In [251…

```
import pandas as pd
from sklearn.cluster import DBSCAN
from sklearn.preprocessing import StandardScaler
import numpy as np
import matplotlib.pyplot as plt
```

Data description

- **loanId** : unique loan identifier.
- **anon_ssn**: This is a hash based on a client's ssn. if it is an ssn to compare if a loan belongs to a previous customer.
- **payFrequency**: repayment frequency of the loan.
- **apr**: the loan apr %
- **applicationDate**: Date of application (start date)
- **originated**: Whether or not a loan has been originated (first step of underwriting before loan is funded)
- **originatedDate**: Date of origination day the loan was originated
- **nPaidOff**: How many MoneyLion loans this client has paid off in the past
- **approved**: Whether or not a loan has been approved (final step of underwriting before a loan deposit is attempted)
- **isFunded**: Whether or not a loan is ultimately funded. –a loan can be voided by a customer shortly after it is approved, so not all approved loans are ultimately funded
- **loanStatus**: Current loan status. Most are self explanatory. Below are the statuses which need **clarification**: application status (**Returned Item**: missed 1 payment (but not more), due to insufficient funds, **Rejected**: Rejected by automated underwriting rules not by human underwriters, **Withdrawn Application** application abandoned for more than 2 weeks, or is withdrawn by a human underwriter or customer, Statuses with the word "void" in them mean a loan that is approved but cancelled. (One reason is the loan failed to be debited into the customer's account).
- **loanAmount**: Principal of loan for non-funded loans this will be the principal in the loan application
- **originallyScheduledPaymentAmount**: This is the originally scheduled repayment amount (if a customer pays off all his scheduled payments, this is the amount we should receive)
- **state**: Client's state
- **Lead type**: The lead type determines the underwriting rules for a lead. **bvMandatory**: leads that are bought from the ping tree required to perform bank verification before loan approval. **lead**: very similar to bvMandatory, except bank verification is optional for loan approval **california**: similar to (ii), but optimized for California lending rules **organic**: customers that came through the MoneyLion website **rc_returning**: customers who have at least 1 paid off loan in another loan portfolio. (The first paid off loan is not in this data set). **prescreen**: preselected customers who have been offered a loan through direct mail campaigns **express**: promotional "express" loans **repeat**: promotional loans offered through sms **instant-offer**: promotional "instant-offer" loans
- **Lead cost**: Cost of the lead

- **fpStatus**: Result of the first payment of the loan: **i. Checked** – payment is successful **ii. Rejected** – payment is unsuccessful **iii. Cancelled** – payment is cancelled **iv. No Payments/No Schedule** – loan is not funded **v. Pending** – ACH attempt has been submitted to clearing house but no response yet **vi. Skipped** – payment has been skipped **vii. None** – No ACH attempt has been made yet – usually because the payment is scheduled for the future
- **clarityFraudId**: unique underwriting id. Can be used to join with columns in the clarity_underwriting_variables.csv file

In [252…
```
loan=pd.read_csv("loan.csv")
```

In [253…
```
loan
```

Out[253]:

	loanId	anon_ssn	payFrequency	apr	applicationDate	o
0	LL-I-07399092	beff4989be82aab4a5b47679216942fd	B	360.0	2016-02-23T17:29:01.940000	
1	LL-I-06644937	464f5d9ae4fa09ece4048d949191865c	B	199.0	2016-01-19T22:07:36.778000	
2	LL-I-10707532	3c174ae9e2505a5f9ddbff9843281845	B	590.0	2016-08-01T13:51:14.709000	
3	LL-I-02272596	9be6f443bb97db7e95fa0c281d34da91	B	360.0	2015-08-06T23:58:08.880000	
4	LL-I-09542882	63b5494f60b5c19c827c7b068443752c	B	590.0	2016-06-05T22:31:34.304000	
…	…	…	…	…	…	
577677	LL-I-12122269	801262d04720d32040612759857f4147	B	590.0	2016-11-08T17:32:33.554000	
577678	LL-I-16183462	e37750de9d99a67e0fa96a51e86fdf5b	S	490.0	2017-01-24T22:20:59.818000	
577679	LL-I-06962710	d7e55e85266208ac4c353f42ebcde5ca	B	590.0	2016-02-02T03:05:47.797000	
577680	LL-I-01253468	c3b35307cb36116bf59574f9138d3dad	B	550.0	2015-05-21T20:19:49.639000	
577681	LL-I-04733921	dc0a43b16c037ee5d0142daebb5db83a	I	590.0	2015-11-17T22:04:20.862000	

577682 rows × 19 columns

Out[253]:

	loanId	anon_ssn	payFrequency	apr	applicationDate	o
0	LL-I-07399092	beff4989be82aab4a5b47679216942fd	B	360.0	2016-02-23T17:29:01.940000	
1	LL-I-06644937	464f5d9ae4fa09ece4048d949191865c	B	199.0	2016-01-19T22:07:36.778000	
2	LL-I-10707532	3c174ae9e2505a5f9ddbff9843281845	B	590.0	2016-08-01T13:51:14.709000	
3	LL-I-02272596	9be6f443bb97db7e95fa0c281d34da91	B	360.0	2015-08-06T23:58:08.880000	
4	LL-I-09542882	63b5494f60b5c19c827c7b068443752c	B	590.0	2016-06-05T22:31:34.304000	
...	...	...	...	...	...	
577677	LL-I-12122269	801262d04720d32040612759857f4147	B	590.0	2016-11-08T17:32:33.554000	
577678	LL-I-16183462	e37750de9d99a67e0fa96a51e86fdf5b	S	490.0	2017-01-24T22:20:59.818000	
577679	LL-I-06962710	d7e55e85266208ac4c353f42ebcde5ca	B	590.0	2016-02-02T03:05:47.797000	
577680	LL-I-01253468	c3b35307cb36116bf59574f9138d3dad	B	550.0	2015-05-21T20:19:49.639000	
577681	LL-I-04733921	dc0a43b16c037ee5d0142daebb5db83a	I	590.0	2015-11-17T22:04:20.862000	

577682 rows × 19 columns

1. This is the sample of the dataset where 577682 rows and 19 columns are avalable

Data description

- **loanId**: This is a unique loan identifier. Use this for joins with the loan.csv file
- **isCollection**: A loan can have a custom made collection plan if the customer has trouble making repayments as per the original schedule. TRUE means the payment is from a custom made collection plan.
- **installmentIndex**: a. This counts the nth payment for the loan. First payment is 1, 2 payment is 2 and so on. nd b. This index resets for collection payment plans. So some loans can have 2 payments with the same installmentIndex. One from the regular plan and one from the collection plan.
- **paymentdate**: a. Effective of payment
- **prinicpal**: principal component of the payment
- **fees**: Fee interest amount of the payment
- **paymentAmount**: Total amount of the payment, Usually equals to fees + principal
- **paymentStatus**: **a. Checked** – payment is successful **b. Rejected** – payment is unsuccessful **c. Cancelled** – payment is cancelled **d. Pending** – ACH attempt has been submitted to clearing house but no response yet **e. Skipped** – payment has been

skipped **f. None** – No ACH attempt has been made yet – usually because the payment is scheduled for the future **g. Rejected awaiting retry** – retrying a failed ACH attempt.

