}\nUpper Limit/Bound: {df[i].quantile(0.75)+1.
 ↪5*IQR}\nLower Limit/Bound: {df[i].quantile(0.25)-1.5*IQR}\nNumber of rows␣
 ↪which are outliners are {df[(df[i]<df[i].quantile(0.25)-1.5*IQR) |␣
 ↪(df[i]>df[i].quantile(0.75)+1.5*IQR)].shape[0]}\n")
    fig, axes = plt.subplots(1, 2, figsize=(14, 6))
    sns.histplot(df[i], kde=True, ax=axes[0])
    axes[0].set_title(f'Histogram and KDE of {i}')
    axes[0].set_xlabel(i)
    axes[0].set_ylabel('Frequency')
    sns.boxplot(x=df[i], ax=axes[1])
    axes[1].set_title(f'Boxplot of {i}')
    axes[1].set_xlabel(i)
    plt.tight_layout()
    plt.show()
```

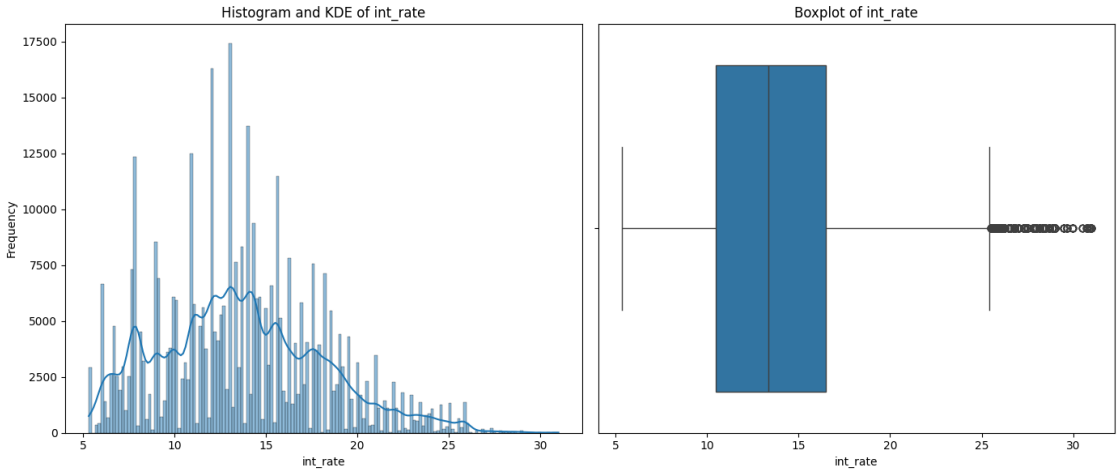
```
--------------------------------------------------
Analysis for Column: loan_amnt
--------------------------------------------------
Skewness: 0.7773
Kurtosis: -0.0626

Top 10 value counts:
```

```
loan_amnt
10000.0    27668
12000.0    21366
15000.0    19903
20000.0    18969
35000.0    14576
8000.0     13539
6000.0     12734
5000.0     12443
16000.0    10129
18000.0     9195
Name: count, dtype: int64

Bottom 10 value counts:
loan_amnt
36525.0    1
36625.0    1
38825.0    1
30050.0    1
39475.0    1
39200.0    1
38750.0    1
36275.0    1
36475.0    1
725.0      1
Name: count, dtype: int64

Z-scores (First 10 values):
0   -0.492243
1   -0.731551
2    0.177819
3   -0.827274
4    1.227783
5    0.704297
6    0.464989
7   -0.133281
8    0.572677
9    1.458117
Name: loan_amnt, dtype: float64

IQR value is 12000.0
Upper Limit/Bound: 38000.0
Lower Limit/Bound: -10000.0
Number of rows which are outliners are 191
```



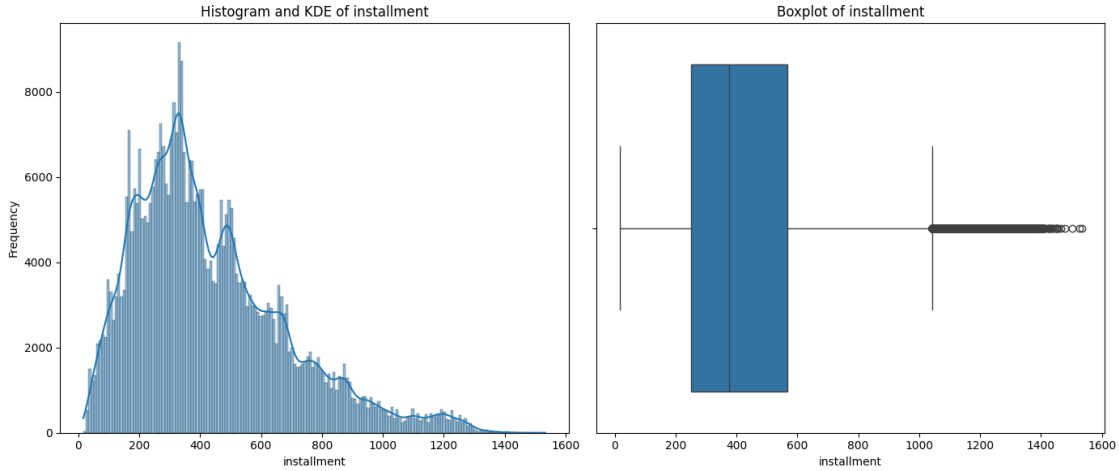

```
--------------------------------------------------
Analysis for Column: int_rate
--------------------------------------------------
Skewness: 0.4207
Kurtosis: -0.1439

Top 10 value counts:
int_rate
10.99    12411
12.99     9632
15.61     9350
11.99     8582
8.90      8019
12.12     7358
7.90      7332
16.29     6632
13.11     6580
6.03      6291
Name: count, dtype: int64

Bottom 10 value counts:
int_rate
14.70    1
17.34    1
16.15    1
18.72    1
18.36    1
14.38    1
24.40    1
22.64    1
```

```
17.54    1
17.44    1
Name: count, dtype: int64

Z-scores (First 10 values):
0   -0.491799
1   -0.368816
2   -0.704225
3   -1.598649
4    0.811824
5   -0.069184
6   -1.860268
7   -0.558881
8   -0.592422
9    0.592690
Name: int_rate, dtype: float64

IQR value is 5.999999999999998
Upper Limit/Bound: 25.489999999999995
Lower Limit/Bound: 1.4900000000000038
Number of rows which are outliners are 3777
```



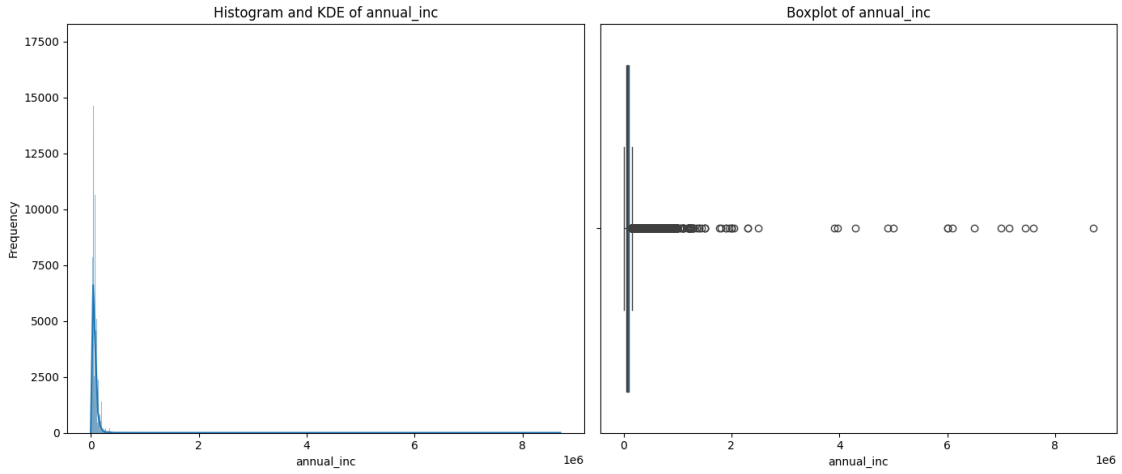

```
--------------------------------------------------
Analysis for Column: installment
--------------------------------------------------
Skewness: 0.9836
Kurtosis: 0.7838

Top 10 value counts:
```

```
installment
327.34    968
332.10    791
491.01    736
336.90    686
392.81    683
332.72    641
337.47    624
317.54    574
654.68    556
261.88    527
Name: count, dtype: int64

Bottom 10 value counts:
installment
144.05     1
430.13     1
301.13     1
281.69     1
395.90     1
1146.14    1
218.49     1
961.66     1
569.10     1
555.96     1
Name: count, dtype: int64

Z-scores (First 10 values):
0   -0.408291
1   -0.662750
2    0.299609
3   -0.842348
4    0.707861
5    0.978035
6    0.439602
7   -0.021456
8   -0.083795
9    1.980438
Name: installment, dtype: float64

IQR value is 316.9699999999999
Upper Limit/Bound: 1042.7549999999999
Lower Limit/Bound: -225.12499999999986
Number of rows which are outliners are 11250
```



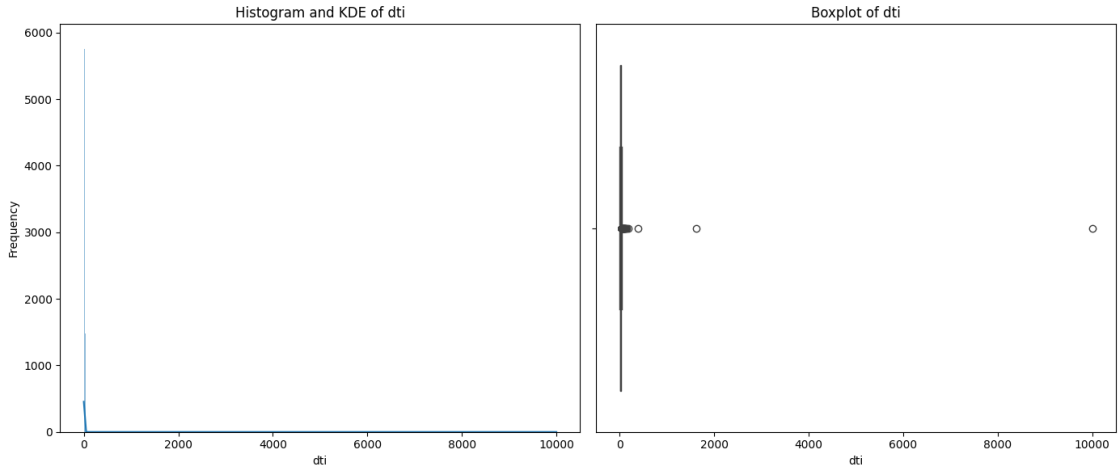

```
--------------------------------------------------
Analysis for Column: annual_inc
--------------------------------------------------
Skewness: 41.0427
Kurtosis: 4238.5506

Top 10 value counts:
annual_inc
60000.0    15313
50000.0    13303
65000.0    11333
70000.0    10674
40000.0    10629
45000.0    10114
80000.0     9971
75000.0     9850
55000.0     9195
90000.0     7573
Name: count, dtype: int64

Bottom 10 value counts:
annual_inc
68455.0     1
745000.0    1
341120.0    1
63908.0     1
49043.0     1
42558.0     1
29899.0     1
25837.2     1
```

```
128647.0    1
23085.0     1
Name: count, dtype: int64

Z-scores (First 10 values):
0    0.694330
1   -0.149311
2   -0.505312
3   -0.327774
4   -0.311550
5    0.204175
6    0.824121
7   -0.457565
8    0.467196
9    0.661883
Name: annual_inc, dtype: float64

IQR value is 45000.0
Upper Limit/Bound: 157500.0
Lower Limit/Bound: -22500.0
Number of rows which are outliners are 16700
```



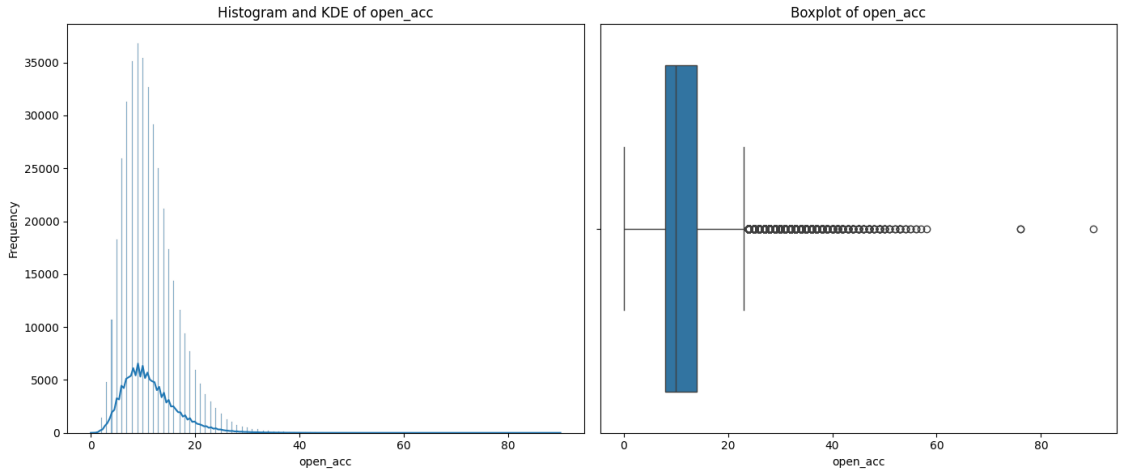

```
--------------------------------------------------
Analysis for Column: dti
--------------------------------------------------
Skewness: 431.0512
Kurtosis: 237923.6765

Top 10 value counts:
```

```
dti
0.0     313
14.4    310
19.2    302
16.8    301
18.0    300
20.4    296
12.0    293
13.2    291
21.6    270
15.6    266
Name: count, dtype: int64

Bottom 10 value counts:
dti
45.15    1
42.30    1
54.90    1
45.72    1
68.30    1
41.38    1
49.83    1
46.32    1
43.98    1
40.61    1
Name: count, dtype: int64

Z-scores (First 10 values):
0    0.491728
1    0.259197
2   -0.254703
3   -0.820215
4    0.919608
5   -0.059355
6   -0.889031
7    0.526691
8   -0.269687
9    0.350212
Name: dti, dtype: float64

IQR value is 11.700000000000001
Upper Limit/Bound: 40.53
Lower Limit/Bound: -6.270000000000001
Number of rows which are outliners are 275
```



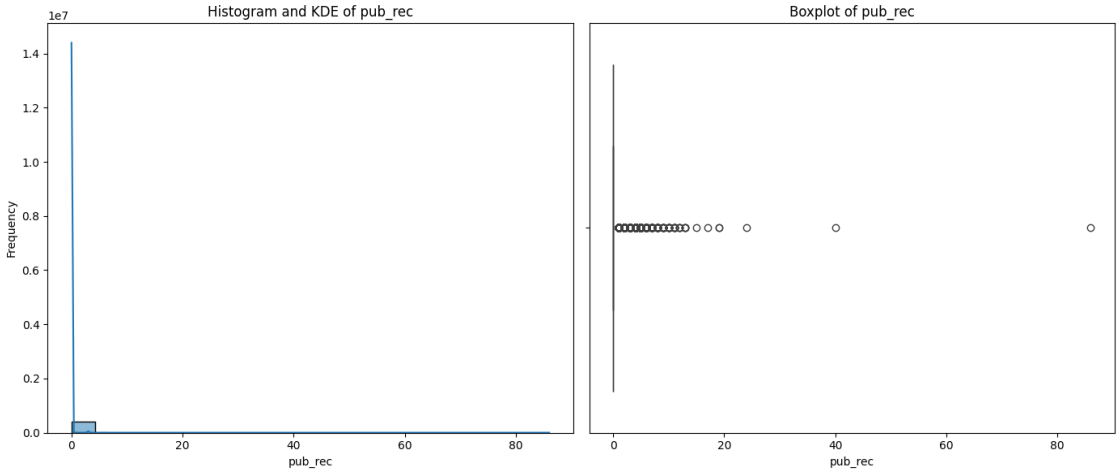

```
--------------------------------------------------
Analysis for Column: open_acc
--------------------------------------------------
Skewness: 1.2130
Kurtosis: 2.9669

Top 10 value counts:
open_acc
9.0     36779
10.0    35441
8.0     35137
11.0    32695
7.0     31328
12.0    29157
6.0     25927
13.0    24983
14.0    21173
5.0     18308
Name: count, dtype: int64

Bottom 10 value counts:
open_acc
50.0    6
51.0    4
54.0    3
52.0    3
76.0    2
56.0    2
55.0    2
57.0    1
```

```
58.0    1
90.0    1
Name: count, dtype: int64

Z-scores (First 10 values):
0    0.912646
1    1.107287
2    0.328720
3   -1.033772
4    0.328720
5   -0.644489
6   -0.644489
7   -0.060563
8    0.328720
9    0.328720
Name: open_acc, dtype: float64

IQR value is 6.0
Upper Limit/Bound: 23.0
Lower Limit/Bound: -1.0
Number of rows which are outliners are 10307
```



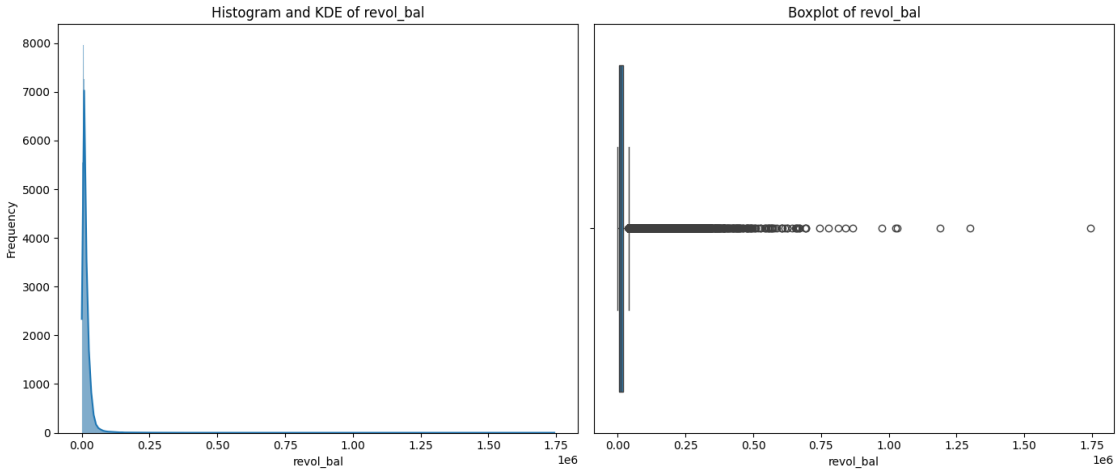

```
--------------------------------------------------
Analysis for Column: pub_rec
--------------------------------------------------
Skewness: 16.5766
Kurtosis: 1867.4666

Top 10 value counts:
```

```
pub_rec
0.0    338272
1.0     49739
2.0      5476
3.0      1521
4.0       527
5.0       237
6.0       122
7.0        56
8.0        34
9.0        12
Name: count, dtype: int64

Bottom 10 value counts:
pub_rec
10.0    11
11.0     8
13.0     4
12.0     4
19.0     2
40.0     1
17.0     1
86.0     1
24.0     1
15.0     1
Name: count, dtype: int64

Z-scores (First 10 values):
0   -0.335785
1   -0.335785
2   -0.335785
3   -0.335785
4   -0.335785
5   -0.335785
6   -0.335785
7   -0.335785
8   -0.335785
9   -0.335785
Name: pub_rec, dtype: float64

IQR value is 0.0
Upper Limit/Bound: 0.0
Lower Limit/Bound: 0.0
Number of rows which are outliners are 57758
```



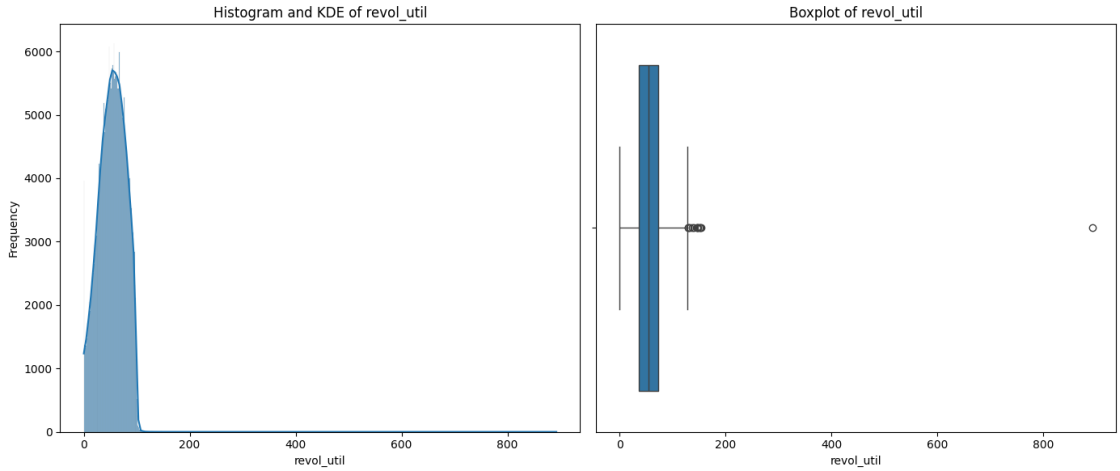

```
--------------------------------------------------
Analysis for Column: revol_bal
--------------------------------------------------
Skewness: 11.7275
Kurtosis: 384.2211

Top 10 value counts:
revol_bal
0.0       2128
5655.0      41
7792.0      38
6095.0      38
3953.0      37
5098.0      36
6077.0      36
8502.0      35
6521.0      35
5235.0      35
Name: count, dtype: int64

Bottom 10 value counts:
revol_bal
321205.0    1
52972.0     1
25021.0     1
73328.0     1
40296.0     1
147559.0    1
50316.0     1
222641.0    1
```

```
568659.0    1
57725.0     1
Name: count, dtype: int64

Z-scores (First 10 values):
0    0.996729
1    0.208163
2   -0.187334
3   -0.503722
4    0.424414
5    0.481379
6   -0.566562
7   -0.117500
8    0.135610
9    0.307232
Name: revol_bal, dtype: float64