- **paymentReturnCode**: these are ACH error codes to explain why the payment failed. You can find more information about this at the end of this document, or visit the following **link**: https://www.vericheck.com/ach-return-codes/

In [254…
```
payment=pd.read_csv("payment.csv")
```

In [255…
```
payment
```

Out[255]:

	loanId	installmentIndex	isCollection	paymentDate	principal	fees	paymentAmount
0	LL-I-00000021	1	False	2014-12-19T05:00:00	22.33	147.28	169.61
1	LL-I-00000021	2	False	2015-01-02T05:00:00	26.44	143.17	169.61
2	LL-I-00000021	3	False	2015-01-16T05:00:00	31.30	138.31	169.61
3	LL-I-00000021	4	False	2015-01-30T05:00:00	37.07	132.54	169.61
4	LL-I-00000021	5	False	2015-02-13T05:00:00	43.89	125.72	169.61
…	…	…	…	…	…	…	..
689359	LL-I-18629478	8	False	2017-07-14T04:00:00	45.62	17.67	63.29
689360	LL-I-18629478	9	False	2017-07-31T04:00:00	45.67	17.62	63.29
689361	LL-I-18629478	10	False	2017-08-15T04:00:00	51.12	12.17	63.29
689362	LL-I-18629478	11	False	2017-08-31T04:00:00	54.35	8.94	63.29
689363	LL-I-18629478	12	False	2017-09-15T04:00:00	58.83	4.36	63.19

689364 rows × 9 columns

Out[255]:

	loanId	installmentIndex	isCollection	paymentDate	principal	fees	paymentAmount
0	LL-I-00000021	1	False	2014-12-19T05:00:00	22.33	147.28	169.61
1	LL-I-00000021	2	False	2015-01-02T05:00:00	26.44	143.17	169.61
2	LL-I-00000021	3	False	2015-01-16T05:00:00	31.30	138.31	169.61
3	LL-I-00000021	4	False	2015-01-30T05:00:00	37.07	132.54	169.61
4	LL-I-00000021	5	False	2015-02-13T05:00:00	43.89	125.72	169.61
...	...	...	...	...	...	...	..
689359	LL-I-18629478	8	False	2017-07-14T04:00:00	45.62	17.67	63.29
689360	LL-I-18629478	9	False	2017-07-31T04:00:00	45.67	17.62	63.29
689361	LL-I-18629478	10	False	2017-08-15T04:00:00	51.12	12.17	63.29
689362	LL-I-18629478	11	False	2017-08-31T04:00:00	54.35	8.94	63.29
689363	LL-I-18629478	12	False	2017-09-15T04:00:00	58.83	4.36	63.19

689364 rows × 9 columns

This is the sample of payment dataset where 689364 rows and 9 columns are avalable.

Data description

In [256…

```
variables=pd.read_csv("clarity_underwriting_variables.csv")
```

```
C:\Users\aviji\AppData\Local\Temp\ipykernel_12940\3244893591.py:1: DtypeWarning: C
olumns (9,11,12,13,14,15,16,17,18,19,20,21,22,23,25,26,27,28,29,31,32,33,36,37) ha
ve mixed types. Specify dtype option on import or set low_memory=False.
  variables=pd.read_csv("clarity_underwriting_variables.csv")
C:\Users\aviji\AppData\Local\Temp\ipykernel_12940\3244893591.py:1: DtypeWarning: C
olumns (9,11,12,13,14,15,16,17,18,19,20,21,22,23,25,26,27,28,29,31,32,33,36,37) ha
ve mixed types. Specify dtype option on import or set low_memory=False.
  variables=pd.read_csv("clarity_underwriting_variables.csv")
```

In [257…

```
variables
```

Out[257]:

	.underwritingdataclarity.clearfraud.clearfraudinquiry.thirtydaysago	.underwritingdataclarity.cle
0	8.0	
1	5.0	
2	9.0	
3	3.0	
4	5.0	
...	...	
49747	2.0	
49748	6.0	
49749	4.0	
49750	3.0	
49751	5.0	

49752 rows × 54 columns

Out[257]:

	.underwritingdataclarity.clearfraud.clearfraudinquiry.thirtydaysago	.underwritingdataclarity.cle
0	8.0	
1	5.0	
2	9.0	
3	3.0	
4	5.0	
...	...	
49747	2.0	
49748	6.0	
49749	4.0	
49750	3.0	
49751	5.0	

49752 rows × 54 columns

Printing sample data where 49752 rows and 54 columns are avaliable.

In [258…
```
loan.shape
```

Out[258]:
```
(577682, 19)
```

Out[258]:
```
(577682, 19)
```

Printing shape data **Rows**: 577682 **columns**: 19

In [259…
```
loan.columns
```

Out[259]:
```
Index(['loanId', 'anon_ssn', 'payFrequency', 'apr', 'applicationDate',
       'originated', 'originatedDate', 'nPaidOff', 'approved', 'isFunded',
       'loanStatus', 'loanAmount', 'originallyScheduledPaymentAmount', 'state',
       'leadType', 'leadCost', 'fpStatus', 'clarityFraudId', 'hasCF'],
      dtype='object')
```
Out[259]:
```
Index(['loanId', 'anon_ssn', 'payFrequency', 'apr', 'applicationDate',
       'originated', 'originatedDate', 'nPaidOff', 'approved', 'isFunded',
       'loanStatus', 'loanAmount', 'originallyScheduledPaymentAmount', 'state',
       'leadType', 'leadCost', 'fpStatus', 'clarityFraudId', 'hasCF'],
      dtype='object')
```

The name of the columns

In [260…
```
loan.isna().sum()
```

Out[260]:
```
loanId                                256
anon_ssn                                0
payFrequency                         1273
apr                                  3922
applicationDate                         0
originated                              0
originatedDate                     531638
nPaidOff                               24
approved                                0
isFunded                                0
loanStatus                            391
loanAmount                           2250
originallyScheduledPaymentAmount        0
state                                 132
leadType                                0
leadCost                                0
fpStatus                           525959
clarityFraudId                     219989
hasCF                                   0
dtype: int64
```
Out[260]:
```
loanId                                256
anon_ssn                                0
payFrequency                         1273
apr                                  3922
applicationDate                         0
originated                              0
originatedDate                     531638
nPaidOff                               24
approved                                0
isFunded                                0
loanStatus                            391
loanAmount                           2250
originallyScheduledPaymentAmount        0
state                                 132
leadType                                0
leadCost                                0
fpStatus                           525959
clarityFraudId                     219989
hasCF                                   0
dtype: int64
```

1. numbers of null values in the data set
2. we can observe that "**fpStatus**" column most of the data are missing, the column **clarityFraudId** 219989 number of rows data are missing. so I will saperate the tow column and I will analysis saperately.
3. **originatedDate** this column data are missing most of the data. I will drop the column.

4. Othere missing data rows i will drop because in the the dataset have large numner of rows with compare to missing rows for analysis.

In [261…
```
loan[loan.duplicated()==True]
```

Out[261]:

loanId	anon_ssn	payFrequency	apr	applicationDate	originated	originatedDate	nPaidOff	ap

Out[261]:

loanId	anon_ssn	payFrequency	apr	applicationDate	originated	originatedDate	nPaidOff	ap

No duplicate values are available

In [262…
```
loan.describe()
```

Out[262]:

	apr	nPaidOff	isFunded	loanAmount	originallyScheduledPaymentAm
count	573760.000000	577658.000000	577682.000000	575432.000000	577682.0(
mean	553.080972	0.037887	0.067480	514.245084	1428.8(
std	110.046159	0.333366	0.250852	320.939929	925.0(
min	0.000000	0.000000	0.000000	0.000000	-816.7
25%	490.000000	0.000000	0.000000	350.000000	1023.6
50%	590.000000	0.000000	0.000000	500.000000	1245.2
75%	601.000000	0.000000	0.000000	500.000000	1615.6(
max	705.590000	21.000000	1.000000	5000.000000	19963.6

Out[262]:

	apr	nPaidOff	isFunded	loanAmount	originallyScheduledPaymentAm
count	573760.000000	577658.000000	577682.000000	575432.000000	577682.0(
mean	553.080972	0.037887	0.067480	514.245084	1428.8(
std	110.046159	0.333366	0.250852	320.939929	925.0(
min	0.000000	0.000000	0.000000	0.000000	-816.7
25%	490.000000	0.000000	0.000000	350.000000	1023.6
50%	590.000000	0.000000	0.000000	500.000000	1245.2
75%	601.000000	0.000000	0.000000	500.000000	1615.6(
max	705.590000	21.000000	1.000000	5000.000000	19963.6

In the data 25 parcentile of datapoints apr is 490 and 75 parcentile of datapoints apr is 601 so 0 apr may be outlayer loanAmount min are 0 those datapoints are outlayer

In [263…
```
loan.dtypes
```

Out[263]:
```
loanId                                object
anon_ssn                              object
payFrequency                          object
apr                                  float64
applicationDate                       object
originated                              bool
originatedDate                        object
nPaidOff                             float64
approved                                bool
isFunded                               int64
loanStatus                            object
loanAmount                           float64
originallyScheduledPaymentAmount     float64
state                                 object
leadType                              object
leadCost                               int64
fpStatus                              object
clarityFraudId                        object
hasCF                                  int64
dtype: object
```

Out[263]:
```
loanId                                object
anon_ssn                              object
payFrequency                          object
apr                                  float64
applicationDate                       object
originated                              bool
originatedDate                        object
nPaidOff                             float64
approved                                bool
isFunded                               int64
loanStatus                            object
loanAmount                           float64
originallyScheduledPaymentAmount     float64
state                                 object
leadType                              object
leadCost                               int64
fpStatus                              object
clarityFraudId                        object
hasCF                                  int64
dtype: object
```

In the dataset we can see the data type and plan to saperate the data, numerical data and catagorical data.

In [264…
```
loan[loan["loanAmount"]==0].index
```

Out[264]:
```
Index([  2914,   4043,   4507,   7027,   8287,  14577,  16178,  16950,  17795,
        20373,
       ...
       560635, 560866, 560874, 564333, 564590, 564969, 566404, 568240, 571420,
       574259],
      dtype='int64', length=417)
```

Out[264]:
```
Index([  2914,   4043,   4507,   7027,   8287,  14577,  16178,  16950,  17795,
        20373,
       ...
       560635, 560866, 560874, 564333, 564590, 564969, 566404, 568240, 571420,
       574259],
      dtype='int64', length=417)
```

In the dataset **loanAmount** should not be 0. Those rows are outlyer. we can drop the rows.

In [265…
```
loan.shape
```

Out[265]: (577682, 19)

Out[265]: (577682, 19)

Before removeing the rows, number of rows avalable in the dataset is 577682

outlyer removeing

In [266…
```
for i in loan[loan["loanAmount"]==0].index:
    loan.drop(i,axis=0,inplace=True)
```

Removing the missing value rows of the column **loanAmount**

In [267…
```
loan.shape # after removing the row loan dataset updated rows.
```

Out[267]: (577265, 19)

Out[267]: (577265, 19)

In [268…
```
loan.drop("originatedDate",axis=1,inplace=True) ## drop the column originatedDate
```

The column **applicationDate** the year and month saperate from the column and create a new columns for year and month to analysis month with year people are more active and drop the column form the dataset.

In [269…
```
loan["Application_Year"]=loan["applicationDate"].str.split("-").str[0]
```

In [270…
```
loan["Application_Month"]=loan["applicationDate"].str.split("-").str[1]
```

In [271…
```
loan.drop("applicationDate",axis=1,inplace=True)
```

In [272…
```
loan.dropna(subset=["loanStatus"], axis=0, inplace=True)
```

In [273…
```
loan.dropna(subset=["loanAmount"], axis=0, inplace=True)
```

In [274…
```
loan.dropna(subset=["payFrequency","apr"], axis=0, inplace=True)
```

In [275…
```
loan.dropna(subset=["nPaidOff"], axis=0, inplace=True)
```

In [276…
```
loan.dropna(subset=["state"], axis=0, inplace=True)
```

In [277…
```
loan.drop(["fpStatus","clarityFraudId"],axis=1,inplace=True)
```

loanStatus, loanAmount, payFrequency, nPaidOff, state, fpStatus, clarityFraudId has been drop na valuses rows because few null values with respect to the dataset total rows which will not so much impact on the full dataset.

In [278…
```
loan.isna().sum()
```

Out[278]:
```
loanId                              0
anon_ssn                            0
payFrequency                        0
apr                                 0
originated                          0
nPaidOff                            0
approved                            0
isFunded                            0
loanStatus                          0
loanAmount                          0
originallyScheduledPaymentAmount    0
state                               0
leadType                            0
leadCost                            0
hasCF                               0
Application_Year                    0
Application_Month                   0
dtype: int64
```

Out[278]:
```
loanId                              0
anon_ssn                            0
payFrequency                        0
apr                                 0
originated                          0
nPaidOff                            0
approved                            0
isFunded                            0
loanStatus                          0
loanAmount                          0
originallyScheduledPaymentAmount    0
state                               0
leadType                            0
leadCost                            0
hasCF                               0
Application_Year                    0
Application_Month                   0
dtype: int64
```

In the datast no null values

In [279…
```
catogorical_column=loan.columns[loan.dtypes=="object"]
```

The catogorical_columns name has been saperated.

In [280…
```
catogorical_column  # catogorical column name
```

Out[280]:
```
Index(['loanId', 'anon_ssn', 'payFrequency', 'loanStatus', 'state', 'leadType',
       'Application_Year', 'Application_Month'],
      dtype='object')
```

Out[280]:
```
Index(['loanId', 'anon_ssn', 'payFrequency', 'loanStatus', 'state', 'leadType',
       'Application_Year', 'Application_Month'],
      dtype='object')
```

In [281…
```
numerical_column=loan.columns[loan.dtypes!="object"]  # numerical column name
```

In [282…
```
# Step 1: Get the value counts of 'anon_ssn'
ssn_counts = loan["anon_ssn"].value_counts()

# Step 2: Filter for 'anon_ssn' values that occur more than once
ssn_more_than_once = ssn_counts[ssn_counts > 1].index

# Step 3: Filter the original DataFrame for these 'anon_ssn' values
result = loan[loan["anon_ssn"].isin(ssn_more_than_once)]
```

```
# If you only want the 'anon_ssn' column
anon_ssn_result = result["anon_ssn"]
```

In [283…
```
anon_ssn_result
```

Out[283]:
```
3         9be6f443bb97db7e95fa0c281d34da91
4         63b5494f60b5c19c827c7b068443752c
5         b5541f49472fa0fce8e473306768f7fb
7         02596517e7633c7e87e6b333a0fb1bbe
9         47bf79119075e41ef65510f2900c8e4a
                       ...
577666    ad6970cdb83f6f5fc0154ac8e2d6746a
577675    43ff47d188fa9350e43f18094254b4d1
577676    3506893b63baae416cf211238a391acc
577680    c3b35307cb36116bf59574f9138d3dad
577681    dc0a43b16c037ee5d0142daebb5db83a
Name: anon_ssn, Length: 195522, dtype: object
```

Out[283]:
```
3         9be6f443bb97db7e95fa0c281d34da91
4         63b5494f60b5c19c827c7b068443752c
5         b5541f49472fa0fce8e473306768f7fb
7         02596517e7633c7e87e6b333a0fb1bbe
9         47bf79119075e41ef65510f2900c8e4a
                       ...
577666    ad6970cdb83f6f5fc0154ac8e2d6746a
577675    43ff47d188fa9350e43f18094254b4d1
577676    3506893b63baae416cf211238a391acc
577680    c3b35307cb36116bf59574f9138d3dad
577681    dc0a43b16c037ee5d0142daebb5db83a
Name: anon_ssn, Length: 195522, dtype: object
```

This is the result where 195522 individual clints whose loan belongs to a previous customer.