IQR value is 13595.0
Upper Limit/Bound: 40012.5
Lower Limit/Bound: -14367.5
Number of rows which are outliners are 21259
```



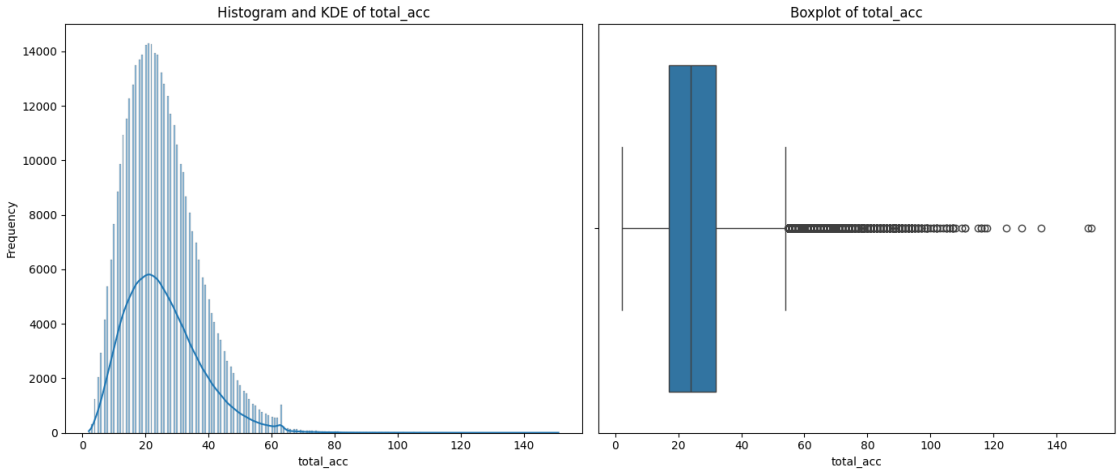

```
--------------------------------------------------
Analysis for Column: revol_util
--------------------------------------------------
Skewness: -0.0718
Kurtosis: 2.7123

Top 10 value counts:
```

```
revol_util
0.0     2213
53.0     752
60.0     739
61.0     734
55.0     730
54.0     725
62.0     721
47.0     720
57.0     719
58.0     717
Name: count, dtype: int64

Bottom 10 value counts:
revol_util
10.61     1
129.50    1
108.60    1
111.10    1
0.75      1
146.10    1
109.30    1
108.10    1
115.30    1
37.63     1
Name: count, dtype: int64

Z-scores (First 10 values):
0   -0.490417
1   -0.020111
2    1.570751
3   -1.320609
4    0.654676
5    1.914279
6   -1.999486
7    0.437927
8   -0.854393
9    1.169968
Name: revol_util, dtype: float64

IQR value is 37.10000000000001
Upper Limit/Bound: 128.55
Lower Limit/Bound: -19.850000000000016
Number of rows which are outliners are 12
```



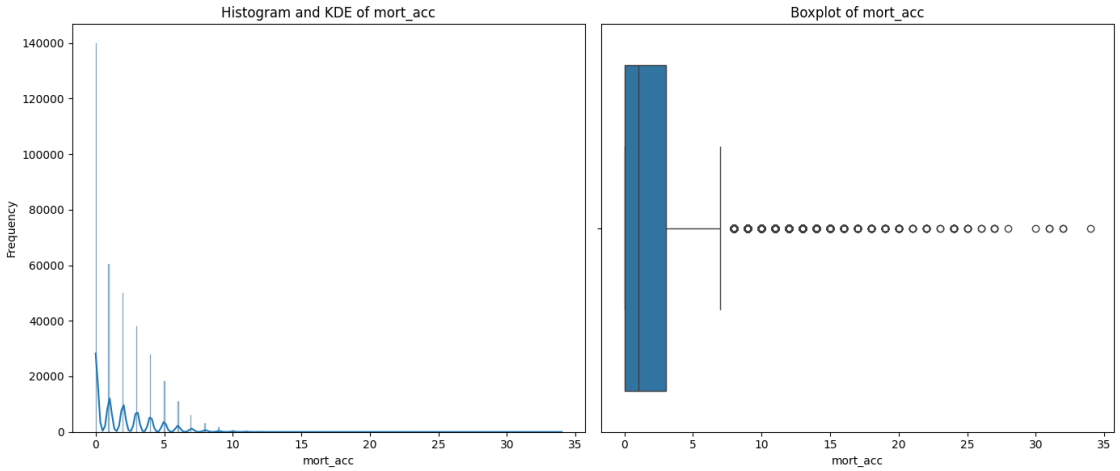

```
--------------------------------------------------
Analysis for Column: total_acc
--------------------------------------------------
Skewness: 0.8643
Kurtosis: 1.2046

Top 10 value counts:
total_acc
21.0    14280
22.0    14260
20.0    14228
23.0    13923
24.0    13878
19.0    13876
18.0    13710
17.0    13495
25.0    13225
26.0    12799
Name: count, dtype: int64

Bottom 10 value counts:
total_acc
110.0    1
129.0    1
118.0    1
151.0    1
124.0    1
150.0    1
117.0    1
115.0    1
```

```
100.0    1
103.0    1
Name: count, dtype: int64

Z-scores (First 10 values):
0   -0.034891
1    0.133361
2    0.049235
3   -1.044399
4    1.479372
5   -0.203142
6   -0.034891
7   -0.876147
8    1.226995
9    0.974618
Name: total_acc, dtype: float64

IQR value is 15.0
Upper Limit/Bound: 54.5
Lower Limit/Bound: -5.5
Number of rows which are outliners are 8499
```



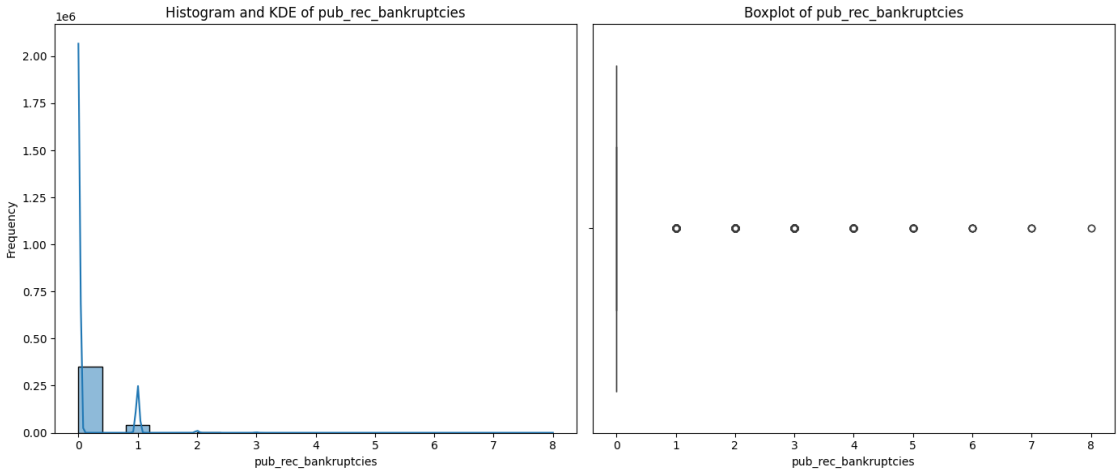

```
--------------------------------------------------
Analysis for Column: mort_acc
--------------------------------------------------
Skewness: 1.6001
Kurtosis: 4.4772

Top 10 value counts:
```

```
mort_acc
0.0    139777
1.0     60416
2.0     49948
3.0     38049
4.0     27887
5.0     18194
6.0     11069
7.0      6052
8.0      3121
9.0      1656
Name: count, dtype: int64

Bottom 10 value counts:
mort_acc
21.0    4
25.0    4
27.0    3
26.0    2
32.0    2
31.0    2
23.0    2
34.0    1
28.0    1
30.0    1
Name: count, dtype: int64

Z-scores (First 10 values):
0   -0.844531
1    0.552164
2   -0.844531
3   -0.844531
4   -0.378966
5    1.017730
6    0.552164
7   -0.844531
8    0.552164
9   -0.378966
Name: mort_acc, dtype: float64

IQR value is 3.0
Upper Limit/Bound: 7.5
Lower Limit/Bound: -4.5
Number of rows which are outliners are 6843
```



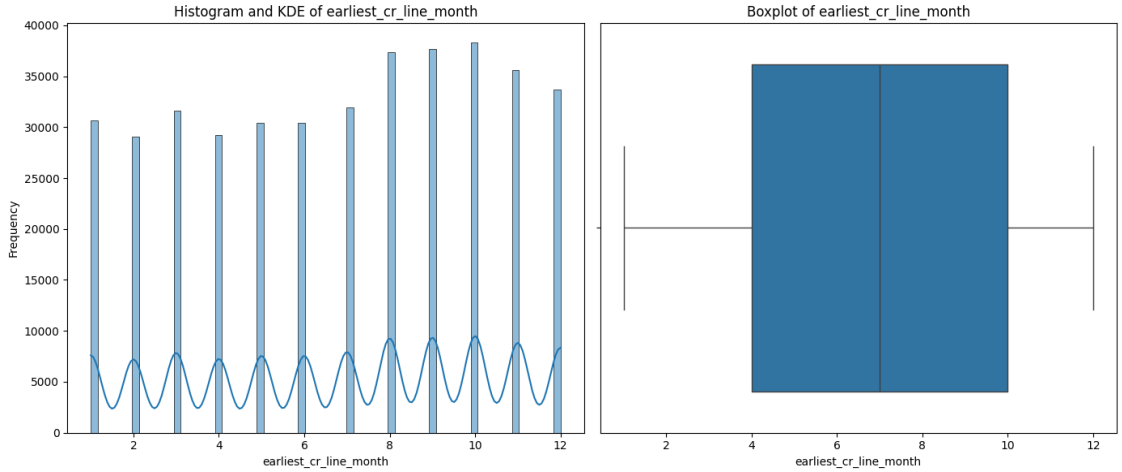

```
--------------------------------------------------
Analysis for Column: pub_rec_bankruptcies
--------------------------------------------------
Skewness: 3.4234
Kurtosis: 18.1042

Top 10 value counts:
pub_rec_bankruptcies
0.0    350380
1.0     42790
2.0      1847
3.0       351
4.0        82
5.0        32
6.0         7
7.0         4
8.0         2
Name: count, dtype: int64

Bottom 10 value counts:
pub_rec_bankruptcies
0.0    350380
1.0     42790
2.0      1847
3.0       351
4.0        82
5.0        32
6.0         7
7.0         4
8.0         2
```

```
Name: count, dtype: int64

Z-scores (First 10 values):
0   -0.34154
1   -0.34154
2   -0.34154
3   -0.34154
4   -0.34154
5   -0.34154
6   -0.34154
7   -0.34154
8   -0.34154
9   -0.34154
Name: pub_rec_bankruptcies, dtype: float64

IQR value is 0.0
Upper Limit/Bound: 0.0
Lower Limit/Bound: 0.0
Number of rows which are outliners are 45115
```



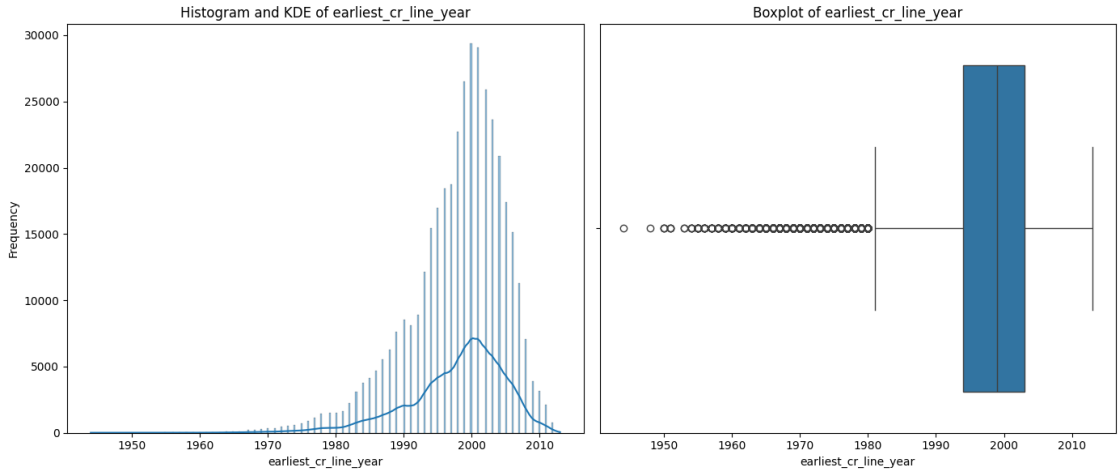

```
--------------------------------------------------
Analysis for Column: earliest_cr_line_month
--------------------------------------------------
Skewness: -0.1285
Kurtosis: -1.2020

Top 10 value counts:
earliest_cr_line_month
10    38291
```

```
9     37673
8     37349
11    35583
12    33687
7     31972
3     31617
1     30694
6     30445
5     30445
Name: count, dtype: int64

Bottom 10 value counts:
earliest_cr_line_month
8     37349
11    35583
12    33687
7     31972
3     31617
1     30694
6     30445
5     30445
4     29231
2     29043
Name: count, dtype: int64

Z-scores (First 10 values):
0   -0.220154
1    0.070966
2    0.362086
3    0.653207
4   -1.093515
5   -1.675755
6    0.362086
7    0.653207
8   -0.220154
9    1.526568
Name: earliest_cr_line_month, dtype: float64

IQR value is 6.0
Upper Limit/Bound: 19.0
Lower Limit/Bound: -5.0
Number of rows which are outliners are 0
```



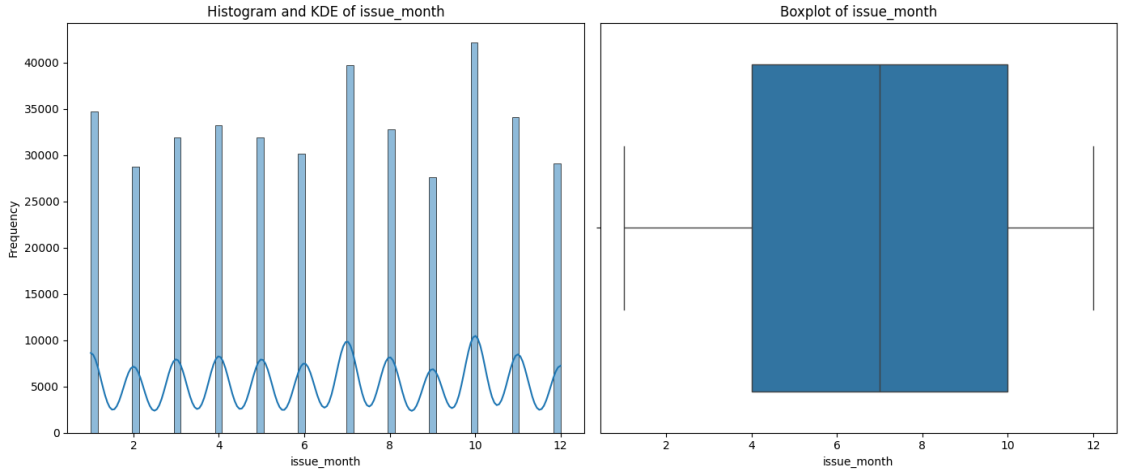

```
--------------------------------------------------
Analysis for Column: earliest_cr_line_year
--------------------------------------------------
Skewness: -1.0736
Kurtosis: 1.7421

Top 10 value counts:
earliest_cr_line_year
2000    29366
2001    29083
1999    26491
2002    25901
2003    23657
1998    22745
2004    20914
1997    18761
1996    18413
2005    17401
Name: count, dtype: int64

Bottom 10 value counts:
earliest_cr_line_year
1958    12
1955     9
1957     7
1956     7
1954     4
1950     3
1951     3
1953     2
```

```
1944     1
1948     1
Name: count, dtype: int64

Z-scores (First 10 values):
0   -1.091589
1    0.853294
2    1.270055
3    1.131134
4    0.158693
5    0.992214
6    0.992214
7   -0.535908
8   -0.535908
9   -0.119147
Name: earliest_cr_line_year, dtype: float64

IQR value is 9.0
Upper Limit/Bound: 2016.5
Lower Limit/Bound: 1980.5
Number of rows which are outliners are 10629
```



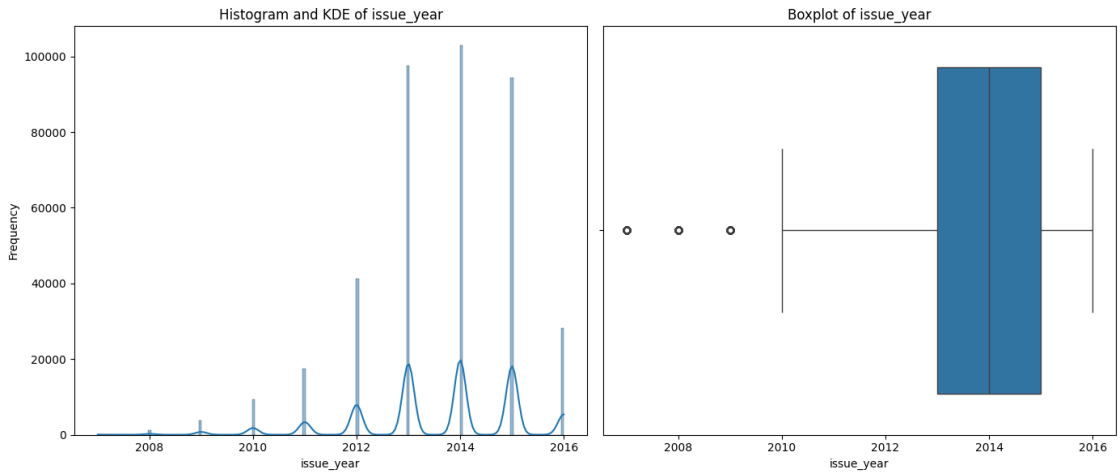

```
--------------------------------------------------
Analysis for Column: issue_month
--------------------------------------------------
Skewness: -0.0523
Kurtosis: -1.2047

Top 10 value counts:
```

```
issue_month
10    42130
7     39714
1     34682
11    34068
4     33223
8     32816
3     31919
5     31895
6     30140
12    29082
Name: count, dtype: int64

Bottom 10 value counts:
issue_month
1     34682
11    34068
4     33223
8     32816
3     31919
5     31895
6     30140
12    29082
2     28742
9     27619
Name: count, dtype: int64

Z-scores (First 10 values):
0   -1.620603
1   -1.620603
2   -1.620603
3    1.297726
4   -0.745104
5    0.714060
6    0.714060
7    0.714060
8    1.005893
9   -0.745104
Name: issue_month, dtype: float64

IQR value is 6.0
Upper Limit/Bound: 19.0
Lower Limit/Bound: -5.0
Number of rows which are outliners are 0
```




```
--------------------------------------------------
Analysis for Column: issue_year
--------------------------------------------------
Skewness: -0.7737
Kurtosis: 0.8477

Top 10 value counts:
issue_year
2014    102860
2013     97662
2015     94264
2012     41202
2016     28088
2011     17435
2010      9258
2009      3826
2008      1240
2007       195
Name: count, dtype: int64

Bottom 10 value counts:
issue_year
2014    102860
2013     97662
2015     94264
2012     41202
2016     28088
2011     17435
2010      9258
2009      3826
```

```
2008      1240
2007       195
Name: count, dtype: int64

Z-scores (First 10 values):
0    0.925224
1    0.925224
2    0.925224
3    0.250334
4   -0.424555
5    0.925224
6    0.925224
7   -1.099445
8    0.250334
9   -1.099445
Name: issue_year, dtype: float64

IQR value is 2.0
Upper Limit/Bound: 2018.0
Lower Limit/Bound: 2010.0
Number of rows which are outliners are 5261
```



Analysis and Insights (Numerical Columns)

- **Loan Amount (loan_amnt)**
  - Skewness: The data is positively skewed, meaning most loan amounts are towards the lower range (closer to the minimum).
  - Top Values: Loan amounts around 10,000 to 20,000 are the most common.
  - Outliers: A few loan amounts (like 36,525, 39,475) appear only once, indicating these are rare or erroneous entries.

  - Z-Scores: Most values fall within typical ranges, with some significant deviations indicating potential outliers.
- **Interest Rate (int_rate)**
  - Skewness: Slight positive skew, meaning there are more loans with lower interest rates.
  - Top Values: The most frequent interest rates are in the 10-16% range.
  - Outliers: A few unique high interest rates (such as 24.40%) indicate possible data anomalies.
  - Z-Scores: Most values are within a reasonable range, though a few exhibit large negative or positive deviations.
- **Installment Amount (installment)**
  - Skewness: Positive skew, meaning most installment values are lower.
  - Top Values: Common installment amounts range from 300 to 600.
  - Outliers: A few very high installment amounts (e.g., 1146.14) that may be errors or represent high-loan amounts with high-interest rates.
  - Z-Scores: Some values (e.g., 1.98) indicate extreme deviations that may require further investigation.
- **Annual Income (annual_inc)**
  - Skewness: Extremely high skewness, with most borrowers earning below average but some extreme high-income values.
  - Top Values: Common annual incomes are around 50,000 to 80,000.
  - Outliers: Many extreme high-income values (e.g., 745,000) are likely outliers or errors in data entry.
  - Z-Scores: Large variation, with significant deviations indicating possible data errors or outliers.
- **Debt-to-Income Ratio (dti)**
  - Skewness: Extremely high skewness, with most values clustered at the lower end (around 0-20%).
  - Top Values: Most common values are multiples of 1.2, such as 14.4, 16.8, etc.
  - Outliers: High DTI ratios (e.g., 45.15) indicate potential outliers.
  - Z-Scores: Indicates a few unusual values, suggesting potential data issues or outliers that require correction.
- **Number of Open Accounts (open_acc)**
  - Skewness: Slightly positive skew, indicating most individuals have around 9 to 14 open accounts.
  - Top Values: Common account numbers are between 5 and 14.
  - Outliers: Very few people have an unusually high number of open accounts (e.g., 90).
  - Z-Scores: Most values are within a reasonable range, with only a few extremes.
- **Public Records (pub_rec)**
  - Skewness: Highly skewed, as most individuals have no public records.
  - Top Values: 0 (no public records) is the most frequent value, with 1 and 2 being common as well.
  - Outliers: Some individuals have multiple public records, though these are rare.
  - Z-Scores: Most values are clustered around zero, but the distribution suggests many zeros and some rare high counts.
- **Revolving Balance (revol_bal)**
  - Skewness: Extremely positive skew, meaning most individuals have a small revolving balance.
  - Top Values: Many entries have a balance of 0, with others ranging from 5,000 to 10,000.

  - Outliers: Some high revolving balances (e.g., 568,659) are rare and likely indicate data errors or rare extreme cases.
  - Z-Scores: There are a few extreme outliers, suggesting potential errors or misreported data.
- **Revolving Utilization (revol_util)**
  - Skewness: Slight negative skew, with most revolving utilization ratios around 50%.
  - Top Values: Most common values are 53, 60, 61%, with some slightly deviating percentages (e.g., 37.63).
  - Outliers: A few extreme values (e.g., 146.1%) are likely errors or unusual cases.
  - Z-Scores: Indicates that most values fall within normal ranges, with a few extreme deviations.
- **Total Accounts (total_acc)**
  - Skewness: Slightly positive skew, with most individuals having between 17 and 26 total accounts.
  - Top Values: Most common account numbers are around 20 to 26.
  - Outliers: A few outliers with very high numbers of total accounts (e.g., 118, 151).
  - Z-Scores: There are some significant deviations from the mean, suggesting a few individuals have exceptionally high or low account counts.
- **Mortgage Accounts (mort_acc)**
  - Skewness: Positive skew, with most individuals having no mortgage accounts.
  - Top Values: Most frequent counts are 0, 1, and 2 mortgage accounts.
  - Outliers: Few individuals have more than 10 mortgage accounts, which might be rare or indicate special cases.
  - Z-Scores: Generally clustered around zero, with a few outliers indicating unusual mortgage patterns.
- **Public Record Bankruptcies (pub_rec_bankruptcies)**
  - Skewness: Highly skewed, with most individuals having no bankruptcies.
  - Top Values: Most common is 0 (no bankruptcies), followed by 1 bankruptcy.
  - Outliers: Few individuals have more than 1 bankruptcy, suggesting rare extreme cases.
  - Z-Scores: Most values are clustered near zero, indicating that most borrowers have no bankruptcies, with a few exceptions.
- **earliest_cr_line_month:**
  - Skewness: -0.1285 (Slightly left-skewed distribution, indicating the values are relatively balanced but slightly weighted towards earlier months.)
  - Kurtosis: -1.2020 (Platykurtic distribution, meaning the data is flatter and has lighter tails than a normal distribution.)
  - Top 10 Frequent Months: The most common months are October (38,291), September (37,673), and August (37,349). These months suggest seasonal trends in the earliest credit line dates, possibly due to financial cycles or borrower behaviors.
  - Bottom 10 Frequent Months: February (29,043) and April (29,231) are the least frequent months, indicating fewer credit lines originated during these periods.
  - Z-Scores: The first 10 z-scores indicate some variation around the mean, with values like 1.5266 showing above-average occurrences and -1.6757 showing below-average occurrences, highlighting moderate dispersion.
- **earliest_cr_line_year:**
  - Skewness: -1.0736 (The distribution is moderately left-skewed, meaning there are more recent years with earlier credit lines compared to older years.)
  - Kurtosis: 1.7421 (Leptokurtic distribution, indicating a higher peak and heavier tails

than a normal distribution, suggesting a concentration of data around a central point with outliers at both ends.)

- Top 10 Frequent Years: The most common years for earliest credit lines are from 2000 to 2003, with 2000 (29,366) and 2001 (29,083) being the most frequent. This suggests a concentration of credit lines issued around the early 2000s, possibly due to a financial boom or lending pattern during that period.
- Bottom 10 Frequent Years: The least frequent years are from the 1950s, with 1958 (12), 1955 (9), and others having very few occurrences. This likely indicates a very small number of older credit lines from individuals whose financial activity was reported long after the initial credit establishment.
- Z-Scores: The first 10 z-scores show moderate variations around the mean, with values like 1.2700 indicating years with higher occurrences and -1.0916 showing below-average frequencies for specific years. These variations suggest that while most data is concentrated in more recent years, some older years have relatively few data points.

2.3 Bivariate Analysis

[18]:
```
df.head()
```

[18]:

	loan_amnt	term	int_rate	installment	grade	sub_grade	\
0	10000.0	36 months	11.44	329.48	B	B4	
1	8000.0	36 months	11.99	265.68	B	B5	
2	15600.0	36 months	10.49	506.97	B	B3	
3	7200.0	36 months	6.49	220.65	A	A2	
4	24375.0	60 months	17.27	609.33	C	C5	

	emp_title	emp_length	home_ownership	annual_inc	…	\
0	Marketing	10+ years	RENT	117000.0	…	
1	Credit analyst	4 years	MORTGAGE	65000.0	…	
2	Statistician	< 1 year	RENT	43057.0	…	
3	Client Advocate	6 years	RENT	54000.0	…	
4	Destiny Management Inc.	9 years	MORTGAGE	55000.0	…	

	initial_list_status	application_type	mort_acc	pub_rec_bankruptcies	\
0	w	INDIVIDUAL	0.0	0.0	
1	f	INDIVIDUAL	3.0	0.0	
2	f	INDIVIDUAL	0.0	0.0	
3	f	INDIVIDUAL	0.0	0.0	
4	f	INDIVIDUAL	1.0	0.0	

	city	state	earliest_cr_line_month	earliest_cr_line_year	\
0	Mendozaberg	OK	6	1990	
1	Loganmouth	SD	7	2004	
2	New Sabrina	WV	8	2007	
3	Delacruzside	MA	9	2006	
4	Greggshire	VA	3	1999	