In [284…
```
anon_ssn_result.shape
```

Out[284]:
```
(195522,)
```

Out[284]:
```
(195522,)
```

In [285…
```
numerical_column
```

Out[285]:
```
Index(['apr', 'originated', 'nPaidOff', 'approved', 'isFunded', 'loanAmount',
       'originallyScheduledPaymentAmount', 'leadCost', 'hasCF'],
      dtype='object')
```

Out[285]:
```
Index(['apr', 'originated', 'nPaidOff', 'approved', 'isFunded', 'loanAmount',
       'originallyScheduledPaymentAmount', 'leadCost', 'hasCF'],
      dtype='object')
```

In [286…
```
loan[numerical_column]
```

Out[286]:

	apr	originated	nPaidOff	approved	isFunded	loanAmount	originallyScheduledPaymer
0	360.0	False	0.0	False	0	500.0	
1	199.0	True	0.0	True	1	3000.0	
2	590.0	False	0.0	False	0	400.0	
3	360.0	False	0.0	False	0	500.0	
4	590.0	False	0.0	False	0	350.0	
...	...	...	...	...	...	...	
577677	590.0	False	0.0	False	0	400.0	
577678	490.0	False	0.0	False	0	1000.0	
577679	590.0	False	0.0	False	0	300.0	
577680	550.0	False	0.0	False	0	300.0	
577681	590.0	False	0.0	False	0	400.0	

571867 rows × 9 columns

Out[286]:

	apr	originated	nPaidOff	approved	isFunded	loanAmount	originallyScheduledPaymer
0	360.0	False	0.0	False	0	500.0	
1	199.0	True	0.0	True	1	3000.0	
2	590.0	False	0.0	False	0	400.0	
3	360.0	False	0.0	False	0	500.0	
4	590.0	False	0.0	False	0	350.0	
...	...	...	...	...	...	...	
577677	590.0	False	0.0	False	0	400.0	
577678	490.0	False	0.0	False	0	1000.0	
577679	590.0	False	0.0	False	0	300.0	
577680	550.0	False	0.0	False	0	300.0	
577681	590.0	False	0.0	False	0	400.0	

571867 rows × 9 columns

In [287…

```
loan[catogorical_column]
```

Out[287]:

	loanId	anon_ssn	payFrequency	loanStatus	state	leadT
0	LL-I-07399092	beff4989be82aab4a5b47679216942fd	B	Withdrawn Application	IL	bvMandat
1	LL-I-06644937	464f5d9ae4fa09ece4048d949191865c	B	Paid Off Loan	CA	prescr
2	LL-I-10707532	3c174ae9e2505a5f9ddbff9843281845	B	Withdrawn Application	MO	bvMandat
3	LL-I-02272596	9be6f443bb97db7e95fa0c281d34da91	B	Withdrawn Application	IL	bvMandat
4	LL-I-09542882	63b5494f60b5c19c827c7b068443752c	B	Rejected	NV	bvMandat
...	...	...	...	...	...	
577677	LL-I-12122269	801262d04720d32040612759857f4147	B	Withdrawn Application	NV	bvMandat
577678	LL-I-16183462	e37750de9d99a67e0fa96a51e86fdf5b	S	Withdrawn Application	MO	I
577679	LL-I-06962710	d7e55e85266208ac4c353f42ebcde5ca	B	Withdrawn Application	IN	bvMandat
577680	LL-I-01253468	c3b35307cb36116bf59574f9138d3dad	B	Withdrawn Application	OH	orga
577681	LL-I-04733921	dc0a43b16c037ee5d0142daebb5db83a	I	Rejected	OH	bvMandat

571867 rows × 8 columns

Out[287]:

	loanId	anon_ssn	payFrequency	loanStatus	state	leadT
0	LL-I-07399092	beff4989be82aab4a5b47679216942fd	B	Withdrawn Application	IL	bvMandat
1	LL-I-06644937	464f5d9ae4fa09ece4048d949191865c	B	Paid Off Loan	CA	prescr
2	LL-I-10707532	3c174ae9e2505a5f9ddbff9843281845	B	Withdrawn Application	MO	bvMandat
3	LL-I-02272596	9be6f443bb97db7e95fa0c281d34da91	B	Withdrawn Application	IL	bvMandat
4	LL-I-09542882	63b5494f60b5c19c827c7b068443752c	B	Rejected	NV	bvMandat
...	...	...	...	...	...	
577677	LL-I-12122269	801262d04720d32040612759857f4147	B	Withdrawn Application	NV	bvMandat
577678	LL-I-16183462	e37750de9d99a67e0fa96a51e86fdf5b	S	Withdrawn Application	MO	I
577679	LL-I-06962710	d7e55e85266208ac4c353f42ebcde5ca	B	Withdrawn Application	IN	bvMandat
577680	LL-I-01253468	c3b35307cb36116bf59574f9138d3dad	B	Withdrawn Application	OH	orga
577681	LL-I-04733921	dc0a43b16c037ee5d0142daebb5db83a	I	Rejected	OH	bvMandat

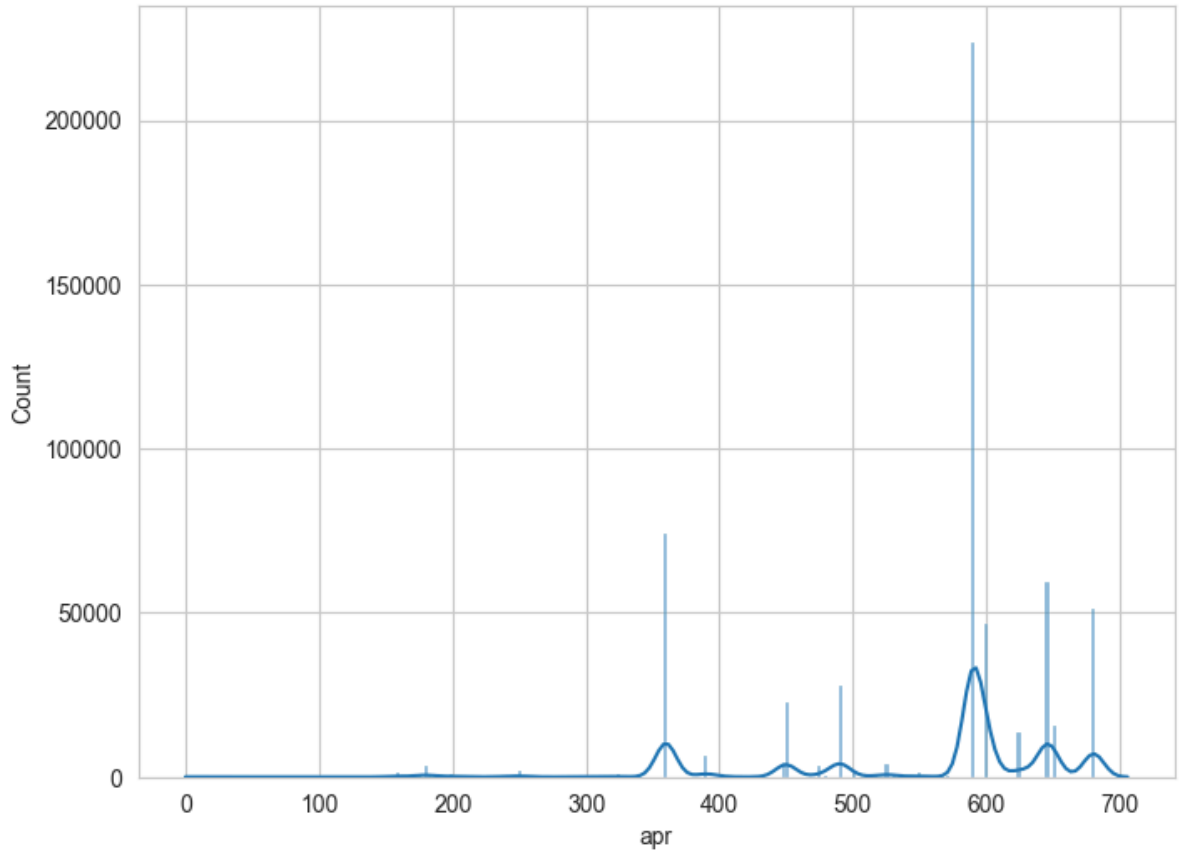
571867 rows × 8 columns

In [289…

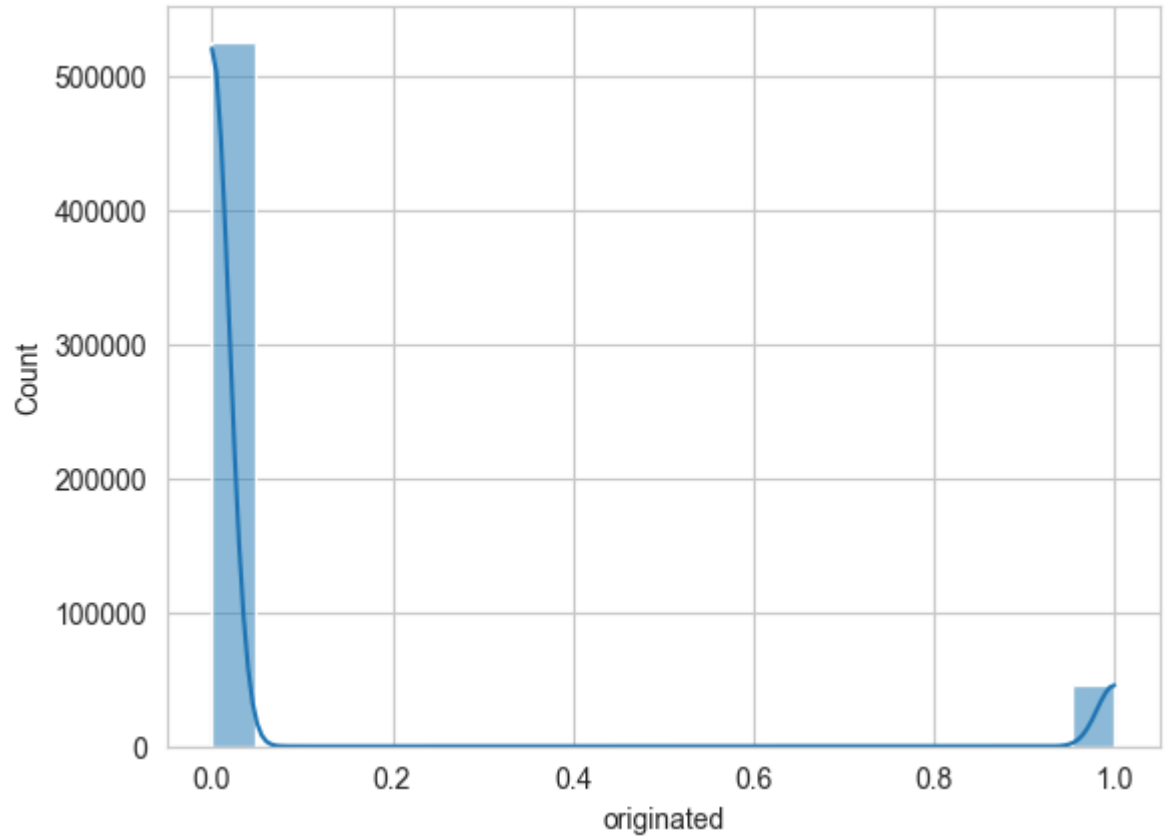
```
a=loan['state'].unique()
```

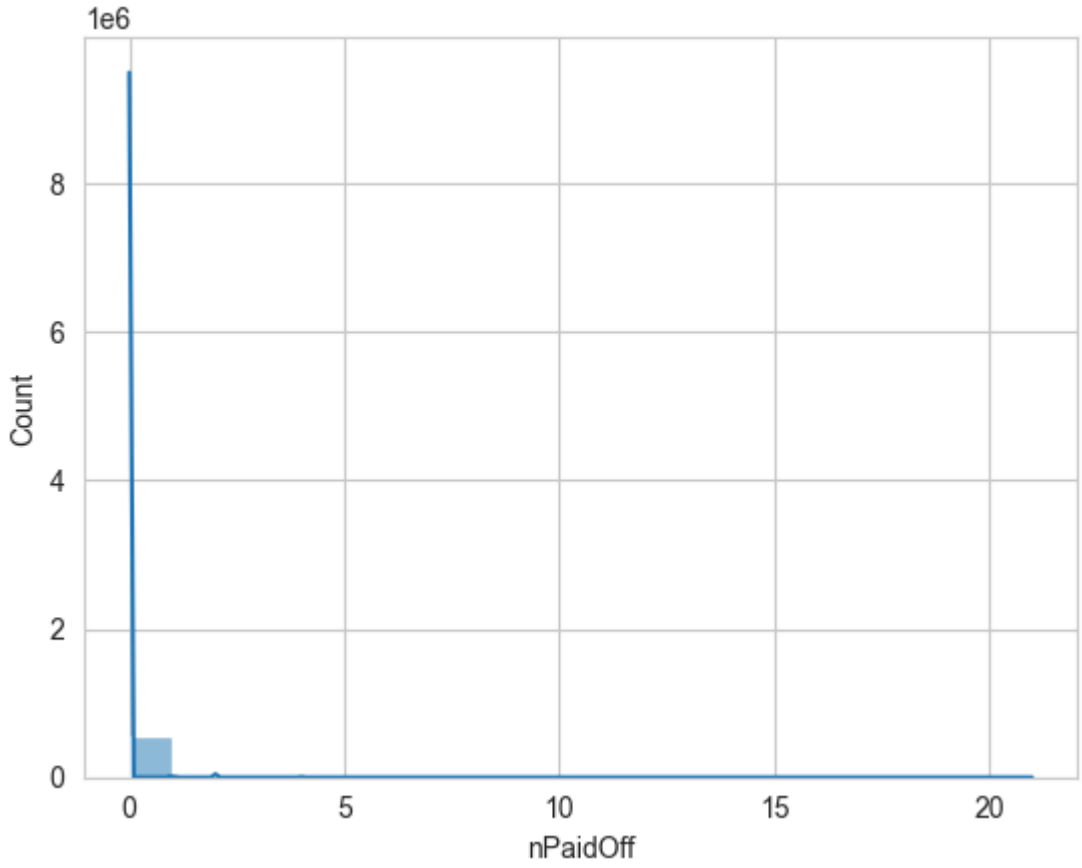
In [290…

```
import seaborn as sns
import matplotlib.pyplot as plt
plt.figure(figsize=(8,6))
x=0
for i in numerical_column:
    sns.histplot(data=loan,x=i,kde=True)
    print('\n')
    plt.show()
```