```
   issue_month  issue_year
0            1        2015
1            1        2015
2            1        2015
3           11        2014
4            4        2013

[5 rows x 30 columns]
```

[19]:
```
df[df.select_dtypes(include=['object']).columns.tolist()].head()
```

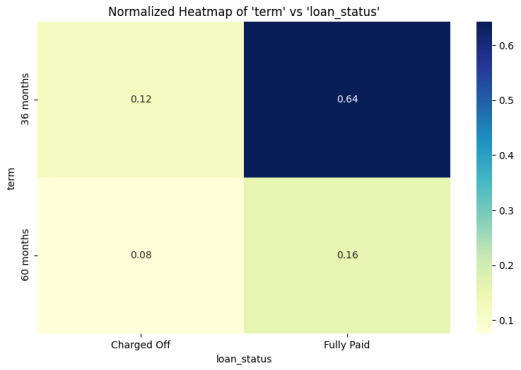
[19]:
```
         term grade sub_grade                emp_title emp_length  \
0   36 months     B        B4                Marketing  10+ years
1   36 months     B        B5          Credit analyst     4 years
2   36 months     B        B3             Statistician   < 1 year
3   36 months     A        A2          Client Advocate     6 years
4   60 months     C        C5  Destiny Management Inc.     9 years

  home_ownership verification_status  loan_status             purpose  \
0           RENT        Not Verified   Fully Paid            vacation
1       MORTGAGE        Not Verified   Fully Paid  debt_consolidation
2           RENT     Source Verified   Fully Paid         credit_card
3           RENT        Not Verified   Fully Paid         credit_card
4       MORTGAGE            Verified  Charged Off         credit_card

                    title initial_list_status application_type          city  \
0                Vacation                    w       INDIVIDUAL   Mendozaberg
1      Debt consolidation                    f       INDIVIDUAL    Loganmouth
2  Credit card refinancing                   f       INDIVIDUAL   New Sabrina
3  Credit card refinancing                   f       INDIVIDUAL  Delacruzside
4    Credit Card Refinance                   f       INDIVIDUAL    Greggshire

  state
0    OK
1    SD
2    WV
3    MA
4    VA
```

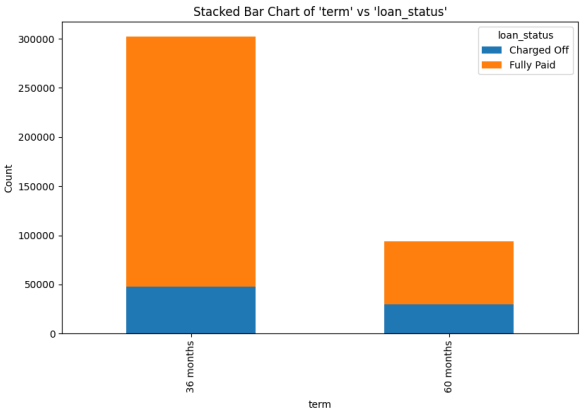
We know that the target value is loan_status where 'Fully Paid' will be 1 and 'Charged Off' will be 0

[20]:
```
categorical_columns = df.select_dtypes(include=['object']).columns.tolist()
for i in categorical_columns:
    j = 'loan_status'
    if i == j:
        continue
    if df[i].nunique() > 15 or df[j].nunique() > 15:
```

```
        print(f"\n{'='*60}")
        print(f"Skipping charts for '{i}' and '{j}' because one of them has␣
↪more than 15 unique values.")
        print(f"{'='*60}\n")
        crosstab_result = pd.crosstab(df[i], df[j])
        display(crosstab_result)
        continue
    crosstab_result = pd.crosstab(df[i], df[j])
    crosstab_normalized = pd.crosstab(df[i], df[j], normalize='all')
    print(f"\n{'='*60}")
    print(f" Crosstab between '{i}' and '{j}' ")
    print(f"{'='*60}\n")
    display(crosstab_result)
    fig, axes = plt.subplots(1, 2, figsize=(16, 6))
    sns.heatmap(crosstab_normalized, annot=True, cmap="YlGnBu", fmt=".2f",␣
↪ax=axes[0])
    axes[0].set_title(f"Normalized Heatmap of '{i}' vs '{j}'")
    axes[0].set_xlabel(j)
    axes[0].set_ylabel(i)
    crosstab_result.plot(kind='bar', stacked=True, ax=axes[1])
    axes[1].set_title(f"Stacked Bar Chart of '{i}' vs '{j}'")
    axes[1].set_xlabel(i)
    axes[1].set_ylabel("Count")
    axes[1].legend(title=j)
    plt.tight_layout()
    plt.show()
```

```
============================================================
 Crosstab between 'term' and 'loan_status'
============================================================


loan_status  Charged Off  Fully Paid
term
36 months          47640      254365
60 months          30033       63992
```



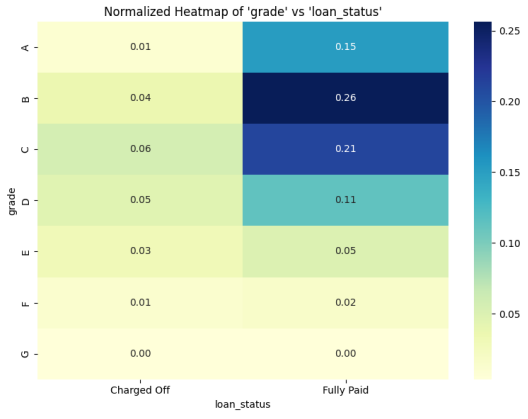


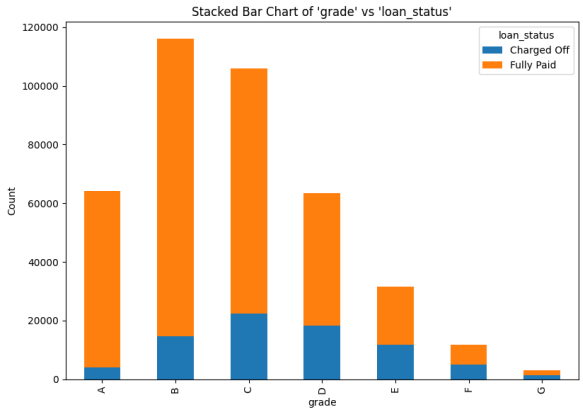
==========

Crosstab between 'grade' and 'loan_status'

==========

loan_status	Charged Off	Fully Paid
grade		
A	4036	60151
B	14587	101431
C	22449	83538
D	18338	45186
E	11765	19723
F	5037	6735
G	1461	1593





==========

Skipping charts for 'sub_grade' and 'loan_status' because one of them has more

than 15 unique values.
==

loan_status	Charged Off	Fully Paid
sub_grade		
A1	279	9450
A2	461	9106
A3	614	9962
A4	1109	14680
A5	1573	16953
B1	1891	17291
B2	2441	20054
B3	3288	23367
B4	3543	22058
B5	3424	18661
C1	4110	19552
C2	4460	18120
C3	4635	16586
C4	4773	15507
C5	4471	13773
D1	4219	11774
D2	3911	10040
D3	3474	8749
D4	3629	8028
D5	3105	6595
E1	2724	5193
E2	2730	4701
E3	2361	3846
E4	2107	3254
E5	1843	2729
F1	1370	2166
F2	1175	1591
F3	997	1289
F4	815	972
F5	680	717
G1	488	570
G2	364	390
G3	282	270
G4	168	206
G5	159	157

==
Skipping charts for 'emp_title' and 'loan_status' because one of them has more than 15 unique values.
==

loan_status	Charged Off	Fully Paid
emp_title		
NSA Industries llc	0	1
Fibro Source	0	1
Long Ilsand College Hospital	1	0
mortgage banker	1	0
Credit rev specialist	1	0
…	…	…
zozaya officiating	1	0
zs backroom	0	1
zueck transportation	0	1
zulily	1	0
License Compliance Investigator	0	1

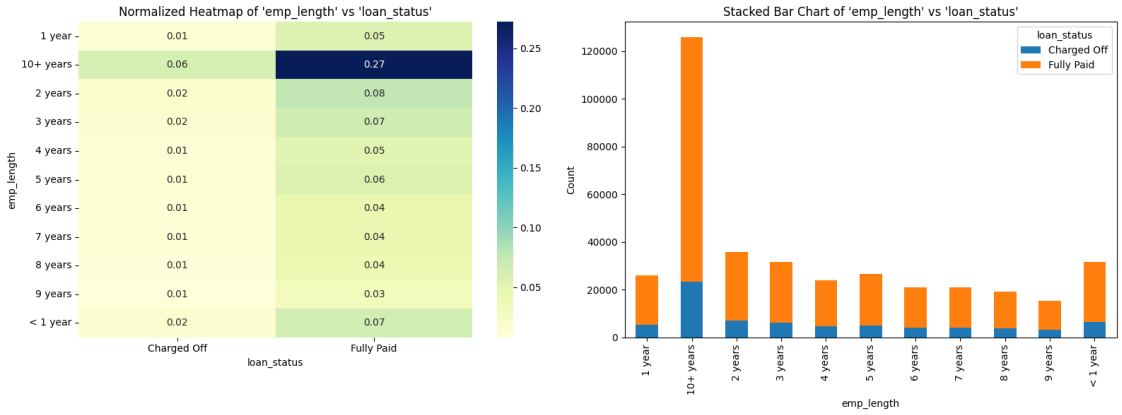
[173105 rows x 2 columns]

==========

Crosstab between 'emp_length' and 'loan_status'

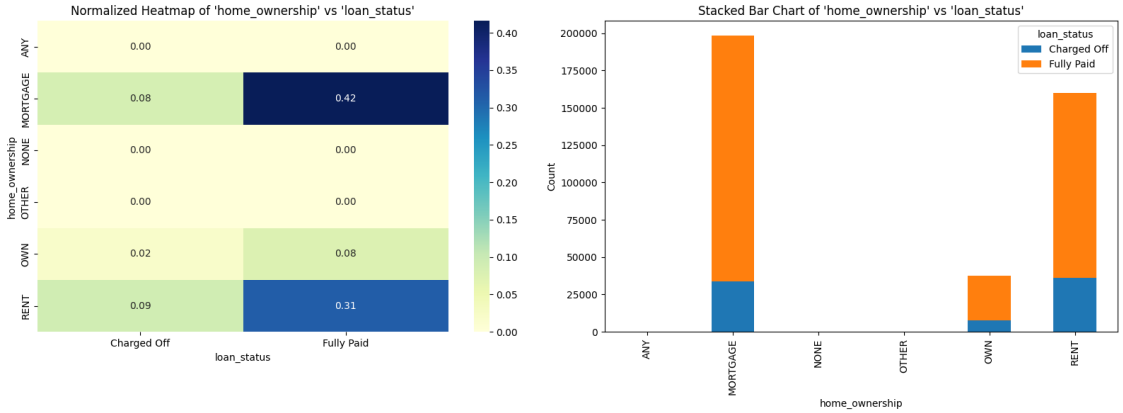
==========

loan_status	Charged Off	Fully Paid
emp_length		
1 year	5154	20728
10+ years	23215	102826
2 years	6924	28903
3 years	6182	25483
4 years	4608	19344
5 years	5092	21403
6 years	3943	16898
7 years	4055	16764
8 years	3829	15339
9 years	3070	12244
< 1 year	6563	25162



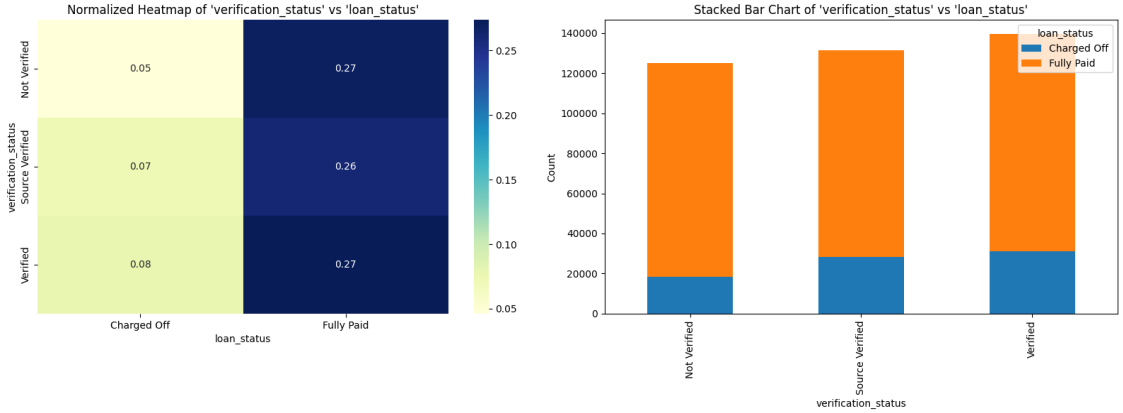
```
============================================================
 Crosstab between 'home_ownership' and 'loan_status'
============================================================
```

loan_status	Charged Off	Fully Paid
home_ownership		
ANY	0	3
MORTGAGE	33632	164716
NONE	7	24
OTHER	16	96
OWN	7806	29940
RENT	36212	123578



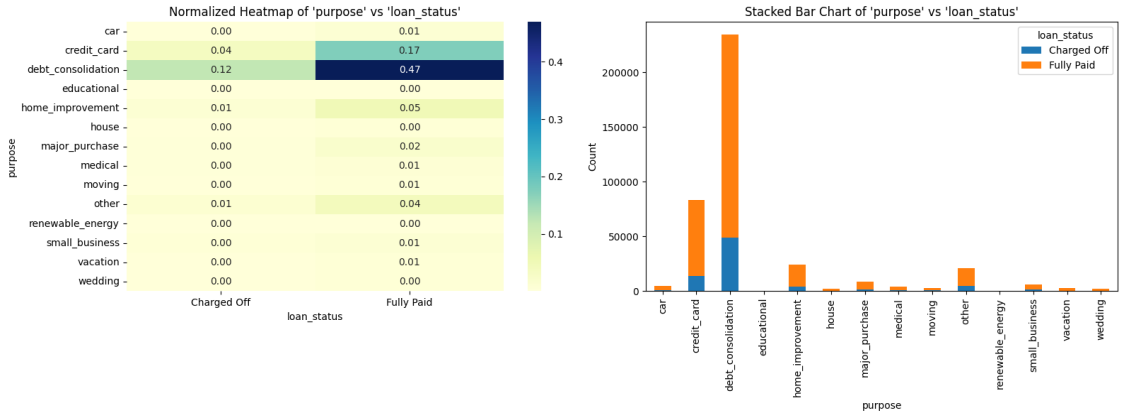

```
============================================================
 Crosstab between 'verification_status' and 'loan_status'
============================================================
```

loan_status	Charged Off	Fully Paid
verification_status		
Not Verified	18307	106775
Source Verified	28214	103171
Verified	31152	108411




```
============================================================
 Crosstab between 'purpose' and 'loan_status'
============================================================
```

loan_status	Charged Off	Fully Paid
purpose		
car	633	4064
credit_card	13874	69145
debt_consolidation	48640	185867
educational	42	215
home_improvement	4087	19943
house	434	1767
major_purchase	1448	7342
medical	911	3285
moving	670	2184
other	4495	16690
renewable_energy	77	252
small_business	1679	4022
vacation	464	1988
wedding	219	1593



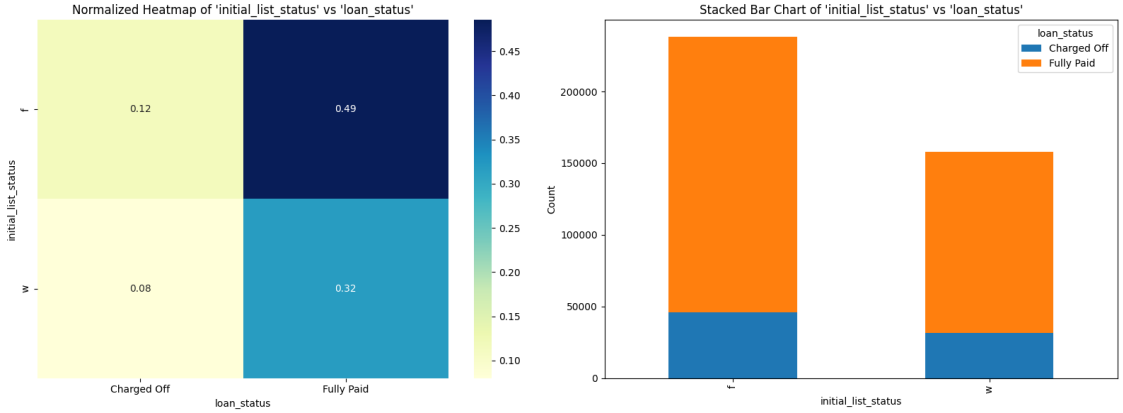

```
============================================================
Skipping charts for 'title' and 'loan_status' because one of them has more than
15 unique values.
============================================================
```

loan_status	Charged Off	Fully Paid
title		
\tcredit_card	0	1
\tdebt_consolidation	0	3
\tother	2	2
\tsmall_business	0	2
debt consolidation	0	1
…	…	…
zipcar	0	1
zonball Loan	1	0
zxcvb	0	1
~Life Reorganization~	1	0
~Summer Fun~	0	1

[48816 rows x 2 columns]

```
============================================================
 Crosstab between 'initial_list_status' and 'loan_status'
============================================================
```

loan_status	Charged Off	Fully Paid
initial_list_status		
f	45961	192105
w	31712	126252

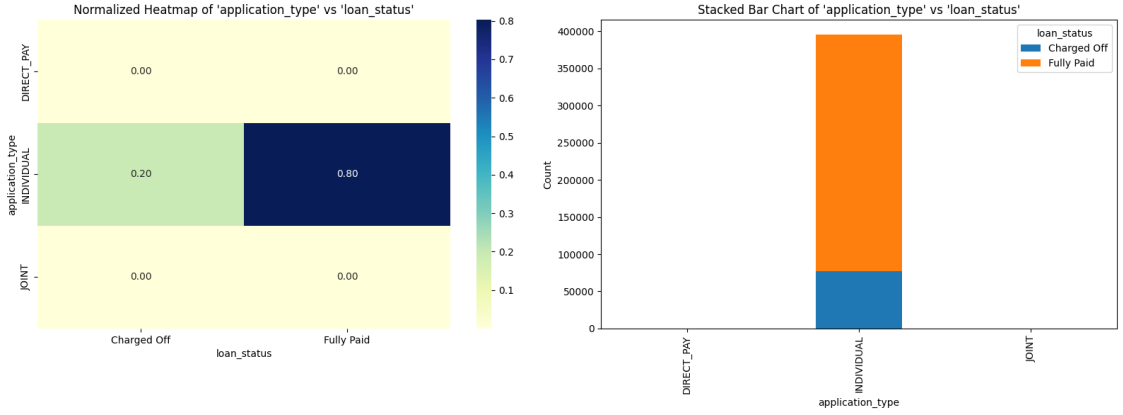


==========

Crosstab between 'application_type' and 'loan_status'

==========

loan_status	Charged Off	Fully Paid
application_type		
DIRECT_PAY	102	184
INDIVIDUAL	77517	317802
JOINT	54	371



==========

Skipping charts for 'city' and 'loan_status' because one of them has more than 15 unique values.

==========

```
loan_status   Charged Off  Fully Paid
city
APO                  2805       11255
Aaronberg               1          15
Aaronborough            6          12
Aaronburgh              4          13
Aaronbury               1          13
…                     …          …
Zunigaport              0           2
Zunigashire             1           3
Zunigaside              1           0
Zunigaton               0           2
Zunigaville             0           2

[67513 rows x 2 columns]


============================================================
Skipping charts for 'state' and 'loan_status' because one of them has more than
15 unique values.
============================================================


loan_status  Charged Off  Fully Paid
state
AA                  2711       11208
AE                  2763       11394
AK                  1390        5644
AL                  1348        5550
AP                  2873       11435
AR                  1381        5588
AZ                  1360        5558
CA                  1314        5584
CO                  1346        5568
CT                  1341        5563
DC                  1338        5504
DE                  1320        5554
FL                  1339        5582
GA                  1390        5577
HI                  1329        5598
IA                  1349        5577
ID                  1389        5569
IL                  1377        5557
IN                  1330        5628
KS                  1386        5559
KY                  1366        5434
LA                  1369        5699
MA                  1377        5645
MD                  1354        5542
```

ME	1355	5617
MI	1332	5522
MN	1249	5655
MO	1371	5568
MS	1414	5589
MT	1390	5493
NC	1370	5531
ND	1330	5528
NE	1381	5546
NH	1319	5499
NJ	1377	5714
NM	1335	5507
NV	1425	5613
NY	1306	5698
OH	1346	5623
OK	1327	5584
OR	1308	5590
PA	1384	5441
RI	1351	5589
SC	1370	5603
SD	1357	5530
TN	1351	5518
TX	1370	5630
UT	1354	5533
VA	1370	5652
VT	1335	5670
WA	1394	5501
WI	1402	5679
WV	1417	5527
WY	1443	5490

Analysis and Insights of 'loan_term' and rest of the categorial column

- **loan term:**
  - Loans with a 60-month term have a significantly higher default rate (31.9%) compared to 36-month term loans (15.8%).
  - 36-month term loans have a higher likelihood of being fully paid (84.2%) compared to 60-month term loans (68.1%).
  - There are significantly more 36-month term loans (301,965) than 60-month term loans (94,025).
- **grade:**
  - Grade A loans have a 93.7% chance of being fully paid, while Grade G loans have only a 52.2% chance of being fully paid. This pattern suggests a clear correlation between lower grades and higher risk of default.
  - Grades B and C have the highest volume of loans, making them popular choices but with a moderate risk of default. Grade G has the lowest volume, likely due to the high risk associated with these loans.
  - High-grade loans (A and B) are safer investments and have a higher likelihood of being

fully repaid. Lower-grade loans (E, F, and G) should be approached with caution, potentially requiring higher interest rates or stricter borrower evaluations to mitigate risk.

- **sub_grade:**
  - As the sub-grade decreases (from A1 to G5), the default rate (charged off rate) increases. For example, A1 has a 2.87% default rate, while G5 has a default rate of 50.3%. This suggests that loans with lower sub-grades are significantly riskier.
  - Higher sub-grades (A1, A2, A3) have a high percentage of fully paid loans, with A1 having about 97.13% of loans fully paid. Lower sub-grades (F and G) show a decline in fully paid rates, with G5 having only 50.3% of loans fully paid.
  - Higher sub-grades (A1, A2, B1, B2, etc.) account for a larger portion of loans and have lower risk. Lower sub-grades (E, F, G) have fewer loans, and the default rates are much higher, indicating these loans are considered high-risk.
- **emp_length:**
  - The default rate (Charged Off Rate) is higher for shorter employment lengths. For example, individuals with less than 1 year of employment have a 19.93% charged off rate, while those with 10+ years of employment have a 18.42% charged off rate. This suggests that shorter employment periods are slightly riskier in terms of loan defaults compared to longer employment histories.
  - Longer employment (such as 10+ years) correlates with higher fully paid rates. For example, individuals with 10+ years of employment have an 81.58% fully paid rate, while those with less than 1 year of employment have 80.07%. This indicates that individuals with longer tenure in their jobs are slightly more likely to fully repay their loans.
  - While there is a slightly higher default rate for individuals with shorter employment lengths, the difference is not substantial. The fully paid rate remains relatively high across all employment categories, suggesting that employment length is a moderate factor in predicting loan repayment.
  - Lenders may want to consider employment history as part of the loan approval process, but other factors like income stability, credit score, and debt-to-income ratio may be more predictive of loan default.
  - Applicants with < 1 year of employment or 1 year of employment may be considered riskier, though not drastically so. Lenders may want to tailor lending rates or use other measures (e.g., higher interest rates, stricter terms) when dealing with such individuals.
- **home_ownership:**
  - The default rate (Charged Off Rate) is highest for individuals with RENT and NONE home ownership status: RENT: 22.65% and NONE: 22.58%. This indicates that individuals who rent or have no home ownership are slightly more likely to default on their loans.
  - The fully paid rate (Fully Paid Rate) is highest for individuals with a home ownership status of ANY, which has a 100% fully paid rate. However, this category contains very few loans (only 3 loans in total). MORTGAGE holders have the second-highest fully paid rate at 83.04%, followed by OTHER at 85.71%. This suggests that homeowners, particularly those with mortgages, tend to repay their loans more reliably compared to renters or individuals with no home ownership.
  - The ANY category has a 0% charged-off rate, but this category is very small (only 3 loans). MORTGAGE holders have a relatively low charged-off rate of 16.96%, which is significantly lower than those with RENT or NONE statuses.

  - A significant proportion of loans are issued to individuals who either rent (159,790 loans) or have a mortgage (198,348 loans). However, individuals with RENT or NONE statuses tend to default more frequently, suggesting that home ownership (especially mortgage holders) may be a predictor of lower risk.
  - Renters and individuals with no home ownership are riskier applicants compared to homeowners (especially those with mortgages). Lenders might consider adjusting loan terms or applying stricter eligibility criteria for renters and individuals without home ownership to mitigate risk.
- **verification_status:**
  - The default rate (Charged Off Rate) is highest for individuals with Verified and Source Verified status: Verified: 22.32% and Source Verified: 21.46%. This indicates that applicants with verification statuses of "Verified" and "Source Verified" have a higher likelihood of defaulting on their loans compared to those who are Not Verified.
  - The fully paid rate (Fully Paid Rate) is highest for individuals with Not Verified status at 85.34%, followed by Source Verified at 78.54% and Verified at 77.68%. Interestingly, individuals who are Not Verified actually have the highest rate of fully paid loans, suggesting that while verification status is important, other factors might be at play in determining loan repayment behavior.
  - A large number of loans are issued to individuals who are either Not Verified (125,082 loans) or Verified (139,563 loans). However, despite a larger number of loans in these categories, the default rates are higher for verified individuals. Not Verified applicants, despite having fewer loans overall, have a higher percentage of loans fully paid off, indicating that loan verification might not always correlate with better repayment performance.
  - Applicants who are Source Verified or Verified appear to have higher chances of default. Lenders might need to assess these applications more carefully and consider other factors, such as credit history or income, as these individuals are more likely to default compared to those who are Not Verified.
  - The Not Verified group has the highest percentage of loans that are fully paid. While the exact reasons behind this trend are unclear, it might suggest that other characteristics (e.g., income level, credit score) of Not Verified individuals are stronger predictors of loan repayment than the verification status itself.
- **purpose:**
  - The small business purpose has the highest default rate at 29.43%, indicating that individuals seeking loans for small businesses have a significantly higher likelihood of defaulting. Renewable energy loans also have a high default rate at 23.41%, which may suggest that loans for energy projects or green initiatives are riskier.
  - Loans for weddings have the lowest default rate at 12.09%, followed by car loans at 13.47%, which indicates that these types of loans are more likely to be paid off in full.
  - Loans for weddings and cars have the highest fully paid rates at 86.53% and 87.91%, respectively. On the other hand, small business loans have the lowest fully paid rate at 70.57%, which is consistent with the high default rate for this category.
  - Debt consolidation loans have a high number of fully paid loans (79.26%) and a significant number of charged off loans. This indicates that while debt consolidation may help borrowers manage their debts, it does not necessarily guarantee repayment. Credit card loans also have a relatively high fully paid rate (83.09%) but a noticeable proportion of defaults (16.91%), suggesting credit card-related loans carry moderate risk.
- **initial_list_status:**

  - The charged off rate for w (20.06%) is slightly higher than that for f (19.33%), indicating that loans under the w status are marginally more likely to default.
  - The fully paid rate for f (80.67%) is slightly higher than that for w (79.94%), indicating that loans listed with f status have a slightly better repayment record.
  - There are more loans in the f status than in the w status. Specifically, the total number of loans with f status is 238,066, while loans with w status are 157,964. This suggests that f status loans are more common than w status loans, yet both types have relatively similar repayment behavior.
- **application_type:**
  - The charged off rate is highest for DIRECT_PAY loans (35.7%), followed by INDIVIDUAL loans (19.6%), and the lowest for JOINT loans (12.7%). This indicates that loans where the application type is DIRECT_PAY have a significantly higher default rate compared to INDIVIDUAL and JOINT loans.
  - The fully paid rate is highest for JOINT loans (87.4%), followed by INDIVIDUAL loans (80.2%), and the lowest for DIRECT_PAY loans (64.3%). This suggests that JOINT loans tend to have better repayment outcomes compared to other types of loans.
  - The majority of the loans are of INDIVIDUAL type, with a total of 395,319 loans, accounting for a significant portion of the dataset. JOINT loans are relatively rare, with only 425 loans, while DIRECT_PAY loans are also quite low in number (286).
- **state:**
  - The Charged Off Rate is consistent across most states, hovering around 19.5% to 19.8%, with only slight variations. Some states like AK and TX have a slightly higher charged-off rate, but overall, this trend seems uniform across the dataset.
  - The Fully Paid Rate is consistently high across all states, generally around 80%. This indicates that most loans, irrespective of the state, are repaid on time, with only a small portion being charged off.
  - States like AA and AE have higher loan volumes, suggesting a greater presence of loans from these regions in the dataset. Smaller states, like AK, still show a reasonable number of loans but represent a smaller proportion of the total dataset.

[21]:
```
df.loan_status
```

[21]:
```
0          Fully Paid
1          Fully Paid
2          Fully Paid
3          Fully Paid
4         Charged Off
              …
396025     Fully Paid
396026     Fully Paid
396027     Fully Paid
396028     Fully Paid
396029     Fully Paid
Name: loan_status, Length: 396030, dtype: object
```

[15]:
```
loan_status_map = {
    'Fully Paid': 1, 'Charged Off': 0
}
```

[16]:
```
df['loan_status_num'] = df['loan_status'].map(loan_status_map)
```

[17]:
```
df.head()
```

[17]:

	loan_amnt	term	int_rate	installment	grade	sub_grade	\
0	10000.0	36 months	11.44	329.48	B	B4	
1	8000.0	36 months	11.99	265.68	B	B5	
2	15600.0	36 months	10.49	506.97	B	B3	
3	7200.0	36 months	6.49	220.65	A	A2	
4	24375.0	60 months	17.27	609.33	C	C5	

	emp_title	emp_length	home_ownership	annual_inc	…	\
0	Marketing	10+ years	RENT	117000.0	…	
1	Credit analyst	4 years	MORTGAGE	65000.0	…	
2	Statistician	< 1 year	RENT	43057.0	…	
3	Client Advocate	6 years	RENT	54000.0	…	
4	Destiny Management Inc.	9 years	MORTGAGE	55000.0	…	

	application_type	mort_acc	pub_rec_bankruptcies	city	state	\
0	INDIVIDUAL	0.0	0.0	Mendozaberg	OK	
1	INDIVIDUAL	3.0	0.0	Loganmouth	SD	
2	INDIVIDUAL	0.0	0.0	New Sabrina	WV	
3	INDIVIDUAL	0.0	0.0	Delacruzside	MA	
4	INDIVIDUAL	1.0	0.0	Greggshire	VA	

	earliest_cr_line_month	earliest_cr_line_year	issue_month	issue_year	\
0	6	1990	1	2015	
1	7	2004	1	2015	
2	8	2007	1	2015	
3	9	2006	11	2014	
4	3	1999	4	2013	

	loan_status_num
0	1
1	1
2	1
3	1
4	0

[5 rows x 31 columns]

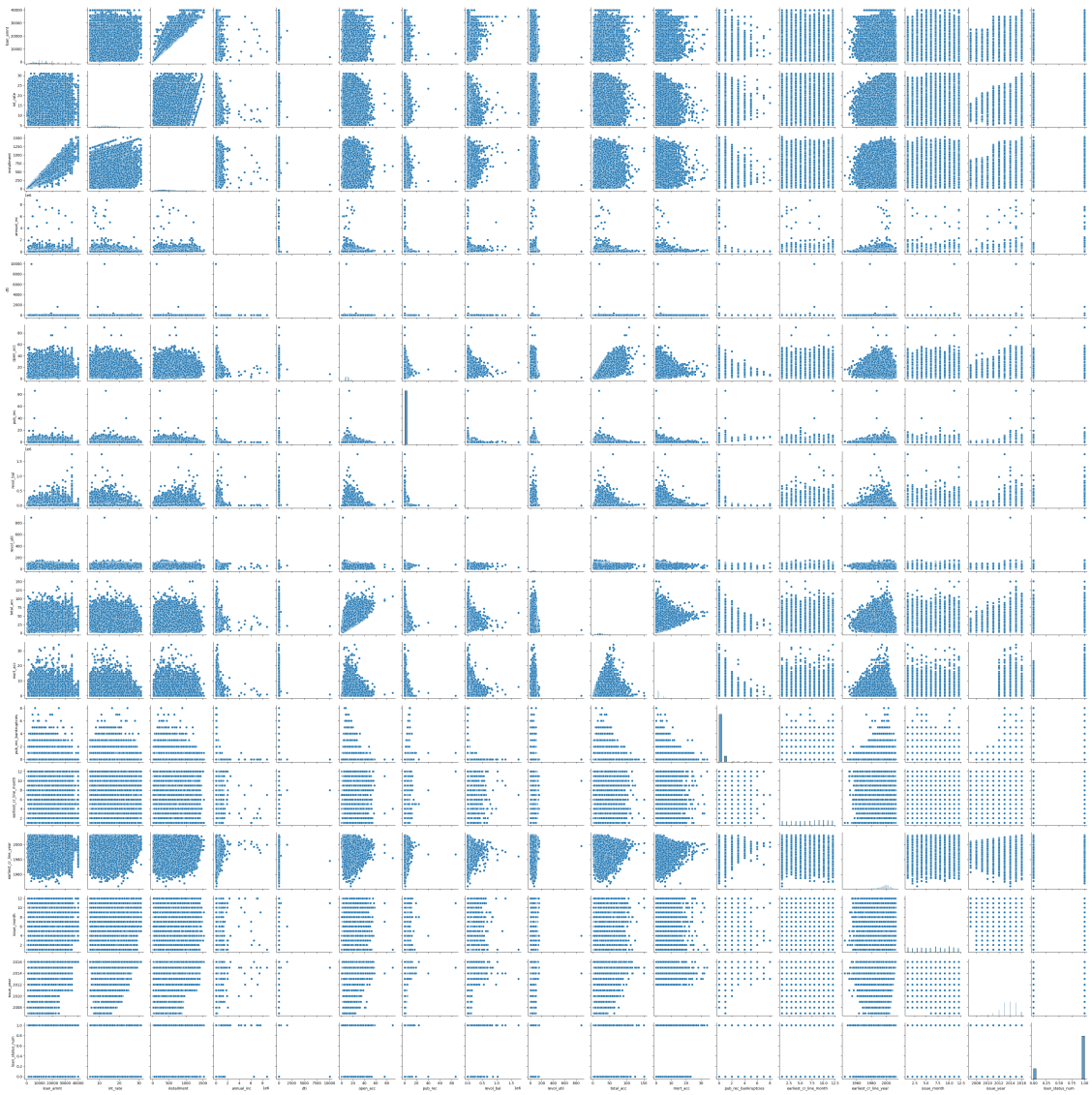
[25]:
```
df.select_dtypes(include=['number']).columns.tolist()
```

[25]:
```
['loan_amnt',
 'int_rate',
 'installment',
 'annual_inc',
```

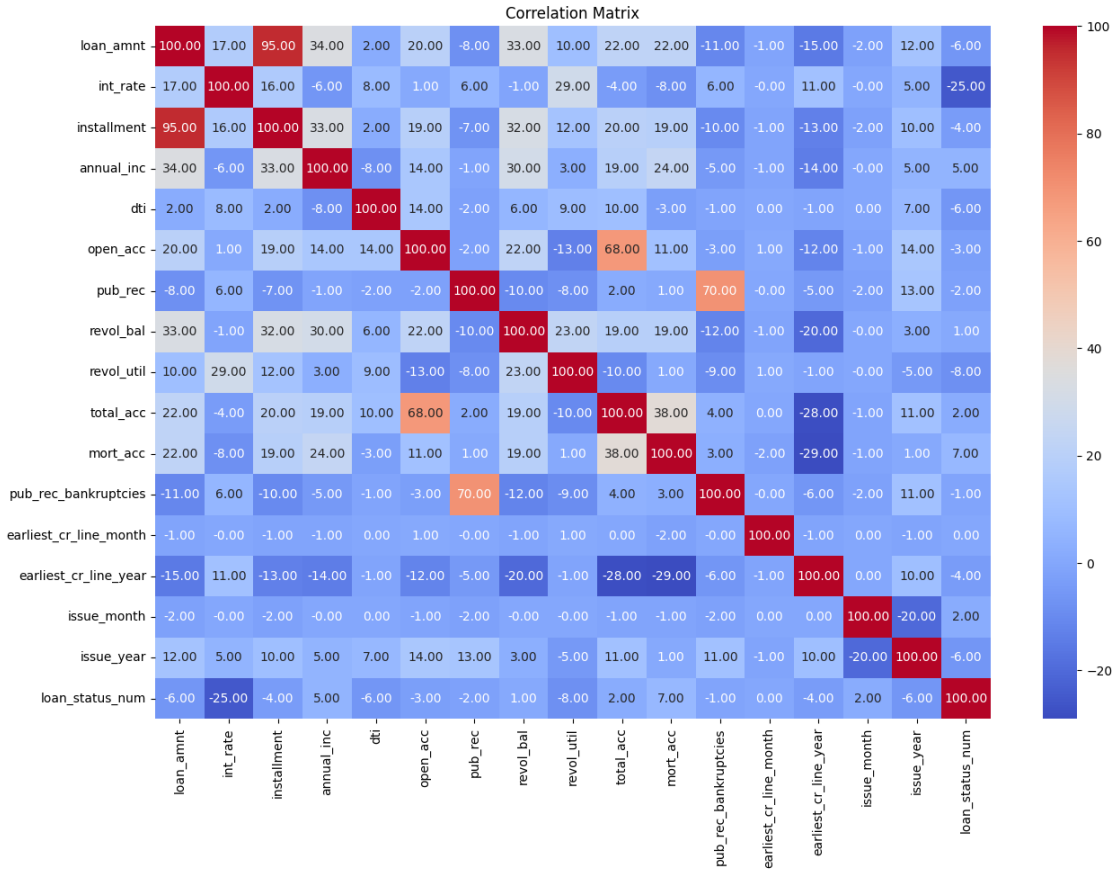
```
 'dti',
 'open_acc',
 'pub_rec',
 'revol_bal',
 'revol_util',
 'total_acc',
 'mort_acc',
 'pub_rec_bankruptcies',
 'earliest_cr_line_month',
 'earliest_cr_line_year',
 'issue_month',
 'issue_year',
 'loan_status_num']
```

[18]:
```
sns.pairplot(df[df.select_dtypes(include=['number']).columns.tolist()])
#plt.savefig('scatter_plot.png', dpi=300, bbox_inches='tight');
```

[18]: <seaborn.axisgrid.PairGrid at 0x23551d9f770>



[27]:
```
plt.figure(figsize=(15, 10))
sns.heatmap((df[df.select_dtypes(include=['number']).columns.tolist()].
  ↪corr()*100).round(0), annot=True, cmap='coolwarm', fmt=".2f")
plt.title("Correlation Matrix");
```



[28]:
```
(df[df.select_dtypes(include=['number']).columns.tolist()].corr()*100).round(2)
```

[28]:

	loan_amnt	int_rate	installment	annual_inc	dti	\
loan_amnt	100.00	16.89	95.39	33.69	1.66	
int_rate	16.89	100.00	16.28	-5.68	7.90	
installment	95.39	16.28	100.00	33.04	1.58	
annual_inc	33.69	-5.68	33.04	100.00	-8.17	
dti	1.66	7.90	1.58	-8.17	100.00	
open_acc	19.86	1.16	18.90	13.62	13.62	
pub_rec	-7.78	6.10	-6.79	-1.37	-1.76	
revol_bal	32.83	-1.13	31.65	29.98	6.36	
revol_util	9.99	29.37	12.39	2.79	8.84	
total_acc	22.39	-3.64	20.24	19.30	10.21	
mort_acc	22.23	-8.26	19.37	23.63	-2.54	
pub_rec_bankruptcies	-10.65	5.75	-9.86	-5.02	-1.46	
earliest_cr_line_month	-1.06	-0.27	-1.08	-1.47	0.25	
earliest_cr_line_year	-14.70	10.85	-13.18	-14.05	-0.95	
issue_month	-2.03	-0.21	-2.33	-0.49	0.01	
issue_year	11.51	5.04	10.35	5.00	7.48	

```
loan_status_num             -5.98    -24.78        -4.11         5.34   -6.24

                         open_acc  pub_rec  revol_bal  revol_util  total_acc  \
loan_amnt                   19.86    -7.78      32.83        9.99      22.39
int_rate                     1.16     6.10      -1.13       29.37      -3.64
installment                 18.90    -6.79      31.65       12.39      20.24
annual_inc                  13.62    -1.37      29.98        2.79      19.30
dti                         13.62    -1.76       6.36        8.84      10.21
open_acc                   100.00    -1.84      22.12      -13.14      68.07
pub_rec                     -1.84   100.00     -10.17       -7.59       1.97
revol_bal                   22.12   -10.17     100.00       22.63      19.16
revol_util                 -13.14    -7.59      22.63      100.00     -10.43
total_acc                   68.07     1.97      19.16      -10.43     100.00
mort_acc                    10.92     1.16      19.49        0.75      38.11
pub_rec_bankruptcies        -2.77    69.94     -12.45       -8.68       4.20
earliest_cr_line_month       0.55    -0.40      -1.22        0.51       0.38
earliest_cr_line_year      -12.35    -5.33     -19.92       -0.75     -27.81
issue_month                 -0.91    -2.35      -0.25       -0.42      -0.79
issue_year                  13.76    12.85       3.44       -5.31      11.11
loan_status_num             -2.80    -1.99       1.09       -8.24       1.79