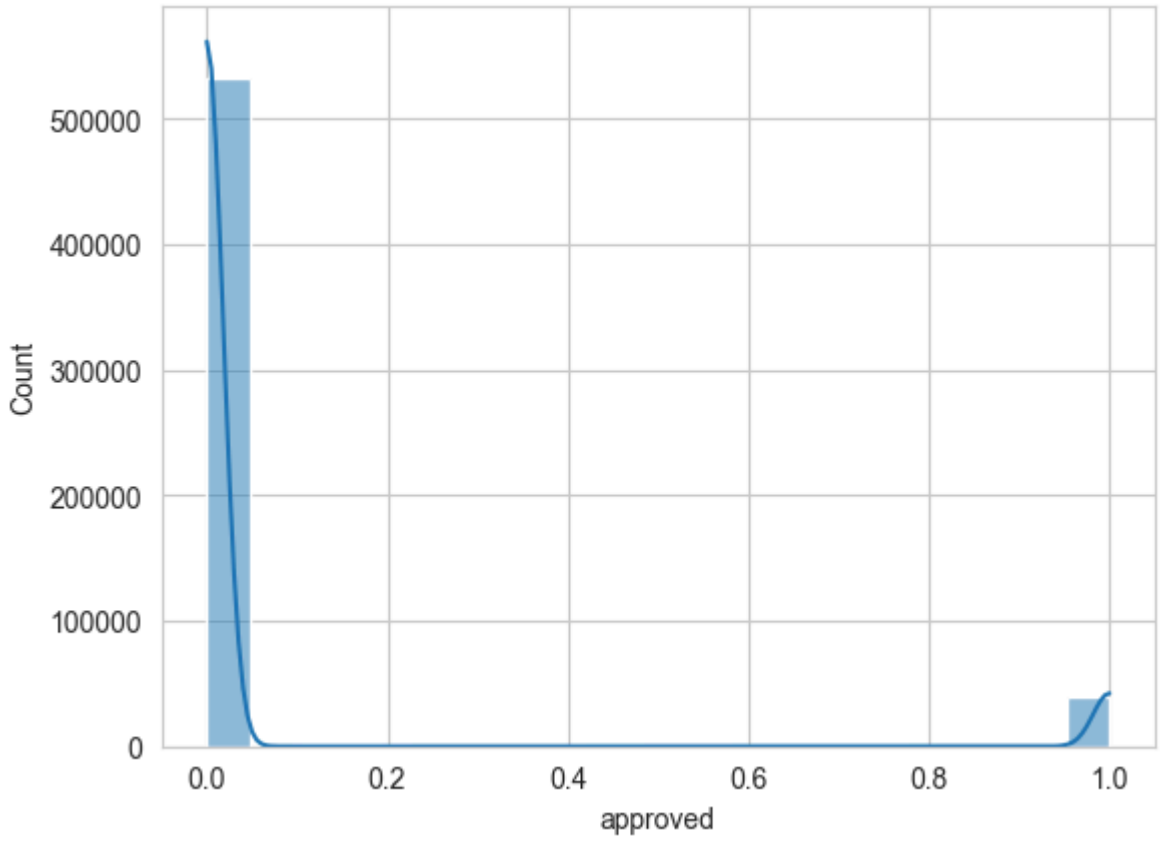


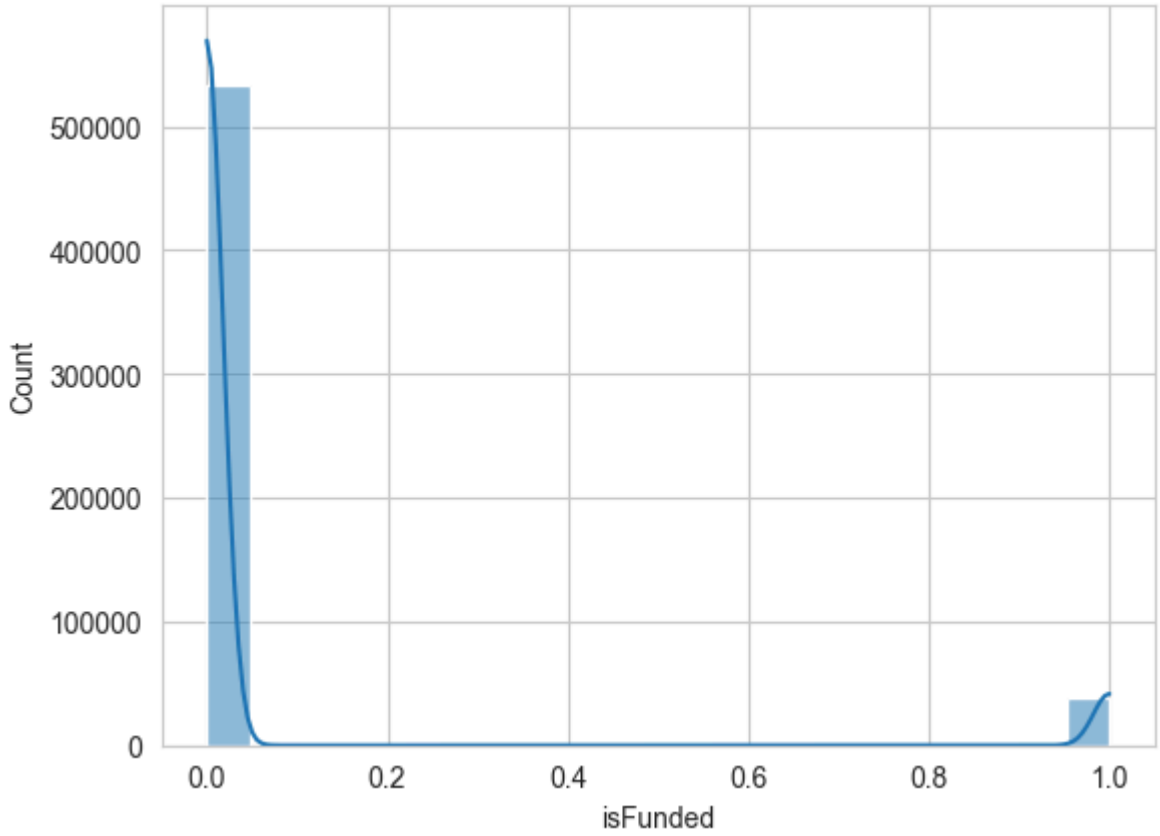

```
<__array_function__ internals>:180: RuntimeWarning: Converting input from bool to
<class 'numpy.uint8'> for compatibility.
```