                         mort_acc  pub_rec_bankruptcies  \
loan_amnt                   22.23                -10.65
int_rate                    -8.26                  5.75
installment                 19.37                 -9.86
annual_inc                  23.63                 -5.02
dti                         -2.54                 -1.46
open_acc                    10.92                 -2.77
pub_rec                      1.16                 69.94
revol_bal                   19.49                -12.45
revol_util                   0.75                 -8.68
total_acc                   38.11                  4.20
mort_acc                   100.00                  2.72
pub_rec_bankruptcies         2.72                100.00
earliest_cr_line_month      -1.95                 -0.45
earliest_cr_line_year      -29.13                 -5.65
issue_month                 -0.67                 -1.73
issue_year                   1.12                 11.04
loan_status_num              7.31                 -0.94

                         earliest_cr_line_month  earliest_cr_line_year  \
loan_amnt                                 -1.06                 -14.70
int_rate                                  -0.27                  10.85
installment                               -1.08                 -13.18
annual_inc                                -1.47                 -14.05
dti                                        0.25                  -0.95
open_acc                                   0.55                 -12.35
```

pub_rec	-0.40	-5.33
revol_bal	-1.22	-19.92
revol_util	0.51	-0.75
total_acc	0.38	-27.81
mort_acc	-1.95	-29.13
pub_rec_bankruptcies	-0.45	-5.65
earliest_cr_line_month	100.00	-1.29
earliest_cr_line_year	-1.29	100.00
issue_month	0.19	0.05
issue_year	-0.80	9.86
loan_status_num	0.39	-3.89

	issue_month	issue_year	loan_status_num
loan_amnt	-2.03	11.51	-5.98
int_rate	-0.21	5.04	-24.78
installment	-2.33	10.35	-4.11
annual_inc	-0.49	5.00	5.34
dti	0.01	7.48	-6.24
open_acc	-0.91	13.76	-2.80
pub_rec	-2.35	12.85	-1.99
revol_bal	-0.25	3.44	1.09
revol_util	-0.42	-5.31	-8.24
total_acc	-0.79	11.11	1.79
mort_acc	-0.67	1.12	7.31
pub_rec_bankruptcies	-1.73	11.04	-0.94
earliest_cr_line_month	0.19	-0.80	0.39
earliest_cr_line_year	0.05	9.86	-3.89
issue_month	100.00	-19.83	1.64
issue_year	-19.83	100.00	-6.05
loan_status_num	1.64	-6.05	100.00

Analysis and Insights of 'loan_term_num' and rest of the numerical columns

- High Positive Correlations
  - loan_amnt and installment (95.39%): Loan amount and installment are strongly correlated, as the installment amount directly depends on the loan amount and interest rate.
  - loan_amnt and annual_inc (33.69%): Borrowers with higher annual incomes tend to take larger loan amounts.
  - total_acc and open_acc (68.07%): The total number of accounts and open accounts are closely related, as open accounts contribute to the total account count.
  - pub_rec and pub_rec_bankruptcies (69.94%): Public records and bankruptcies are highly correlated, which makes sense as bankruptcies are part of public records.
  - mort_acc and total_acc (38.11%): Borrowers with higher total accounts often have more mortgage accounts, reflecting greater financial activity.
- Moderate Positive Correlations
  - int_rate and revol_util (29.37%): Borrowers with higher revolving credit utilization may face higher interest rates, possibly due to increased credit risk.

  - revol_bal and annual_inc (29.98%): Higher annual incomes are moderately correlated with higher revolving balances, possibly indicating higher credit usage.
  - issue_year and loan_amnt (11.51%): Recent loan issues tend to have higher amounts, potentially reflecting inflation or changes in lending practices.
- High Negative Correlations
  - earliest_cr_line_year and total_acc (-27.81%): Borrowers with older credit lines tend to have fewer total accounts, likely reflecting a more conservative credit approach.
  - loan_status_num and int_rate (-24.78%): A negative correlation suggests that higher interest rates are associated with poorer loan outcomes (loan_status_num may represent default likelihood).
  - earliest_cr_line_year and mort_acc (-29.13%): Borrowers with older credit lines tend to have fewer mortgage accounts, possibly due to earlier financial habits or different financial goals.
- Weak or Insignificant Correlations
  - Variables like issue_month and loan_amnt, dti, or open_acc show correlations close to 0, indicating no meaningful relationship.
  - earliest_cr_line_month has consistently low correlations with most variables, suggesting it has little predictive power in this context.
- Key Insights
  - Loan Performance Indicators: High interest rates and high int_rate correlate negatively with loan_status_num, suggesting they might be risk factors for defaults.
  - Credit Utilization: Higher revol_util (credit utilization) correlates positively with int_rate, potentially signaling risk assessment by lenders.
  - Financial Activity: Strong positive correlations among total_acc, open_acc, and mort_acc reflect that borrowers with more financial accounts are likely to have diverse credit profiles.
  - Temporal Trends: Negative correlation between earliest_cr_line_year and loan_amnt suggests that newer borrowers tend to take higher loan amounts.
  - Public Records: Strong correlation between pub_rec and pub_rec_bankruptcies underscores the importance of public records in credit analysis.

3 Data Preprocessing

[29]:
```
df.head(5)
```

[29]:

	loan_amnt	term	int_rate	installment	grade	sub_grade	\
0	10000.0	36 months	11.44	329.48	B	B4	
1	8000.0	36 months	11.99	265.68	B	B5	
2	15600.0	36 months	10.49	506.97	B	B3	
3	7200.0	36 months	6.49	220.65	A	A2	
4	24375.0	60 months	17.27	609.33	C	C5	

	emp_title	emp_length	home_ownership	annual_inc	…	\
0	Marketing	10+ years	RENT	117000.0	…	
1	Credit analyst	4 years	MORTGAGE	65000.0	…	
2	Statistician	< 1 year	RENT	43057.0	…	
3	Client Advocate	6 years	RENT	54000.0	…	

```
4  Destiny Management Inc.    9 years       MORTGAGE     55000.0  …

  application_type mort_acc pub_rec_bankruptcies          city  state  \
0       INDIVIDUAL      0.0                  0.0   Mendozaberg     OK
1       INDIVIDUAL      3.0                  0.0    Loganmouth     SD
2       INDIVIDUAL      0.0                  0.0   New Sabrina     WV
3       INDIVIDUAL      0.0                  0.0  Delacruzside     MA
4       INDIVIDUAL      1.0                  0.0    Greggshire     VA

   earliest_cr_line_month  earliest_cr_line_year  issue_month  issue_year  \
0                       6                   1990            1        2015
1                       7                   2004            1        2015
2                       8                   2007            1        2015
3                       9                   2006           11        2014
4                       3                   1999            4        2013

   loan_status_num
0                1
1                1
2                1
3                1
4                0

[5 rows x 31 columns]
```

3.1 Duplicate rows

[30]:
```
print(f'Count of duplicated rows is {df[df.duplicated()].shape[0]}')
```

```
Count of duplicated rows is 0
```

3.2 Null Values and Treatment

[31]:
```
print('Presence of null values in the dataset')
pd.DataFrame({
    'Null Count': df.isnull().sum(),
    'Null Percentage': round((df.isnull().sum() / df.shape[0]) * 100, 2)
})
```

```
Presence of null values in the dataset
```

[31]:

	Null Count	Null Percentage
loan_amnt	0	0.00
term	0	0.00
int_rate	0	0.00
installment	0	0.00
grade	0	0.00
sub_grade	0	0.00

```
emp_title                22927            5.79
emp_length               18301            4.62
home_ownership               0            0.00
annual_inc                   0            0.00
verification_status          0            0.00
loan_status                  0            0.00
purpose                      0            0.00
title                     1756            0.44
dti                          0            0.00
open_acc                     0            0.00
pub_rec                      0            0.00
revol_bal                    0            0.00
revol_util                 276            0.07
total_acc                    0            0.00
initial_list_status          0            0.00
application_type             0            0.00
mort_acc                 37795            9.54
pub_rec_bankruptcies       535            0.14
city                         0            0.00
state                        0            0.00
earliest_cr_line_month       0            0.00
earliest_cr_line_year        0            0.00
issue_month                  0            0.00
issue_year                   0            0.00
loan_status_num              0            0.00
```

[32]:
```
df['emp_title'] = df['emp_title'].fillna(df['emp_title'].mode()[0])
```

[33]:
```
df['emp_length'] = df['emp_length'].fillna(df['emp_length'].mode()[0])
df['title'] = df['title'].fillna(df['title'].mode()[0])
```

[34]:
```
from sklearn.impute import SimpleImputer
```

[35]:
```
imputer = SimpleImputer(strategy='median')
```

[36]:
```
df['revol_util'] = imputer.fit_transform(df[['revol_util']])
```

[37]:
```
df['mort_acc'] = imputer.fit_transform(df[['mort_acc']])
df['pub_rec_bankruptcies'] = imputer.fit_transform(df[['pub_rec_bankruptcies']])
```

[38]:
```
df.isnull().sum().sum()
```

[38]:
```
np.int64(0)
```

3.3 Outlier Treatment

[39]:
```
from scipy.stats import zscore
```

[40]:
```
numerical_cols = ['loan_amnt', 'int_rate', 'installment', 'annual_inc', 'dti',␣
 ↪'open_acc', 'revol_bal', 'revol_util', 'total_acc', 'mort_acc']
z_scores = np.abs(zscore(df[numerical_cols]))
threshold = 3
df[numerical_cols] = np.where(z_scores > threshold, np.sign(df[numerical_cols])␣
 ↪* threshold, df[numerical_cols])
```

[41]:
```
df.head()
```

[41]:

	loan_amnt	term	int_rate	installment	grade	sub_grade	\
0	10000.0	36 months	11.44	329.48	B	B4	
1	8000.0	36 months	11.99	265.68	B	B5	
2	15600.0	36 months	10.49	506.97	B	B3	
3	7200.0	36 months	6.49	220.65	A	A2	
4	24375.0	60 months	17.27	609.33	C	C5	

	emp_title	emp_length	home_ownership	annual_inc	…	\
0	Marketing	10+ years	RENT	117000.0	…	
1	Credit analyst	4 years	MORTGAGE	65000.0	…	
2	Statistician	< 1 year	RENT	43057.0	…	
3	Client Advocate	6 years	RENT	54000.0	…	
4	Destiny Management Inc.	9 years	MORTGAGE	55000.0	…	

	application_type	mort_acc	pub_rec_bankruptcies	city	state	\
0	INDIVIDUAL	0.0	0.0	Mendozaberg	OK	
1	INDIVIDUAL	3.0	0.0	Loganmouth	SD	
2	INDIVIDUAL	0.0	0.0	New Sabrina	WV	
3	INDIVIDUAL	0.0	0.0	Delacruzside	MA	
4	INDIVIDUAL	1.0	0.0	Greggshire	VA	

	earliest_cr_line_month	earliest_cr_line_year	issue_month	issue_year	\
0	6	1990	1	2015	
1	7	2004	1	2015	
2	8	2007	1	2015	
3	9	2006	11	2014	
4	3	1999	4	2013	

	loan_status_num
0	1
1	1
2	1
3	1
4	0

```
[5 rows x 31 columns]
```

3.4 Feature Engineering

[42]:
```
df_main=df.copy()
```

[43]:
```
df_object=df[df.select_dtypes(include=['object']).columns.tolist()]
df_object.head()
```

[43]:

	term	grade	sub_grade	emp_title	emp_length	home_ownership	verification_status	loan_status	purpose	title	initial_list_status	application_type	city	state
0	36 months	B	B4	Marketing	10+ years	RENT	Not Verified	Fully Paid	vacation	Vacation	w	INDIVIDUAL	Mendozaberg	OK
1	36 months	B	B5	Credit analyst	4 years	MORTGAGE	Not Verified	Fully Paid	debt_consolidation	Debt consolidation	f	INDIVIDUAL	Loganmouth	SD
2	36 months	B	B3	Statistician	< 1 year	RENT	Source Verified	Fully Paid	credit_card	Credit card refinancing	f	INDIVIDUAL	New Sabrina	WV
3	36 months	A	A2	Client Advocate	6 years	RENT	Not Verified	Fully Paid	credit_card	Credit card refinancing	f	INDIVIDUAL	Delacruzside	MA
4	60 months	C	C5	Destiny Management Inc.	9 years	MORTGAGE	Verified	Charged Off	credit_card	Credit Card Refinance	f	INDIVIDUAL	Greggshire	VA

[44]:
```
df['term']=df['term'].str.split().str[0]
```

[45]:
```
df['pub_rec'] = np.where(df['pub_rec'] > 0, 1, 0)
```

[46]:
```
df['mort_acc'] = np.where(df['mort_acc'] > 0, 1, 0)
df['pub_rec_bankruptcies'] = np.where(df['pub_rec_bankruptcies'] > 0, 1, 0)
```

[47]:
```
emp_length_mapping = {'< 1 year': 0, '1 year': 1, '2 years': 2, '3 years': 3,␣
 ↪'4 years': 4, '5 years': 5, '6 years': 6, '7 years': 7, '8 years': 8, '9␣
 ↪years': 9, '10+ years': 10}
```

```
df['emp_length'] = df['emp_length'].map(emp_length_mapping)
```

[48]:
```
df.drop(columns=['loan_status'], axis=1, inplace=True)
```

[49]:
```
categorical_columns = df.select_dtypes(include=['object']).columns.tolist()
global_mean = df['loan_status_num'].mean()
def target_encode_binary(df, categorical_columns, target_column,␣
 ↪smoothing_factor=10):
    for col in categorical_columns:
        mean_per_category = df.groupby(col)[target_column].mean()
        count_per_category = df.groupby(col)[target_column].count()
        smooth_mean = (mean_per_category * count_per_category + global_mean *␣
 ↪smoothing_factor) / (count_per_category + smoothing_factor)
        df[col] = df[col].map(smooth_mean)
    return df
df = target_encode_binary(df, categorical_columns,␣
 ↪target_column='loan_status_num', smoothing_factor=10)
```

[50]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 396030 entries, 0 to 396029
Data columns (total 30 columns):
 #   Column                  Non-Null Count   Dtype
---  ------                  --------------   -----
 0   loan_amnt               396030 non-null  float64
 1   term                    396030 non-null  float64
 2   int_rate                396030 non-null  float64
 3   installment             396030 non-null  float64
 4   grade                   396030 non-null  float64
 5   sub_grade               396030 non-null  float64
 6   emp_title               396030 non-null  float64
 7   emp_length              396030 non-null  int64
 8   home_ownership          396030 non-null  float64
 9   annual_inc              396030 non-null  float64
 10  verification_status     396030 non-null  float64
 11  purpose                 396030 non-null  float64
 12  title                   396030 non-null  float64
 13  dti                     396030 non-null  float64
 14  open_acc                396030 non-null  float64
 15  pub_rec                 396030 non-null  int64
 16  revol_bal               396030 non-null  float64
 17  revol_util              396030 non-null  float64
 18  total_acc               396030 non-null  float64
 19  initial_list_status     396030 non-null  float64
 20  application_type        396030 non-null  float64
 21  mort_acc                396030 non-null  int64
 22  pub_rec_bankruptcies    396030 non-null  int64
```

```
 23  city                    396030 non-null  float64
 24  state                   396030 non-null  float64
 25  earliest_cr_line_month  396030 non-null  int64
 26  earliest_cr_line_year   396030 non-null  int64
 27  issue_month             396030 non-null  int64
 28  issue_year              396030 non-null  int64
 29  loan_status_num         396030 non-null  int64
dtypes: float64(21), int64(9)
memory usage: 90.6 MB
```

[51]:
```
df.head()
```

[51]:

	loan_amnt	term	int_rate	installment	grade	sub_grade	emp_title	\
0	10000.0	0.842253	11.44	329.48	0.874263	0.861584	0.757967	
1	8000.0	0.842253	11.99	265.68	0.874263	0.844944	0.772208	
2	15600.0	0.842253	10.49	506.97	0.874263	0.876619	0.811367	
3	7200.0	0.842253	6.49	220.65	0.937100	0.951659	0.821701	
4	24375.0	0.680598	17.27	609.33	0.788192	0.754960	0.730792	

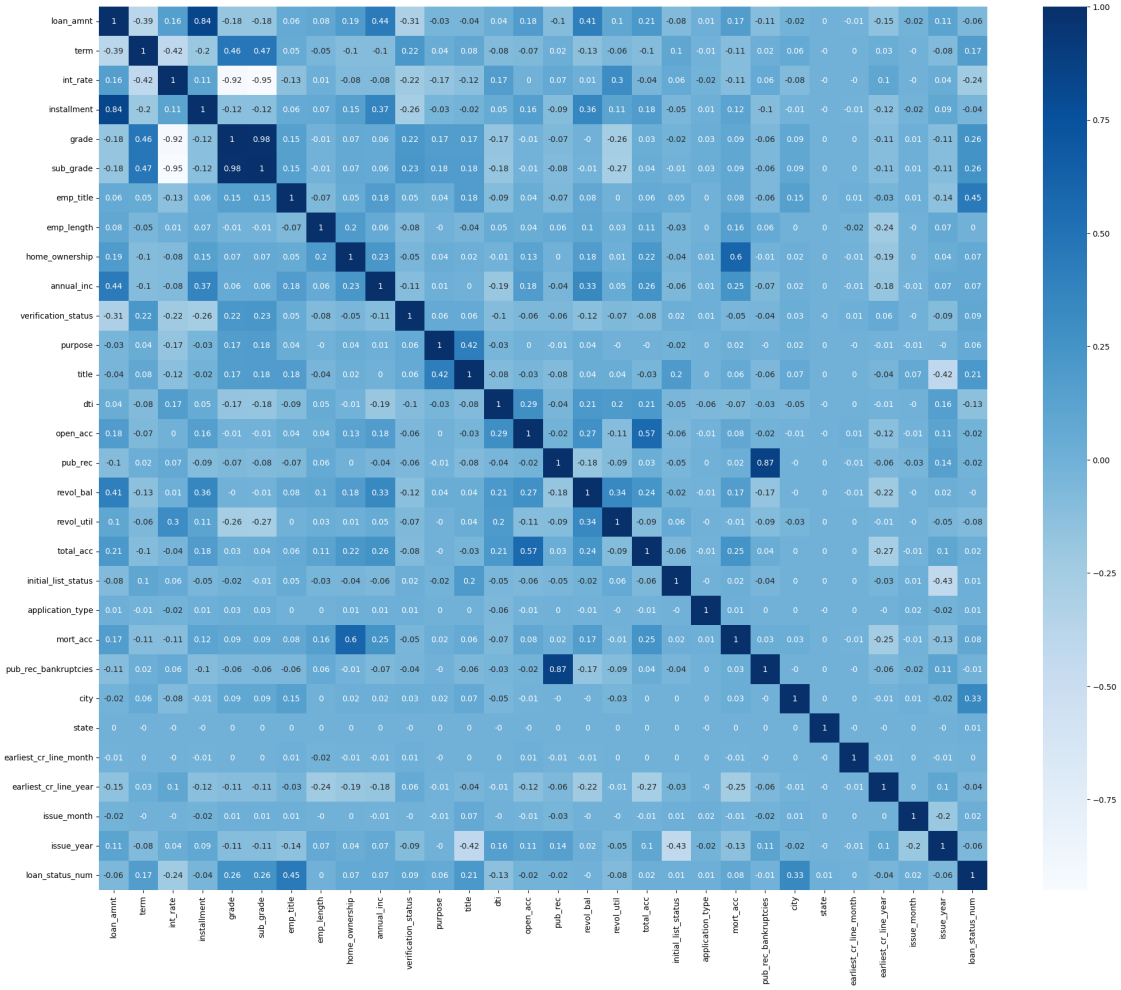
	emp_length	home_ownership	annual_inc	…	application_type	mort_acc	\
0	10	0.773379	117000.0	…	0.803913	0	
1	4	0.830438	65000.0	…	0.803913	1	
2	0	0.773379	43057.0	…	0.803913	0	
3	6	0.773379	54000.0	…	0.803913	0	
4	9	0.830438	55000.0	…	0.803913	1	

	pub_rec_bankruptcies	city	state	earliest_cr_line_month	\
0	0	0.819941	0.807981	6	
1	0	0.709946	0.802963	7	
2	0	0.751210	0.795950	8	
3	0	0.849131	0.803902	9	
4	0	0.730792	0.804897	3	

	earliest_cr_line_year	issue_month	issue_year	loan_status_num
0	1990	1	2015	1
1	2004	1	2015	1
2	2007	1	2015	1
3	2006	11	2014	1
4	1999	4	2013	0

[5 rows x 30 columns]

[52]:
```
plt.figure(figsize=(25,20))
sns.heatmap(round(df.corr(), 2), cmap='Blues', annot=True);
```



[53]:
```
df.corr()
```

[53]:

	loan_amnt	term	int_rate	installment	grade	\
loan_amnt	1.000000	-0.393600	0.162918	0.841649	-0.176716	
term	-0.393600	1.000000	-0.422110	-0.198785	0.458392	
int_rate	0.162918	-0.422110	1.000000	0.113799	-0.924424	
installment	0.841649	-0.198785	0.113799	1.000000	-0.117899	
grade	-0.176716	0.458392	-0.924424	-0.117899	1.000000	
sub_grade	-0.182553	0.468767	-0.945364	-0.121844	0.977568	
emp_title	0.059294	0.051119	-0.134644	0.055127	0.149467	
emp_length	0.077083	-0.046748	0.012963	0.066312	-0.007266	
home_ownership	0.187212	-0.099212	-0.077480	0.146711	0.071705	
annual_inc	0.439489	-0.100111	-0.076874	0.368396	0.061947	
verification_status	-0.308471	0.217986	-0.218256	-0.258876	0.218657	
purpose	-0.031192	0.039813	-0.166610	-0.030373	0.171352	
title	-0.036534	0.078453	-0.119098	-0.022163	0.173095	

dti	0.040845	-0.082572	0.171039	0.045286	-0.172829
open_acc	0.183849	-0.071466	0.002258	0.163995	-0.006268
pub_rec	-0.101545	0.019770	0.066046	-0.088080	-0.074394
revol_bal	0.414579	-0.125678	0.008174	0.356466	-0.003962
revol_util	0.101800	-0.055272	0.296313	0.107791	-0.255630
total_acc	0.213582	-0.095036	-0.042717	0.177114	0.034372
initial_list_status	-0.076008	0.104799	0.058435	-0.048832	-0.016211
application_type	0.013307	-0.005248	-0.019175	0.006395	0.026485
mort_acc	0.171056	-0.105871	-0.111227	0.123952	0.087551
pub_rec_bankruptcies	-0.110183	0.020231	0.055871	-0.095260	-0.058660
city	-0.019153	0.056217	-0.081581	-0.012297	0.085361
state	0.000284	-0.001160	-0.001257	-0.000895	0.002602
earliest_cr_line_month	-0.010600	0.001998	-0.002108	-0.009231	0.001591
earliest_cr_line_year	-0.146889	0.029011	0.103764	-0.121672	-0.110854
issue_month	-0.019988	-0.002117	-0.002496	-0.021654	0.007271
issue_year	0.111667	-0.077146	0.036187	0.090908	-0.107772
loan_status_num	-0.060762	0.173246	-0.243047	-0.038797	0.257886

	sub_grade	emp_title	emp_length	home_ownership	\
loan_amnt	-0.182553	0.059294	0.077083	0.187212	
term	0.468767	0.051119	-0.046748	-0.099212	
int_rate	-0.945364	-0.134644	0.012963	-0.077480	
installment	-0.121844	0.055127	0.066312	0.146711	
grade	0.977568	0.149467	-0.007266	0.071705	
sub_grade	1.000000	0.153416	-0.007613	0.074535	
emp_title	0.153416	1.000000	-0.068495	0.049506	
emp_length	-0.007613	-0.068495	1.000000	0.195498	
home_ownership	0.074535	0.049506	0.195498	1.000000	
annual_inc	0.063959	0.176939	0.064149	0.231329	
verification_status	0.228242	0.046201	-0.076034	-0.046118	
purpose	0.176111	0.039919	-0.002675	0.043425	
title	0.178278	0.184197	-0.044923	0.020356	
dti	-0.178121	-0.094062	0.048009	-0.005597	
open_acc	-0.006289	0.035853	0.035129	0.134977	
pub_rec	-0.077338	-0.073208	0.061374	0.000051	
revol_bal	-0.005697	0.083616	0.099668	0.184515	
revol_util	-0.265806	0.001346	0.029644	0.006470	
total_acc	0.035425	0.056018	0.106461	0.220476	
initial_list_status	-0.011634	0.049559	-0.030670	-0.038779	
application_type	0.026912	0.004903	0.003326	0.012274	
mort_acc	0.090026	0.080234	0.159385	0.599903	
pub_rec_bankruptcies	-0.060984	-0.064795	0.055348	-0.005551	
city	0.087603	0.150176	0.002534	0.021754	
state	0.002793	0.002707	0.000844	0.002241	
earliest_cr_line_month	0.001625	0.007965	-0.023425	-0.012427	
earliest_cr_line_year	-0.114307	-0.032221	-0.235148	-0.187590	
issue_month	0.008742	0.006752	-0.000326	0.002788	

```
issue_year              -0.108578  -0.136912    0.069729        0.038313
loan_status_num          0.263799   0.454040    0.002727        0.068535

                        annual_inc  …  application_type  mort_acc  \
loan_amnt                 0.439489  …          0.013307  0.171056
term                     -0.100111  …         -0.005248 -0.105871
int_rate                 -0.076874  …         -0.019175 -0.111227
installment               0.368396  …          0.006395  0.123952
grade                     0.061947  …          0.026485  0.087551
sub_grade                 0.063959  …          0.026912  0.090026
emp_title                 0.176939  …          0.004903  0.080234
emp_length                0.064149  …          0.003326  0.159385
home_ownership            0.231329  …          0.012274  0.599903
annual_inc                1.000000  …          0.005691  0.248547
verification_status      -0.114561  …          0.007453 -0.050177
purpose                   0.008931  …          0.001435  0.021301
title                     0.004860  …          0.004198  0.063432
dti                      -0.185554  …         -0.059241 -0.071946
open_acc                  0.180031  …         -0.012127  0.083760
pub_rec                  -0.044686  …          0.003677  0.023705
revol_bal                 0.329402  …         -0.007910  0.168359
revol_util                0.047528  …         -0.000593 -0.007620
total_acc                 0.258789  …         -0.009348  0.254351
initial_list_status      -0.059694  …         -0.004581  0.021024
application_type          0.005691  …          1.000000  0.011756
mort_acc                  0.248547  …          0.011756  1.000000
pub_rec_bankruptcies     -0.065831  …          0.003433  0.031784
city                      0.024995  …          0.003998  0.026102
state                     0.003331  …         -0.000819  0.000334
earliest_cr_line_month   -0.013022  …          0.002201 -0.012134
earliest_cr_line_year    -0.183904  …         -0.001837 -0.253846
issue_month              -0.009089  …          0.021737 -0.008518
issue_year                0.070702  …         -0.015302 -0.128347
loan_status_num           0.073409  …          0.012268  0.076418

                        pub_rec_bankruptcies      city     state  \
loan_amnt                          -0.110183 -0.019153  0.000284
term                                0.020231  0.056217 -0.001160
int_rate                            0.055871 -0.081581 -0.001257
installment                        -0.095260 -0.012297 -0.000895
grade                              -0.058660  0.085361  0.002602
sub_grade                          -0.060984  0.087603  0.002793
emp_title                          -0.064795  0.150176  0.002707
emp_length                          0.055348  0.002534  0.000844
home_ownership                     -0.005551  0.021754  0.002241
annual_inc                         -0.065831  0.024995  0.003331
verification_status                -0.038020  0.028669 -0.001526
```

```
purpose                       -0.004380  0.019822  0.003526
title                         -0.061732  0.068719  0.003293
dti                           -0.029619 -0.045763 -0.001368
open_acc                      -0.024797 -0.008729 -0.000886
pub_rec                        0.867734 -0.003502  0.000860
revol_bal                     -0.174262 -0.001643  0.001763
revol_util                    -0.085267 -0.027654  0.001244
total_acc                      0.038954  0.004006  0.001532
initial_list_status           -0.040886  0.004224  0.000014
application_type               0.003433  0.003998 -0.000819
mort_acc                       0.031784  0.026102  0.000334
pub_rec_bankruptcies           1.000000 -0.001518  0.002035
city                          -0.001518  1.000000  0.004077
state                          0.002035  0.004077  1.000000
earliest_cr_line_month        -0.004934  0.001678 -0.001758
earliest_cr_line_year         -0.060817 -0.012999 -0.001643
issue_month                   -0.017454  0.006886  0.001010
issue_year                     0.109584 -0.020507 -0.000382
loan_status_num               -0.008339  0.330576  0.011067

                      earliest_cr_line_month  earliest_cr_line_year  \
loan_amnt                          -0.010600              -0.146889
term                                0.001998               0.029011
int_rate                           -0.002108               0.103764
installment                        -0.009231              -0.121672
grade                               0.001591              -0.110854
sub_grade                           0.001625              -0.114307
emp_title                           0.007965              -0.032221
emp_length                         -0.023425              -0.235148
home_ownership                     -0.012427              -0.187590
annual_inc                         -0.013022              -0.183904
verification_status                 0.008409               0.064272
purpose                            -0.000553              -0.012605
title                               0.004739              -0.044627
dti                                 0.004380              -0.012563
open_acc                            0.007328              -0.117138
pub_rec                            -0.005999              -0.063598
revol_bal                          -0.008574              -0.224238
revol_util                          0.004973              -0.007483
total_acc                           0.003878              -0.271086
initial_list_status                 0.001466              -0.026916
application_type                    0.002201              -0.001837
mort_acc                           -0.012134              -0.253846
pub_rec_bankruptcies               -0.004934              -0.060817
city                                0.001678              -0.012999
state                              -0.001758              -0.001643
earliest_cr_line_month              1.000000              -0.012946
```

```
earliest_cr_line_year               -0.012946              1.000000
issue_month                          0.001918              0.000498
issue_year                          -0.007975              0.098615
loan_status_num                      0.003861             -0.038928

                        issue_month  issue_year  loan_status_num
loan_amnt                 -0.019988    0.111667        -0.060762
term                      -0.002117   -0.077146         0.173246
int_rate                  -0.002496    0.036187        -0.243047
installment               -0.021654    0.090908        -0.038797
grade                      0.007271   -0.107772         0.257886
sub_grade                  0.008742   -0.108578         0.263799
emp_title                  0.006752   -0.136912         0.454040
emp_length                -0.000326    0.069729         0.002727
home_ownership             0.002788    0.038313         0.068535
annual_inc                -0.009089    0.070702         0.073409
verification_status       -0.004196   -0.093005         0.085618
purpose                   -0.005016   -0.004960         0.059394
title                      0.066512   -0.416627         0.206052
dti                       -0.003913    0.161702        -0.132441
open_acc                  -0.006366    0.109794        -0.022381
pub_rec                   -0.026835    0.142183        -0.018125
revol_bal                 -0.004920    0.021155        -0.000883
revol_util                -0.004193   -0.053254        -0.082357
total_acc                 -0.007751    0.097357         0.018417
initial_list_status        0.011538   -0.433227         0.009489
application_type           0.021737   -0.015302         0.012268
mort_acc                  -0.008518   -0.128347         0.076418
pub_rec_bankruptcies      -0.017454    0.109584        -0.008339
city                       0.006886   -0.020507         0.330576
state                      0.001010   -0.000382         0.011067
earliest_cr_line_month     0.001918   -0.007975         0.003861
earliest_cr_line_year      0.000498    0.098615        -0.038928
issue_month                1.000000   -0.198280         0.016368
issue_year                -0.198280    1.000000        -0.060502
loan_status_num            0.016368   -0.060502         1.000000

[30 rows x 30 columns]
```

3.5 Multicollinearity and Feature Selection

[54]:
```
df_temp_x=df.drop(columns=['loan_status_num'])
df_temp_y=df[['loan_status_num']]
```

[55]:
```
df_temp_x.head()
```

[55]:
```
   loan_amnt      term  int_rate  installment     grade  sub_grade  emp_title  \
0    10000.0  0.842253     11.44       329.48  0.874263   0.861584   0.757967
1     8000.0  0.842253     11.99       265.68  0.874263   0.844944   0.772208
2    15600.0  0.842253     10.49       506.97  0.874263   0.876619   0.811367
3     7200.0  0.842253      6.49       220.65  0.937100   0.951659   0.821701
4    24375.0  0.680598     17.27       609.33  0.788192   0.754960   0.730792

   emp_length  home_ownership  annual_inc  …  initial_list_status  \
0          10        0.773379    117000.0  …             0.799246
1           4        0.830438     65000.0  …             0.806940
2           0        0.773379     43057.0  …             0.806940
3           6        0.773379     54000.0  …             0.806940
4           9        0.830438     55000.0  …             0.806940

   application_type  mort_acc  pub_rec_bankruptcies      city     state  \
0          0.803913         0                     0  0.819941  0.807981
1          0.803913         1                     0  0.709946  0.802963
2          0.803913         0                     0  0.751210  0.795950
3          0.803913         0                     0  0.849131  0.803902
4          0.803913         1                     0  0.730792  0.804897

   earliest_cr_line_month  earliest_cr_line_year  issue_month  issue_year
0                       6                   1990            1        2015
1                       7                   2004            1        2015
2                       8                   2007            1        2015
3                       9                   2006           11        2014
4                       3                   1999            4        2013

[5 rows x 29 columns]
```

[56]:
```
df_temp_y.head()
```

[56]:
```
   loan_status_num
0                1
1                1
2                1
3                1
4                0
```

[57]:
```
from sklearn.preprocessing import MinMaxScaler
scaler = MinMaxScaler()
df_temp_x = pd.DataFrame(scaler.fit_transform(df_temp_x), columns=df_temp_x.
 ↪columns)
df_temp_x
```

[57]:
```
        loan_amnt  term  int_rate  installment     grade  sub_grade  \
0        0.257509   1.0  0.351813     0.276488  0.848428   0.770021
```

```
1         0.205991   1.0  0.374739     0.222457  0.848428   0.735093
2         0.401757   1.0  0.312213     0.426800  0.848428   0.801578
3         0.185385   1.0  0.145477     0.184323  1.000000   0.959087
4         0.627788   0.0  0.594831     0.513487  0.640812   0.546218
…            …    …       …           …        …        …
396025    0.257509   0.0  0.333055     0.181553  0.848428   0.770021
396026    0.540853   1.0  0.387245     0.590628  0.640812   0.695948
396027    0.128716   1.0  0.291371     0.134077  0.848428   0.853525
396028    0.540853   0.0  0.513130     0.423455  0.640812   0.645963
396029    0.051440   1.0  0.442268     0.055030  0.640812   0.645963

         emp_title  emp_length  home_ownership  annual_inc  …  \
0         0.558312         1.0        0.000000    0.451737  …
1         0.589368         0.4        0.718651    0.250965  …
2         0.674759         0.0        0.000000    0.166243  …
3         0.697294         0.6        0.000000    0.208494  …
4         0.499054         0.9        0.718651    0.212355  …
…            …          …              …          …  …
396025    0.697294         0.2        0.000000    0.154440  …
396026    0.608125         0.5        0.718651    0.424710  …
396027    0.520796         1.0        0.000000    0.218147  …
396028    0.697294         1.0        0.718651    0.247104  …
396029    0.617917         1.0        0.000000    0.166008  …

         initial_list_status  application_type  mort_acc  pub_rec_bankruptcies  \
0                        0.0          0.696997       0.0                   0.0
1                        1.0          0.696997       1.0                   0.0
2                        1.0          0.696997       0.0                   0.0
3                        1.0          0.696997       0.0                   0.0
4                        1.0          0.696997       1.0                   0.0
…                          …                …       …                     …
396025                   0.0          0.696997       0.0                   0.0
396026                   1.0          0.696997       1.0                   0.0
396027                   1.0          0.696997       0.0                   0.0
396028                   1.0          0.696997       1.0                   0.0
396029                   1.0          0.696997       1.0                   0.0

             city     state  earliest_cr_line_month  earliest_cr_line_year  \
0        0.702766  0.592179                0.454545               0.666667
1        0.459686  0.407603                0.545455               0.869565
2        0.550875  0.149638                0.636364               0.913043
3        0.767273  0.442127                0.727273               0.898551
4        0.505753  0.478743                0.181818               0.797101
…           …       …                       …                      …
396025   0.664400  0.462002                0.909091               0.869565
396026   0.634503  0.530580                0.090909               0.898551
396027   0.662119  0.795961                0.181818               0.768116
```

```
396028  0.684284  0.538685                0.909091                0.666667
396029  0.859886  0.366266                0.727273                0.782609

        issue_month  issue_year
0          0.000000    0.888889
1          0.000000    0.888889
2          0.000000    0.888889
3          0.909091    0.777778
4          0.272727    0.666667
…             …          …
396025     0.818182    0.888889
396026     0.090909    0.888889
396027     0.818182    0.666667
396028     0.636364    0.555556
396029     0.454545    0.333333

[396030 rows x 29 columns]
```

[58]:
```
scaler = MinMaxScaler()
df_temp_y = pd.DataFrame(scaler.fit_transform(df_temp_y), columns=df_temp_y.
 ↪columns)
df_temp_y
```

[58]:
```
        loan_status_num
0                   1.0
1                   1.0
2                   1.0
3                   1.0
4                   0.0
…                   …
396025              1.0
396026              1.0
396027              1.0
396028              1.0
396029              1.0

[396030 rows x 1 columns]
```

[59]:
```
from statsmodels.stats.outliers_influence import variance_inflation_factor
from statsmodels.tools.tools import add_constant
df_temp_x=add_constant(df_temp_x)
vif = pd.DataFrame()
vif['Feature'] = df_temp_x.columns
vif['VIF'] = [variance_inflation_factor(df_temp_x.values, i) for i in␣
 ↪range(df_temp_x.shape[1])]
vif = vif.sort_values(by='VIF', ascending=False)
vif
```

[59]:

	Feature	VIF
0	const	1625.563249
6	sub_grade	32.345017
5	grade	22.598371
3	int_rate	10.347590
1	loan_amnt	4.771016
16	pub_rec	4.149083
23	pub_rec_bankruptcies	4.087171
4	installment	3.721154
22	mort_acc	1.794233
29	issue_year	1.765517
19	total_acc	1.734525
15	open_acc	1.675852
17	revol_bal	1.668048
9	home_ownership	1.665729
2	term	1.650757
13	title	1.593266
10	annual_inc	1.546512
18	revol_util	1.420271
14	dti	1.377615
12	purpose	1.302943
20	initial_list_status	1.272294
27	earliest_cr_line_year	1.247537
11	verification_status	1.168488
7	emp_title	1.123738
8	emp_length	1.112096
28	issue_month	1.056882
24	city	1.031277
21	application_type	1.006228
26	earliest_cr_line_month	1.001762
25	state	1.000105

[60]:
```
df_temp_x.drop(columns = ['sub_grade', 'const'], axis=1, inplace=True)
```

[61]:
```
df_temp_x=add_constant(df_temp_x)
vif = pd.DataFrame()
vif['Feature'] = df_temp_x.columns
vif['VIF'] = [variance_inflation_factor(df_temp_x.values, i) for i in␣
 ↪range(df_temp_x.shape[1])]
vif = vif.sort_values(by='VIF', ascending=False)
vif
```

[61]:

	Feature	VIF
0	const	1538.011256
5	grade	7.582907
3	int_rate	7.471774
1	loan_amnt	4.770767

```
15                 pub_rec     4.146769
22    pub_rec_bankruptcies     4.086308
4              installment     3.721146
21                mort_acc     1.789063
28              issue_year     1.759245
18               total_acc     1.734227
14                open_acc     1.675848
16               revol_bal     1.667812
8           home_ownership     1.662241
2                     term     1.629597
12                   title     1.587084
9               annual_inc     1.546505
17              revol_util     1.419901
13                     dti     1.377080
11                 purpose     1.302852
19     initial_list_status     1.271288
26   earliest_cr_line_year     1.246315
10     verification_status     1.167516
6                emp_title     1.123109
7               emp_length     1.111705
27             issue_month     1.056849
23                    city     1.031229
20        application_type     1.005966
25  earliest_cr_line_month     1.001756
24                   state     1.000095
```

[62]:
```
df_temp_x.drop(columns = ['grade', 'const'], axis=1, inplace=True)
df_temp_x=add_constant(df_temp_x)
vif = pd.DataFrame()
vif['Feature'] = df_temp_x.columns
vif['VIF'] = [variance_inflation_factor(df_temp_x.values, i) for i in␣
 ↪range(df_temp_x.shape[1])]
vif = vif.sort_values(by='VIF', ascending=False)
vif
```

[62]:
```
                   Feature          VIF
0                    const  1384.673971
1                loan_amnt     4.770683
14                 pub_rec     4.144219
21    pub_rec_bankruptcies     4.084582
4              installment     3.721009
20                mort_acc     1.777747
17               total_acc     1.733500
27              issue_year     1.730062
13                open_acc     1.675779
15               revol_bal     1.667381
7           home_ownership     1.656570
```

```
2                     term     1.581934
11                   title     1.578630
3                int_rate     1.575759
8              annual_inc     1.546417
16             revol_util     1.419671
12                    dti     1.376214
10                purpose     1.302670
18    initial_list_status     1.271068
25  earliest_cr_line_year     1.245204
9     verification_status     1.167501
5               emp_title     1.122078
6              emp_length     1.110733
26            issue_month     1.056083
22                   city     1.031136
19       application_type     1.005399
24 earliest_cr_line_month     1.001744
23                  state     1.000083
```

Rest of the column's VIF is less than 5 so Feature selection based on vif is complete

[63]:
```
df_temp_x.drop(columns = 'const', axis=1, inplace=True)
```

[64]:
```
df_temp_x.head()
```

[64]:
```
   loan_amnt  term  int_rate  installment  emp_title  emp_length  \
0   0.257509   1.0  0.351813     0.276488   0.558312         1.0
1   0.205991   1.0  0.374739     0.222457   0.589368         0.4
2   0.401757   1.0  0.312213     0.426800   0.674759         0.0
3   0.185385   1.0  0.145477     0.184323   0.697294         0.6
4   0.627788   0.0  0.594831     0.513487   0.499054         0.9

   home_ownership  annual_inc  verification_status   purpose  …  \
0        0.000000    0.451737              1.00000  0.607154  …
1        0.718651    0.250965              1.00000  0.502267  …
2        0.000000    0.166243              0.11019  0.735081  …
3        0.000000    0.208494              1.00000  0.735081  …
4        0.718651    0.212355              0.00000  0.735081  …

   initial_list_status  application_type  mort_acc  pub_rec_bankruptcies  \
0                  0.0          0.696997       0.0                   0.0
1                  1.0          0.696997       1.0                   0.0
2                  1.0          0.696997       0.0                   0.0
3                  1.0          0.696997       0.0                   0.0
4                  1.0          0.696997       1.0                   0.0

       city     state  earliest_cr_line_month  earliest_cr_line_year  \
0  0.702766  0.592179                0.454545               0.666667
1  0.459686  0.407603                0.545455               0.869565
```

```
2  0.550875  0.149638                0.636364                0.913043
3  0.767273  0.442127                0.727273                0.898551
4  0.505753  0.478743                0.181818                0.797101

   issue_month  issue_year
0     0.000000    0.888889
1     0.000000    0.888889
2     0.000000    0.888889
3     0.909091    0.777778
4     0.272727    0.666667

[5 rows x 27 columns]
```

[65]:
```
df.drop(columns=['grade', 'sub_grade'], inplace=True)
```

[66]:
```
df.head()
```

[66]:
```
   loan_amnt      term  int_rate  installment  emp_title  emp_length  \
0    10000.0  0.842253     11.44       329.48   0.757967          10
1     8000.0  0.842253     11.99       265.68   0.772208           4
2    15600.0  0.842253     10.49       506.97   0.811367           0
3     7200.0  0.842253      6.49       220.65   0.821701           6
4    24375.0  0.680598     17.27       609.33   0.730792           9

   home_ownership  annual_inc  verification_status   purpose  …  \
0        0.773379    117000.0             0.853636  0.810739  …
1        0.830438     65000.0             0.853636  0.792587  …
2        0.773379     43057.0             0.785258  0.832878  …
3        0.773379     54000.0             0.853636  0.832878  …
4        0.830438     55000.0             0.776791  0.832878  …

   application_type  mort_acc  pub_rec_bankruptcies      city     state  \
0          0.803913         0                     0  0.819941  0.807981
1          0.803913         1                     0  0.709946  0.802963
2          0.803913         0                     0  0.751210  0.795950
3          0.803913         0                     0  0.849131  0.803902
4          0.803913         1                     0  0.730792  0.804897

   earliest_cr_line_month  earliest_cr_line_year  issue_month  issue_year  \
0                       6                   1990            1        2015
1                       7                   2004            1        2015
2                       8                   2007            1        2015
3                       9                   2006           11        2014
4                       3                   1999            4        2013

   loan_status_num
0                1
```

```
1                1
2                1
3                1
4                0

[5 rows x 28 columns]
```

4 Model Building

[67]:
```
x=df.drop(columns='loan_status_num')
y=df['loan_status_num']
```

[68]:
```
from sklearn.model_selection import train_test_split
x_train, x_temp, y_train, y_temp = train_test_split(x, y, test_size=0.3,␣
 ↪random_state=42, stratify=y)
x_val, x_test, y_val, y_test = train_test_split(x_temp, y_temp, test_size=0.5,␣
 ↪random_state=49, stratify=y_temp)
```