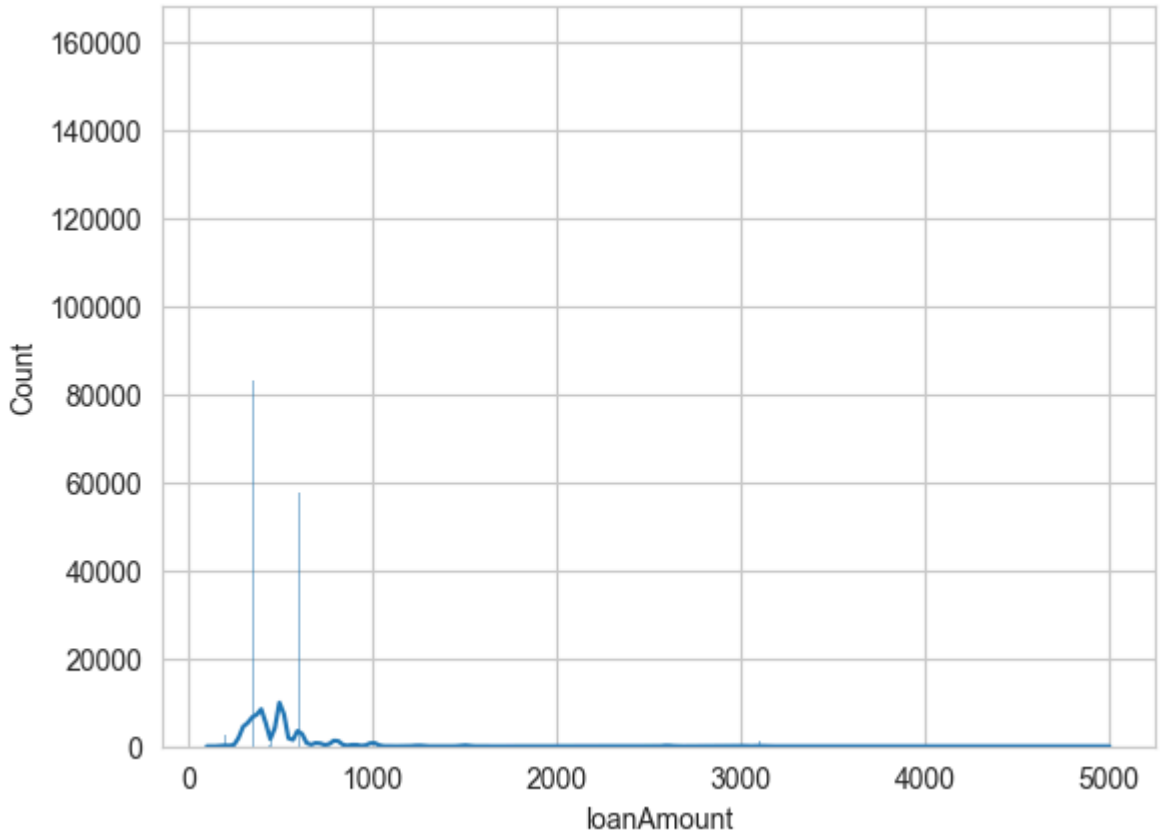


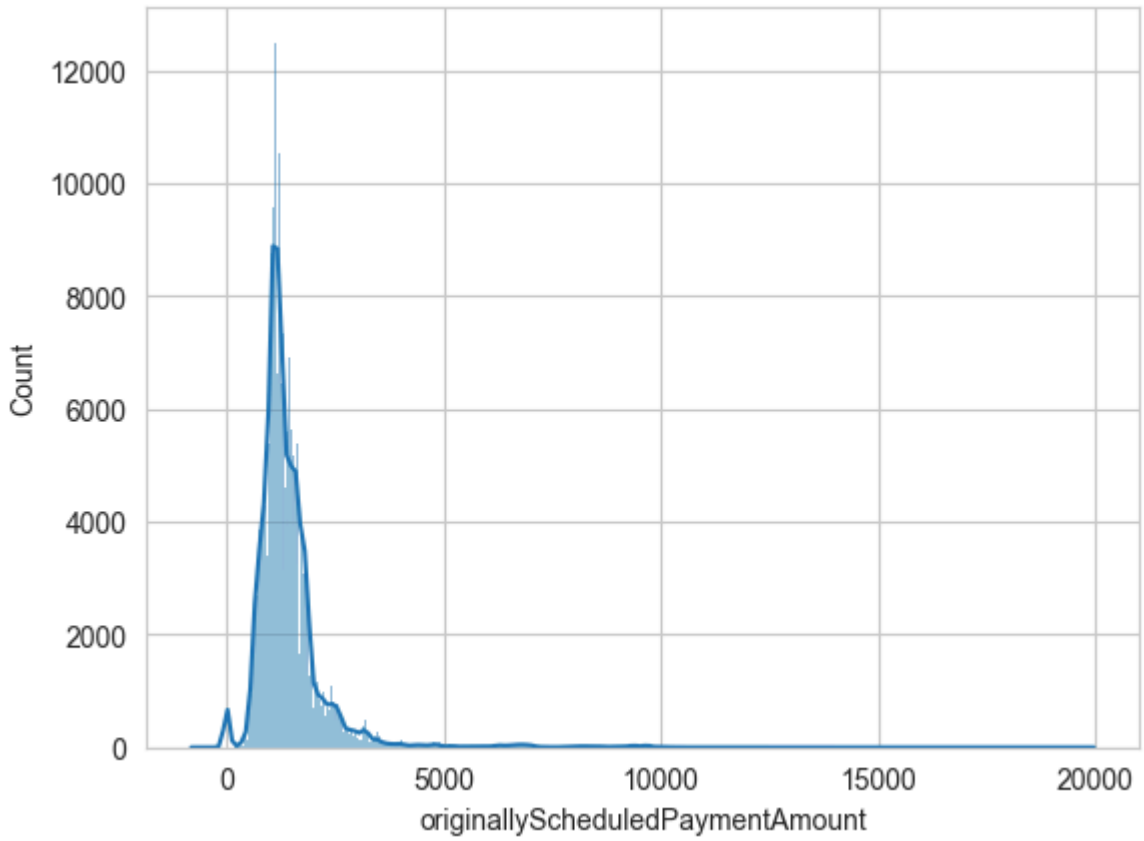


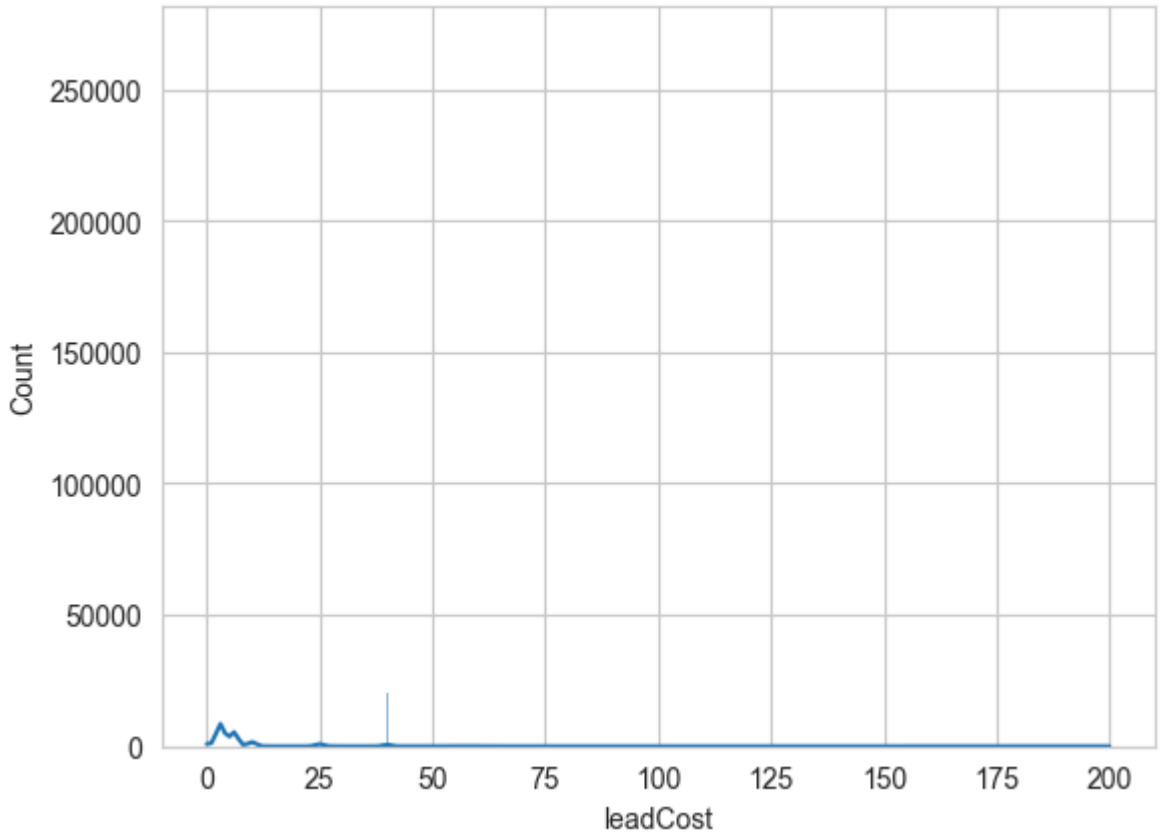

```
<__array_function__ internals>:180: RuntimeWarning: Converting input from bool to
<class 'numpy.uint8'> for compatibility.
```