[69]:
```
x_train
```

[69]:
```
        loan_amnt      term  int_rate  installment  emp_title  emp_length  \
214484     5000.0  0.842253     15.31       174.09   0.751081          10
137516    24650.0  0.680598     24.50       716.31   0.896774          10
381950     7200.0  0.842253     12.12       239.56   0.821701          10
266047    21625.0  0.842253     11.14       709.41   0.821701           1
340704    18000.0  0.842253     19.52       664.56   0.821701           8
…             …         …         …            …          …           …
47268     20000.0  0.680598     18.55       513.88   0.751081          10
272632    35000.0  0.842253     13.99         3.00   0.847643           3
176286    10000.0  0.842253      8.90       317.54   0.751081          10
21915     18000.0  0.842253      7.89       563.15   0.821701           0
257563    24000.0  0.842253      5.32       722.76   0.902292           4

        home_ownership  annual_inc  verification_status   purpose  …  \
214484        0.773379     51000.0             0.776791  0.832878  …
137516        0.793199     65000.0             0.785258  0.792587  …
381950        0.830438     60000.0             0.853636  0.832878  …
266047        0.830438     99000.0             0.776791  0.832878  …
340704        0.830438    102000.0             0.853636  0.792587  …
…                  …           …                    …         …  …
47268         0.830438    110000.0             0.776791  0.792587  …
272632        0.830438         3.0             0.785258  0.832878  …
176286        0.793199     84872.0             0.785258  0.792587  …
21915         0.773379     65000.0             0.853636  0.832878  …
257563        0.830438    210000.0             0.785258  0.792587  …
```

```
        initial_list_status  application_type  mort_acc  pub_rec_bankruptcies  \
214484             0.806940          0.803913         1                     0
137516             0.806940          0.803913         0                     0
381950             0.806940          0.803913         1                     0
266047             0.799246          0.803913         1                     0
340704             0.806940          0.803913         1                     0
…                     …                 …       …                     …
47268              0.806940          0.803913         1                     0
272632             0.806940          0.803913         1                     0
176286             0.806940          0.803913         0                     0
21915              0.799246          0.803913         0                     0
257563             0.799246          0.803913         1                     0

            city     state  earliest_cr_line_month  earliest_cr_line_year  \
214484  0.800500  0.804831                       8                   2000
137516  0.804325  0.804831                       8                   2001
381950  0.856069  0.801482                      11                   1986
266047  0.767957  0.802963                       9                   2002
340704  0.825806  0.806538                       5                   1999
…          …       …                      …                   …
47268   0.804346  0.799207                       1                   1999
272632  0.830212  0.797226                       9                   2002
176286  0.865396  0.808846                       5                   1967
21915   0.802581  0.802391                       8                   2003
257563  0.840774  0.797226                      10                   1999

        issue_month  issue_year
214484           12        2012
137516            8        2014
381950            1        2013
266047            1        2013
340704            8        2013
…               …           …
47268            11        2013
272632            8        2016
176286           11        2013
21915             2        2016
257563           10        2015

[277221 rows x 27 columns]
```

[70]:
```
x_val
```

[70]:
```
        loan_amnt      term  int_rate  installment  emp_title  emp_length  \
266476    21000.0  0.842253      6.49       643.54   0.849131           3
301543    18000.0  0.842253      9.67       578.03   0.793280           9
37882     35000.0  0.680598     23.40       994.73   0.793280          10
```

```
292166     5000.0  0.842253      9.76       160.78   0.751081          10
269604     7000.0  0.842253     11.99       232.47   0.871982           7
…             …       …       …          …          …        …
7497       7500.0  0.842253      7.69       233.96   0.875156          10
219714     1500.0  0.842253     22.20        57.45   0.730792           8
395098    16525.0  0.842253     14.49       568.73   0.768677           4
388791     6000.0  0.842253     12.49       200.70   0.755456          10
233101    13200.0  0.680598     16.99       327.99   0.836559          10

        home_ownership  annual_inc  verification_status   purpose  …  \
266476        0.830438    150000.0             0.785258  0.832878  …
301543        0.830438     60507.0             0.853636  0.832878  …
37882         0.830438     94700.0             0.785258  0.792587  …
292166        0.773379     36000.0             0.776791  0.787829  …
269604        0.830438    110000.0             0.776791  0.792587  …
…                ….          …                    …       …  …
7497          0.773379    100000.0             0.785258  0.792587  …
219714        0.773379     43000.0             0.776791  0.787829  …
395098        0.830438     57000.0             0.785258  0.792587  …
388791        0.793199     29000.0             0.785258  0.832878  …
233101        0.830438    113000.0             0.776791  0.829910  …

        initial_list_status  application_type  mort_acc  pub_rec_bankruptcies  \
266476             0.806940          0.803913         1                     0
301543             0.806940          0.803913         0                     0
37882              0.806940          0.803913         1                     0
292166             0.799246          0.803913         0                     0
269604             0.806940          0.803913         1                     0
…                     …                 …       …                    …
7497               0.799246          0.803913         1                     1
219714             0.806940          0.803913         0                     0
395098             0.799246          0.803913         1                     0
388791             0.799246          0.803913         1                     0
233101             0.806940          0.803913         1                     0

            city     state  earliest_cr_line_month  earliest_cr_line_year  \
266476  0.800387  0.804442                       1                   1999
301543  0.728446  0.797833                       2                   2001
37882   0.804346  0.805229                      12                   2002
292166  0.884630  0.805762                       4                   1994
269604  0.747678  0.801482                       9                   1993
…          …       …                        …                      …
7497    0.836559  0.802008                       3                   1997
219714  0.801548  0.805649                       7                   2000
395098  0.869247  0.800640                       4                   2001
388791  0.804346  0.804831                       6                   1998
233101  0.804325  0.799207                       9                   2002
```

```
        issue_month  issue_year
266476            9        2014
301543           12        2013
37882            10        2013
292166           12        2015
269604            1        2015
…               …           …
7497             10        2014
219714            7        2013
395098            6        2014
388791            4        2014
233101            7        2014

[59404 rows x 27 columns]
```

[71]:
```
x_test
```

[71]:
```
        loan_amnt      term  int_rate  installment  emp_title  emp_length  \
53855     16800.0  0.680598      8.18       342.10   0.782522           9
370313     5000.0  0.842253      9.32       159.74   0.779989          10
11669     28000.0  0.680598     24.08       806.81   0.781464          10
211404    10000.0  0.842253      7.90       312.91   0.730792           5
330847    25000.0  0.842253     12.42       835.39   0.821701           6
…             …       …        …            …          …                …
321130    15000.0  0.842253      5.32       451.73   0.836559          10
39951      7000.0  0.842253      7.89       219.00   0.854603          10
252090    12000.0  0.842253      8.19       377.09   0.821701          10
73327     17500.0  0.680598     11.12       381.55   0.849131           3
365343    10000.0  0.842253     16.99       356.48   0.730792          10

        home_ownership  annual_inc  verification_status   purpose  …  \
53855         0.793199    110000.0             0.853636  0.829910  …
370313        0.830438     50004.0             0.853636  0.829910  …
11669         0.830438    102000.0             0.776791  0.792587  …
211404        0.773379     97600.0             0.776791  0.829910  …
330847        0.830438     54996.0             0.776791  0.792587  …
…                …           …                    …       …  …
321130        0.830438    136000.0             0.853636  0.792587  …
39951         0.773379     57000.0             0.776791  0.792587  …
252090        0.793199     43000.0             0.853636  0.792587  …
73327         0.830438     92700.0             0.776791  0.792587  …
365343        0.773379     45000.0             0.785258  0.792587  …

        initial_list_status  application_type  mort_acc  pub_rec_bankruptcies  \
53855              0.806940          0.803913         1                     0
370313             0.806940          0.803913         1                     0
```

```
11669              0.799246          0.803913         1                    0
211404             0.806940          0.803913         1                    0
330847             0.806940          0.803913         1                    0
…                     …                 …       …                    …
321130             0.806940          0.803913         1                    0
39951              0.799246          0.803913         0                    0
252090             0.806940          0.803913         1                    0
73327              0.806940          0.803913         1                    0
365343             0.806940          0.803913         1                    0

            city     state  earliest_cr_line_month  earliest_cr_line_year  \
53855   0.803750  0.803902                       9                   2003
370313  0.836559  0.797226                       7                   1990
11669   0.815910  0.806063                       5                   2000
211404  0.717051  0.806063                       6                   2000
330847  0.867312  0.806527                      11                   1998
…          …       …                        …                   …
321130  0.653015  0.804580                       9                   1997
39951   0.772208  0.803320                      11                   1999
252090  0.821701  0.803412                       8                   1987
73327   0.821701  0.806307                      12                   1992
365343  0.816248  0.808846                       4                   1996

        issue_month  issue_year
53855             5        2015
370313            7        2009
11669            10        2014
211404            8        2012
330847           12        2011
…              …         …
321130            7        2015
39951            10        2015
252090            2        2015
73327            10        2010
365343            8        2014

[59405 rows x 27 columns]
```

[72]:
```
scaler = MinMaxScaler()
x_train = pd.DataFrame(scaler.fit_transform(x_train), columns=x_train.columns)
x_val = pd.DataFrame(scaler.fit_transform(x_val), columns=x_val.columns)
x_test = pd.DataFrame(scaler.fit_transform(x_test), columns=x_test.columns)
```

[73]:
```
x_train
```

[73]:
```
        loan_amnt  term  int_rate  installment  emp_title  emp_length  \
0        0.128716   1.0  0.513130     0.144892   0.543297         1.0
```

```
1         0.634872   0.0  0.896207     0.604085   0.861002          1.0
2         0.185385   1.0  0.380158     0.200337   0.697294          1.0
3         0.556952   1.0  0.339308     0.598242   0.697294          0.1
4         0.463577   1.0  0.688620     0.560259   0.697294          0.8
…            …    …       …            …          …                 …
277216    0.515095   0.0  0.648187     0.432652   0.543297          1.0
277217    0.901473   1.0  0.458108     0.000000   0.753863          0.3
277218    0.257509   1.0  0.245936     0.266376   0.543297          1.0
277219    0.463577   1.0  0.203835     0.474378   0.697294          0.0
277220    0.618129   1.0  0.096707     0.609548   0.873035          0.4

        home_ownership  annual_inc  verification_status   purpose  …  \
0             0.000000    0.196902              0.00000  0.735081  …
1             0.249632    0.250957              0.11019  0.502267  …
2             0.718651    0.231651              1.00000  0.735081  …
3             0.718651    0.382232              0.00000  0.735081  …
4             0.718651    0.393815              1.00000  0.502267  …
…                …          …                     …        …  …
277216        0.718651    0.424704              0.00000  0.502267  …
277217        0.718651    0.000000              0.11019  0.735081  …
277218        0.249632    0.327683              0.11019  0.502267  …
277219        0.000000    0.250957              1.00000  0.735081  …
277220        0.718651    0.810809              0.11019  0.502267  …

        initial_list_status  application_type  mort_acc  pub_rec_bankruptcies  \
0                       1.0          0.696997       1.0                   0.0
1                       1.0          0.696997       0.0                   0.0
2                       1.0          0.696997       1.0                   0.0
3                       0.0          0.696997       1.0                   0.0
4                       1.0          0.696997       1.0                   0.0
…                         …             …          …                     …
277216                  1.0          0.696997       1.0                   0.0
277217                  1.0          0.696997       1.0                   0.0
277218                  1.0          0.696997       0.0                   0.0
277219                  0.0          0.696997       0.0                   0.0
277220                  0.0          0.696997       1.0                   0.0

            city     state  earliest_cr_line_month  earliest_cr_line_year  \
0       0.659803  0.476294                0.636364               0.800000
1       0.668254  0.476294                0.636364               0.815385
2       0.782605  0.353093                0.909091               0.584615
3       0.587885  0.407603                0.727273               0.830769
4       0.715727  0.539074                0.363636               0.784615
…          …       …                       …                      …
277216  0.668301  0.269411                0.000000               0.784615
277217  0.725464  0.196555                0.727273               0.830769
277218  0.803216  0.623984                0.363636               0.292308
```

```
277219  0.664400  0.386529                0.636364                0.846154
277220  0.748805  0.196555                0.818182                0.784615

        issue_month  issue_year
0          1.000000    0.555556
1          0.636364    0.777778
2          0.000000    0.666667
3          0.000000    0.666667
4          0.636364    0.666667
…                …           …
277216     0.909091    0.666667
277217     0.636364    1.000000
277218     0.909091    0.666667
277219     0.090909    1.000000
277220     0.818182    0.888889

[277221 rows x 27 columns]
```

4.1 Handling Imbalance Class

[74]:
```
from imblearn.over_sampling import SMOTE
```

[75]:
```
smote = SMOTE(random_state=42)
```

[76]:
```
x_train, y_train = smote.fit_resample(x_train, y_train)
```

[77]:
```
x_train
```

[77]:
```
        loan_amnt  term  int_rate  installment  emp_title  emp_length  \
0        0.128716   1.0  0.513130     0.144892   0.543297    1.000000
1        0.634872   0.0  0.896207     0.604085   0.861002    1.000000
2        0.185385   1.0  0.380158     0.200337   0.697294    1.000000
3        0.556952   1.0  0.339308     0.598242   0.697294    0.100000
4        0.463577   1.0  0.688620     0.560259   0.697294    0.800000
…             …   …         …            …          …           …
445695   0.299019   1.0  0.355996     0.321753   0.661282    0.935254
445696   0.535668   0.0  0.508520     0.418101   0.506933    0.906709
445697   0.335064   1.0  0.567302     0.387635   0.499054    0.000000
445698   0.901473   0.0  0.829298     0.831385   0.612340    0.099592
445699   0.754320   0.0  0.628546     0.628428   0.722047    1.000000

        home_ownership  annual_inc  verification_status   purpose  …  \
0             0.000000    0.196902             0.000000  0.735081  …
1             0.249632    0.250957             0.110190  0.502267  …
2             0.718651    0.231651             1.000000  0.735081  …
3             0.718651    0.382232             0.000000  0.735081  …
4             0.718651    0.393815             1.000000  0.502267  …
```

```
…                …          …                    …       …  …
445695        0.718651    0.232764             0.074518  0.502267  …
445696        0.249632    0.247614             0.007393  0.719460  …
445697        0.000000    0.205321             0.037129  0.647370  …
445698        0.000000    0.716048             0.110190  0.619149  …
445699        0.718651    0.292239             0.110190  0.502267  …

        initial_list_status  application_type  mort_acc  pub_rec_bankruptcies  \
0                       1.0          0.696997       1.0                   0.0
1                       1.0          0.696997       0.0                   0.0
2                       1.0          0.696997       1.0                   0.0
3                       0.0          0.696997       1.0                   0.0
4                       1.0          0.696997       1.0                   0.0
…                      …                 …       …                     …
445695                  1.0          0.696997       1.0                   0.0
445696                  1.0          0.696997       1.0                   1.0
445697                  1.0          0.696997       0.0                   0.0
445698                  1.0          0.696997       0.0                   0.0
445699                  0.0          0.696997       1.0                   0.0

            city     state  earliest_cr_line_month  earliest_cr_line_year  \
0       0.659803  0.476294                0.636364               0.800000
1       0.668254  0.476294                0.636364               0.815385
2       0.782605  0.353093                0.909091               0.584615
3       0.587885  0.407603                0.727273               0.830769
4       0.715727  0.539074                0.363636               0.784615
…          …       …                      …                     …
445695  0.393608  0.101195                0.847612               0.784172
445696  0.610381  0.585980                0.896892               0.858442
445697  0.565617  0.612224                0.182806               0.743813
445698  0.587359  0.404194                0.091280               0.753658
445699  0.526066  0.413078                0.389746               0.803354

        issue_month  issue_year
0          1.000000    0.555556
1          0.636364    0.777778
2          0.000000    0.666667
3          0.000000    0.666667
4          0.636364    0.666667
…             …         …
445695     0.483976    0.813748
445696     0.987801    0.681577
445697     1.000000    0.555556
445698     0.909091    0.666667
445699     0.909091    0.888889

[445700 rows x 27 columns]
```

[78]:
```
y_train
```

[78]:
```
0         1
1         1
2         1
3         1
4         1
         ..
445695    0
445696    0
445697    0
445698    0
445699    0
Name: loan_status_num, Length: 445700, dtype: int64
```

[79]:
```
from sklearn.linear_model import LogisticRegression
```

[80]:
```
model=LogisticRegression()
model.fit(x_train, y_train)
```

[80]:
```
LogisticRegression()
```

[81]:
```
coefficients = model.coef_[0]
feature_names = x_train.columns
coef_df = pd.DataFrame({
    'Feature': feature_names,
    'Coefficient': coefficients
})
coef_df['Abs_Coefficient'] = coef_df['Coefficient'].abs()
coef_df_sorted = coef_df.sort_values(by='Abs_Coefficient', ascending=False)
print(coef_df_sorted)
```

```
                  Feature  Coefficient  Abs_Coefficient
4               emp_title    12.520493        12.520493
21                   city     9.101287         9.101287
10                  title     6.368007         6.368007
2                 int_rate    -2.058853         2.058853
11                     dti    -1.900265         1.900265
9                  purpose    -1.417275         1.417275
26              issue_year     1.267377         1.267377
15              revol_util    -1.038355         1.038355
0                loan_amnt    -0.907050         0.907050
12                open_acc    -0.851476         0.851476
1                     term     0.543733         0.543733
14               revol_bal     0.519647         0.519647
24  earliest_cr_line_year    -0.425742         0.425742
6           home_ownership     0.410089         0.410089
16                total_acc     0.332927         0.332927
```

```
22                  state     0.302856         0.302856
5               emp_length     0.259336         0.259336
20   pub_rec_bankruptcies     0.215582         0.215582
7                annual_inc     0.215224         0.215224
18       application_type     0.203957         0.203957
13                pub_rec    -0.161163         0.161163
25            issue_month     0.150082         0.150082
8      verification_status     0.111595         0.111595
17     initial_list_status    -0.080764         0.080764
3               installment     0.057546         0.057546
23  earliest_cr_line_month    -0.021282         0.021282
19                mort_acc     0.008922         0.008922
```

[82]:
```
model.intercept_
```

[82]:
```
array([-15.49766984])
```

[83]:
```
accuracy = model.score(x_val, y_val)
print(f"Accuracy on validation set: {accuracy * 100:.2f}%")
```

```
Accuracy on validation set: 81.31%
```

[84]:
```
accuracy = model.score(x_test, y_test)
print(f"Accuracy on test set: {accuracy * 100:.2f}%")
```

```
Accuracy on test set: 81.66%
```

4.2 Key Insights from model Coeeficients:

- Significant Positive Coefficients:
  - emp_title (12.52): This feature has the highest positive coefficient, meaning that higher values of emp_title strongly increase the likelihood of the positive class (e.g., loan approval or a similar outcome in the model). It suggests that the title of employment plays a crucial role in predicting the target.
  - city (9.10): The city feature also has a large positive coefficient, indicating that the location of the applicant significantly influences the outcome. This suggests that certain cities may be associated with higher probabilities of success in the target variable (e.g., loan approval or application acceptance).
  - title (6.37): The title feature (which could refer to the type of loan or its designation) is positively correlated with the outcome. This reinforces the idea that specific loan types or titles are more likely to result in a positive outcome.
- Significant Negative Coefficients:
  - int_rate (-2.06): A higher interest rate decreases the likelihood of a positive outcome (e.g., loan approval or acceptance). This is expected, as higher rates are typically associated with riskier loans or less favorable conditions.
  - dti (-1.90): The Debt-to-Income ratio (dti) also has a negative coefficient, meaning that higher debt-to-income ratios reduce the likelihood of the positive class. This is logical, as individuals with higher DTI ratios are seen as riskier candidates for approval.

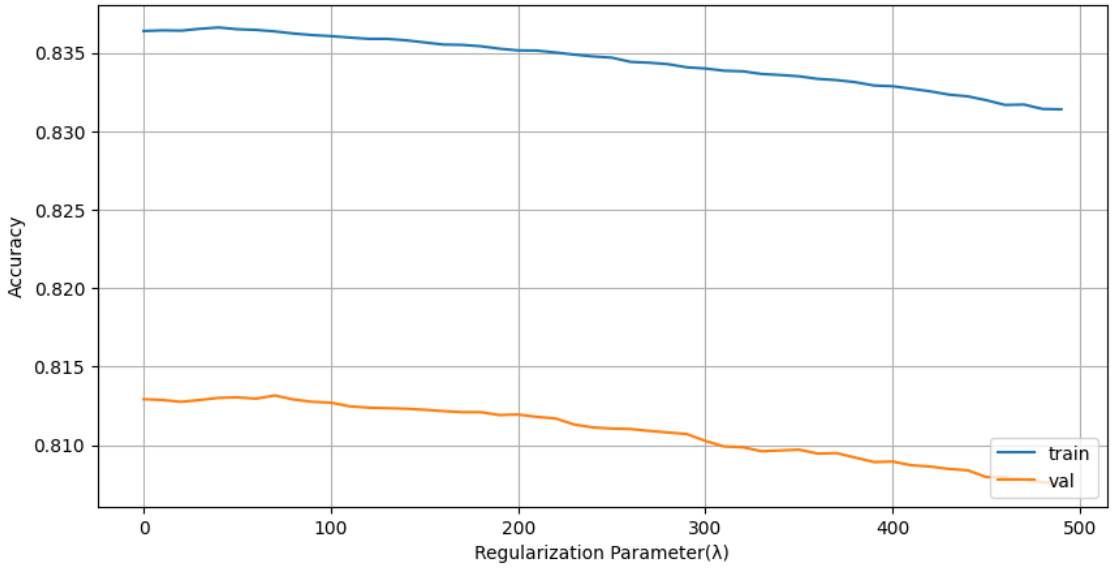
  - purpose (-1.42): The purpose of the loan (e.g., debt consolidation, home improvement, etc.) negatively affects the outcome, suggesting that certain loan purposes are less likely to lead to the positive class.
  - revol_util (-1.04): Higher revolving utilization (revol_util) negatively impacts the outcome, which indicates that applicants with higher credit card utilization may have a lower likelihood of success in the model.
- Moderate Positive Coefficients:
  - issue_year (1.27): The year the loan was issued has a moderate positive effect on the outcome. Newer loans or loans from a specific time period might be more likely to have a favorable outcome.
  - revol_bal (0.52): The balance of revolving credit positively affects the outcome, though it has a smaller effect compared to other features.
  - loan_amnt (-0.91): The loan amount has a negative impact on the likelihood of a positive class, suggesting that higher loan amounts are less likely to result in the target outcome.
- Small Positive Coefficients:
  - term (0.54): The term of the loan (e.g., 36 months or 60 months) has a small positive impact on the outcome. A longer loan term may slightly increase the likelihood of success.
  - emp_length (0.26): Length of employment has a positive, though relatively smaller, impact on the target variable. Longer employment durations may suggest more financial stability, improving chances of success.
  - home_ownership (0.41): Whether the applicant owns a home also plays a small positive role in the outcome, implying that homeowners may be favored in the model.
  - annual_inc (0.22): A higher annual income also has a minor positive effect on the outcome.
- Negligible Effects:
  - mort_acc (0.01): This feature has a very small positive coefficient, suggesting that it has a negligible effect on the outcome.
  - earliest_cr_line_year (-0.43) and earliest_cr_line_month (-0.02): Both of these features have negative coefficients, but their absolute values are relatively small, implying that the timing of the earliest credit line has a minor impact.
  - initial_list_status (-0.08): The status when the loan was initially listed also has a negligible negative effect.
- Inconsistent Effects:
  - pub_rec_bankruptcies (0.22): This feature, representing the number of bankruptcies on public record, has a small positive effect. It seems counterintuitive, as more bankruptcies are generally seen as risk factors, but the small magnitude suggests it may not be a strong predictor in your model.
  - application_type (0.20): This feature has a small positive effect, which could indicate that the type of application (e.g., individual or joint) has a slight influence on the outcome.

4.3 Hyperparameter Tuning

[85]:
```
from sklearn.pipeline import make_pipeline
train_scores = []
val_scores = []
```

```
for la in np.arange(0.01, 500.0, 10):
  scaled_lr = make_pipeline(scaler, LogisticRegression(C=1/la))
  scaled_lr.fit(x_train, y_train)
  train_score = scaled_lr.score(x_train, y_train)
  val_score = scaled_lr.score(x_val, y_val)
  train_scores.append(train_score)
  val_scores.append(val_score)
```

[86]:
```
plt.figure(figsize=(10,5))
plt.plot(list(np.arange(0.01, 500.0, 10)), train_scores, label="train")
plt.plot(list(np.arange(0.01, 500.0, 10)), val_scores, label="val")
plt.legend(loc='lower right')
plt.xlabel("Regularization Parameter( )")
plt.ylabel("Accuracy")
plt.grid()
plt.show()
```



it looks like at lambda = 90 the accuracy was the highest but the difference is so less, that we can ignore

5 Metrics

[87]:
```
from sklearn.metrics import roc_auc_score, precision_score, recall_score,␣
  ↪f1_score, classification_report, confusion_matrix, ConfusionMatrixDisplay
from sklearn.metrics import RocCurveDisplay
```

5.1 Validation set

[88]:
```
y_val_pred_proba = model.predict_proba(x_val)[:, 1]
y_val_pred = model.predict(x_val)
```

[89]:
```
y_val_pred_proba
```

[89]:
```
array([0.94831111, 0.42522048, 0.3014404 , …, 0.68782433, 0.5442933 ,
       0.81534162], shape=(59404,))
```

[90]:
```
y_val_pred
```

[90]:
```
array([1, 0, 0, …, 1, 1, 1], shape=(59404,))
```

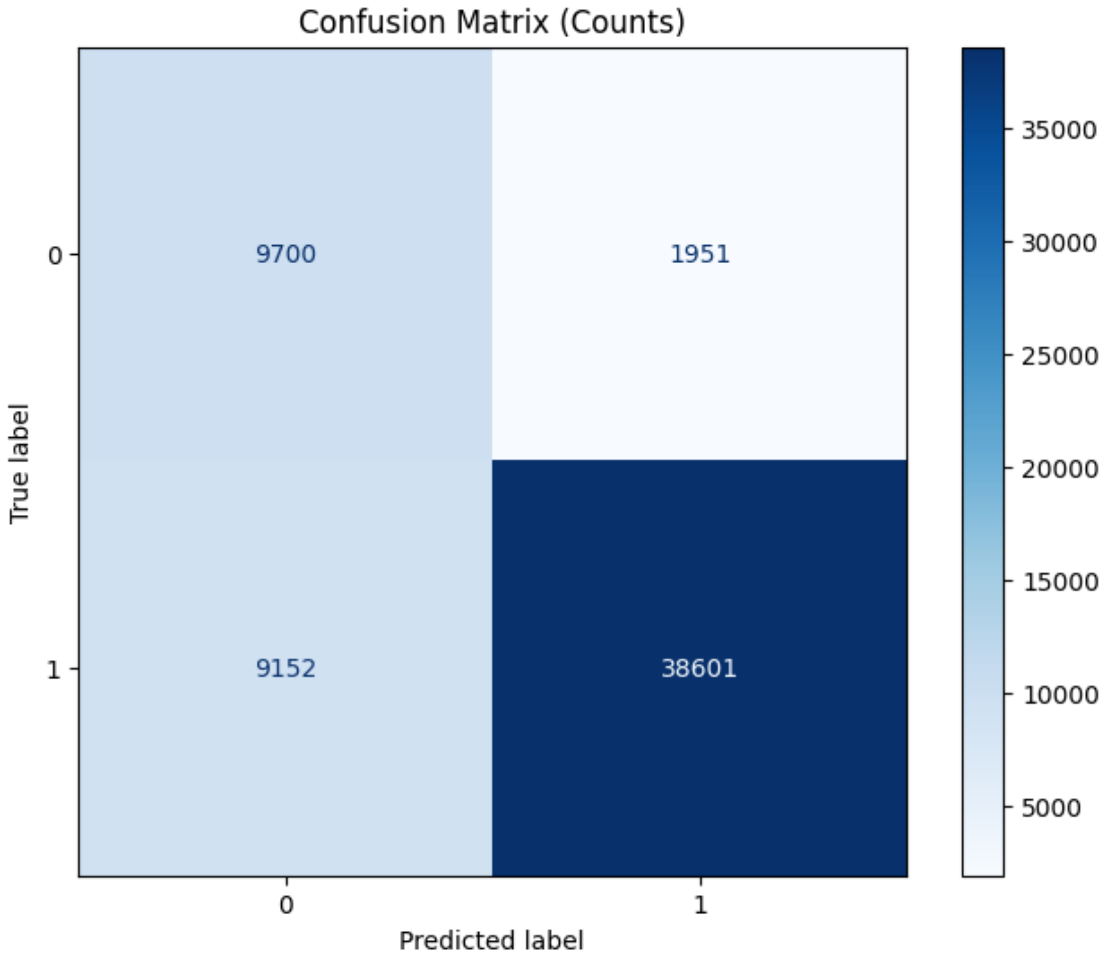
[91]:
```
roc_auc = roc_auc_score(y_val, y_val_pred_proba)
precision = precision_score(y_val, y_val_pred)
recall = recall_score(y_val, y_val_pred)
f1 = f1_score(y_val, y_val_pred)
```

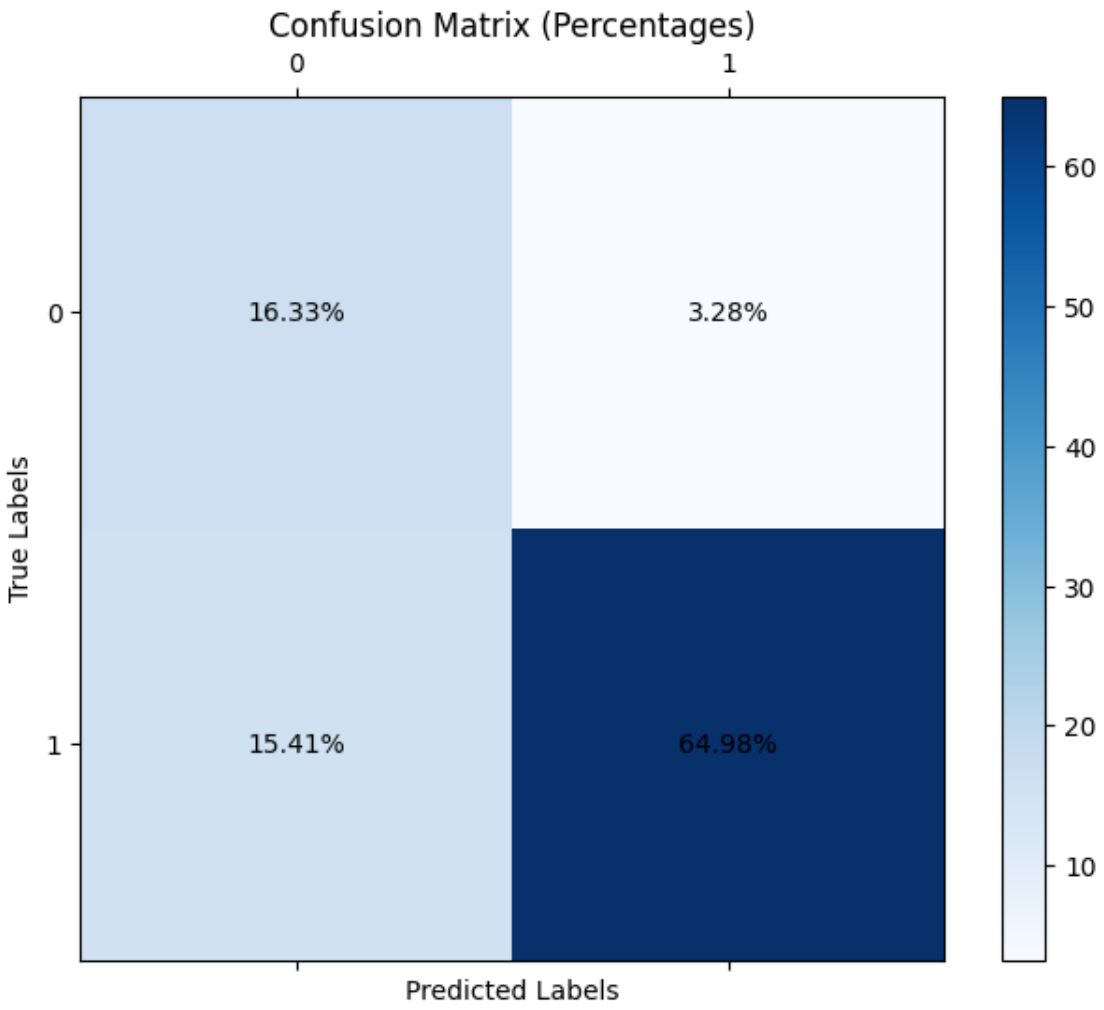
[92]:
```
print("Evaluation Metrics:")
print(f"ROC AUC Score: {roc_auc:.2f}")
print(f"Precision: {precision:.2f}")
print(f"Recall: {recall:.2f}")
print(f"F1 Score: {f1:.2f}")
print("\nClassification Report:")
print(classification_report(y_val, y_val_pred))
```

```
Evaluation Metrics:
ROC AUC Score: 0.89
Precision: 0.95
Recall: 0.81
F1 Score: 0.87

Classification Report:
              precision    recall  f1-score   support

           0       0.51      0.83      0.64     11651
           1       0.95      0.81      0.87     47753

    accuracy                           0.81     59404
   macro avg       0.73      0.82      0.76     59404
weighted avg       0.87      0.81      0.83     59404
```

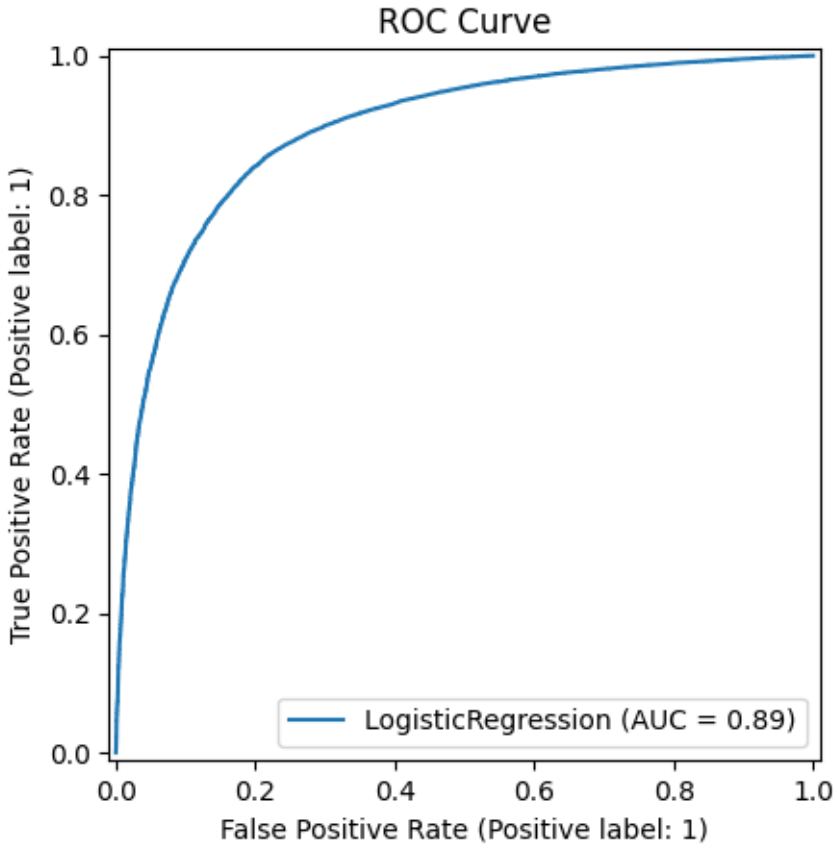
[95]:
```
cm = confusion_matrix(y_val, y_val_pred)
cm_percent = cm.astype('float') / cm.sum() * 100
fig, ax = plt.subplots(figsize=(8, 6))
```

```
ConfusionMatrixDisplay(confusion_matrix=cm, display_labels=model.classes_).
 ↪plot(cmap="Blues", ax=ax)
plt.title("Confusion Matrix (Counts)")
fig, ax = plt.subplots(figsize=(8, 6))
im = ax.matshow(cm_percent, cmap="Blues")
for (i, j), val in np.ndenumerate(cm_percent):
    ax.text(j, i, f"{val:.2f}%", ha="center", va="center", color="black")
plt.title("Confusion Matrix (Percentages)")
plt.xlabel("Predicted Labels")
plt.ylabel("True Labels")
plt.colorbar(im, ax=ax)
plt.show()
```





[94]:
```
RocCurveDisplay.from_estimator(model, x_val, y_val)
plt.title("ROC Curve")
plt.show()
```



5.2 Insights and Analysis of the metrics

- ROC Curve and AUC:
  - The ROC curve shows a strong classifier with an Area Under Curve (AUC) of 0.89. This indicates the model has good separability between the two classes (0 and 1). The closer the AUC is to 1, the better the model is at predicting positive instances as positive and negative ones as negative.
- Confusion Matrix (Percentages):
  - True Negative Rate (16.33%): These are instances where the actual class was 0, and the model predicted 0 correctly.
  - False Positive Rate (3.28%): These are instances where the actual class was 0, but the model incorrectly predicted 1.
  - False Negative Rate (15.41%): These are instances where the actual class was 1, but the model predicted 0.
  - True Positive Rate (64.98%): These are instances where the actual class was 1, and the model predicted 1 correctly.
  - The model performs very well on predicting class 1 but has some misclassification in predicting class 0.
  - This indicates that the model has good accuracy in predicting "Fully Paid" loans (class 1), as shown by the high true positive rate (64.98%). However, the false negative rate

(15.41%) suggests that a significant portion of "Fully Paid" loans are incorrectly classified as "Charged Off" (class 0). The false positive rate (3.28%) is low, indicating good performance for avoiding false alarms for "Charged Off" loans.

- Evaluation Metrics:
  - Precision (0.95): The model is highly precise for class 1 predictions. Out of all instances predicted as 1, 95% are correct.
  - Recall (0.81): The recall is lower than precision, indicating some false negatives exist. This means the model missed some instances of class 1.
  - F1 Score (0.87): A good balance between precision and recall, showing the model is reliable in predicting class 1.
- Classification Report:
  - Class 0 has a lower precision (0.51) and recall (0.83), leading to an F1 score of 0.64. This indicates difficulty in classifying this minority class.
  - Class 1 has excellent metrics, showcasing the model's bias toward the majority class due to the dataset's imbalance.
  - Accuracy (81%): Indicates that the model correctly classifies 81% of all instances.
- Observations:
  - High precision for class 1 suggests the model makes fewer false positive predictions, which is good for minimizing unnecessary interventions or actions in a real-world scenario.
  - The recall for class 1 could be improved further to ensure the model does not miss important cases.

6 Summary

- Business Objectives:
  - Predict if an individual qualifies for a loan using their creditworthiness and risk profile.
  - Optimize loan terms (repayment duration, interest rates, loan amount) to balance borrower satisfaction and lender profitability.
  - Reduce default rates while maximizing loan disbursement for trustworthy borrowers.
- Challenges:
  - Managing imbalanced datasets where most loans are fully paid (80%), with only 20% being charged off. This skews risk prediction.
  - Addressing outliers in key variables like loan amounts, annual incomes, and revolving balances.
  - Accounting for missing data (e.g., mort_acc has 9.54% missing values).
  - Balancing false positives (granting loans to risky applicants) and false negatives (rejecting eligible borrowers).
- KPIs:
  - High AUC (currently 0.89) and F1 scores to ensure robust performance.
  - Monitoring default rates across borrower groups (e.g., employment length, loan purpose).
  - Identifying and reducing high-risk categories like small business loans and Grade G loans.
- Important Positive Coefficients:
  - emp_title (12.52): Borrower employment titles like "Teacher" or "Manager" have a strong positive impact on loan approval likelihood, possibly due to perceived income stability.
  - city (9.10): Loans issued in certain cities show higher approval rates, indicating geographic trends in creditworthiness or lender strategy.

  - title (6.37): Loan titles like "Debt Consolidation" or "Credit Card Refinancing" are associated with higher approval probabilities.
- Important Negative Coefficients:
  - int_rate (-2.06): Higher interest rates are linked to riskier loans, lowering approval chances.
  - dti (-1.90): A higher debt-to-income ratio signals financial strain, reducing approval likelihood.
  - revol_util (-1.04): Elevated revolving utilization (credit usage) indicates credit stress, deterring approval.
- Neutral or Weak Coefficients:
  - mort_acc (0.01): Number of mortgage accounts has minimal impact, likely because its effect is already captured by other variables like home_ownership.
  - application_type (0.20): Application type has a weak effect, though JOINT applications show slightly better repayment outcomes than INDIVIDUAL.
- Visualization Suggestions:
  - ROC Curve and Precision-Recall Curve: Illustrate model performance with an AUC of 0.89 and highlight the trade-offs between precision (0.95) and recall (0.81).
  - Bar Chart for Loan Grades: Show that lower grades (E, F, G) have higher default rates (up to 50.3%), emphasizing credit risk trends.
  - Box Plots for Income and Loan Amounts: Identify outliers in annual income and loan amounts, such as extremely high income values around 745,000.
  - Scatter Plot for Revolving Utilization: Display the correlation between revol_util and default rates to visualize how higher utilization increases risk.
  - Heatmap of Correlations: Highlight strong relationships, such as loan_amnt with installment (95.39%) and total_acc with open_acc (68.07%).
- False Positives (FP):
  - Predicting "Fully Paid" when the borrower defaults leads to financial losses for the lender. Groups like Grade G borrowers and those with high dti or int_rate pose higher risks for FPs.
- False Negatives (FN):
  - Rejecting qualified borrowers (predicted "Charged Off" but actually "Fully Paid") impacts revenue and customer satisfaction. For instance, loans with verified income or stable employment histories may be wrongly rejected.
- Strategies to Balance FP and FN:
  - Adjust the classification threshold to favor precision or recall based on business priorities.
  - Introduce secondary evaluations for borderline cases, especially for groups with moderate risks like Grade C borrowers.
  - Incorporate cost-sensitive metrics into model evaluation to reduce the financial impact of errors.
- Improving Loan Approval Processes:
  - Automate decision-making for low-risk groups, such as borrowers with high income, long employment history, and low dti.
  - Offer customized terms for borderline applicants, such as higher interest rates for borrowers with moderate risk profiles (e.g., Grade C loans).
- Mitigating Risks:
  - Implement stricter eligibility criteria for risky categories, such as small business loans (29.43% default rate).
  - Adjust terms for higher-risk groups, such as increasing interest rates for borrowers with

high revol_util.

- Capitalizing on Opportunities:
  - Promote loans with low default rates, such as wedding loans (12.09%) and car loans (13.47%).
  - Focus on regions and grades with high repayment rates (e.g., Grade A with 93.7% fully paid loans).
- Model Monitoring Techniques:
  - Regularly track AUC, precision, recall, and F1 scores. Focus on reducing the false negative rate (currently 15.41%).
  - Use statistical drift detection to identify changes in data patterns, such as shifts in borrower income distributions or loan purposes.
- Data Refresh and Model Retraining:
  - Retrain the model periodically using updated datasets to reflect trends like rising average loan amounts (from 8,000–12,000 to 14,114).
  - Monitor performance by loan category, especially for evolving trends in grades or employment lengths.
- Feedback Integration:
  - Incorporate real-time repayment data to update risk profiles dynamically.
  - Create a feedback loop to identify patterns in misclassified loans (e.g., analyzing FPs and FNs) and adjust the model accordingly.

7 Questionaire

1. What percentage of customers have fully paid their loan amount?

- **80%** of customers have fully paid their loan, while the remaining **20%** have been charged off.

2. Comment about the correlation between Loan Amount and Installment features.

- There is a **high positive correlation** (**95.39%**) between Loan Amount and Installment. This is expected as the installment amount directly depends on the loan amount and the interest rate. Larger loans naturally have higher installment amounts.

3. The majority of people have homeownership as _____.

- The majority of people have homeownership as **Mortgage** (50%), followed by **Renters** (40%) and a smaller percentage owning their homes outright (9%).

4. People with grades 'A' are more likely to fully pay their loan. (T/F)

- **True.** Loans with grades 'A' have a **93.7%** chance of being fully paid, indicating a low risk of default.

5. Name the top 2 most common job titles.

- The top 2 most common job titles are:
  1. **Teacher**
  2. **Manager**

6. **Thinking from a bank's perspective, which metric should our primary focus be on?**

- **Precision**
  - Precision is critical for the bank to minimize false positives, i.e., approving loans for customers who are likely to default.
  - A high precision (currently 0.95) ensures that most approved loans are repaid, reducing financial losses.

7. **How does the gap in precision and recall affect the bank?**

- **High Precision (0.95):** Reduces false positives, ensuring fewer risky loans are approved, protecting the bank from financial losses.
- **Moderate Recall (0.81):** Indicates some missed opportunities, where eligible borrowers are rejected (false negatives). This can affect customer satisfaction and revenue.
- **Impact:** The gap highlights a trade-off between **risk mitigation** and **business growth**. A focus on improving recall without sacrificing precision is critical for optimizing lending operations.

8. **Which features heavily affected the outcome?**

- **Positive Impact:**
  - **emp_title:** Stable employment titles like "Manager" or "Teacher."
  - **city:** Indicates location-based trends in creditworthiness.
  - **annual_inc:** Higher income is associated with lower default risk.
- **Negative Impact:**
  - **int_rate:** High interest rates indicate riskier loans.
  - **dti:** Higher debt-to-income ratios reduce approval likelihood.
  - **revol_util:** High credit utilization is linked to increased default risk.

9. **Will the results be affected by geographical location? (Yes/No)**

- **Yes.** The model's coefficients indicate that **city** has a high positive impact (**coefficient = 9.10**). Certain cities are associated with higher approval probabilities, suggesting geographical location influences loan outcomes. This could reflect regional economic conditions or borrower characteristics.

[ ]:
```
!jupyter nbconvert --to pdf finte.ipynb
```