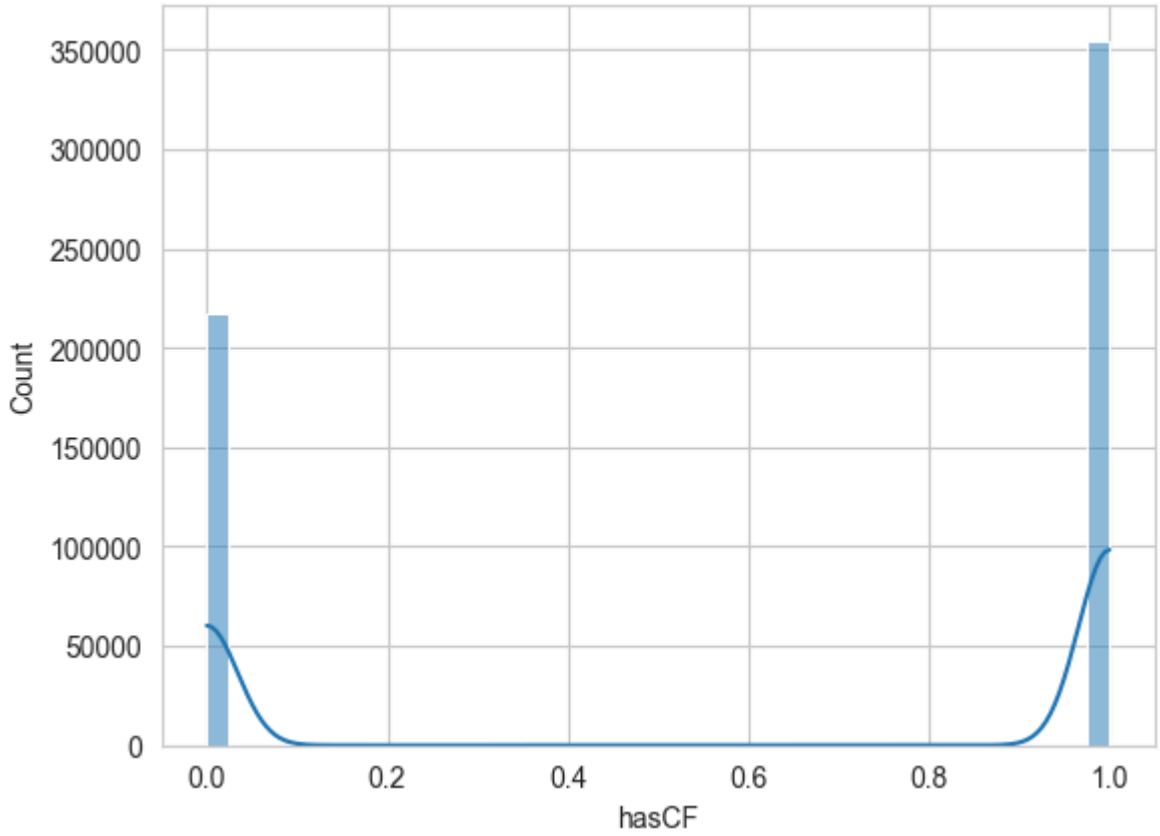


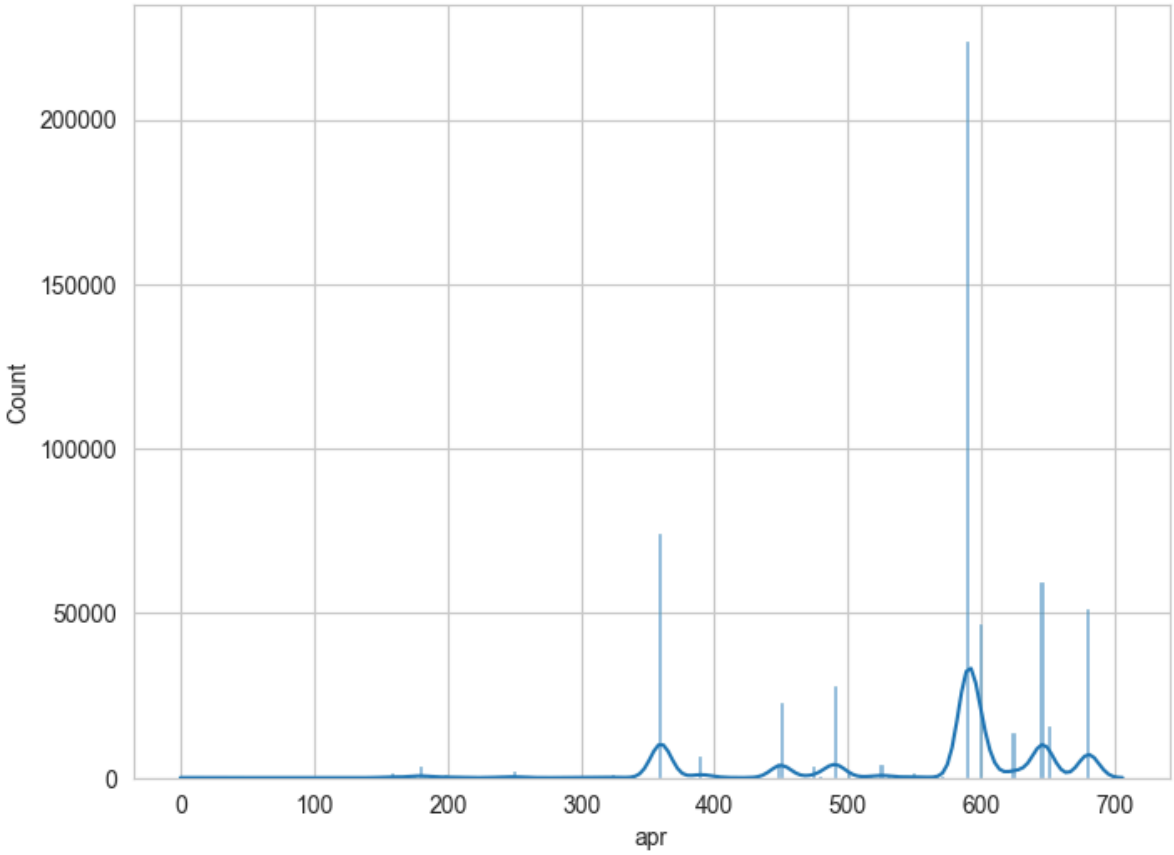

500000
400000
300000
200000
100000
0
Count
0.0
0.2
0.4
0.6
0.8
1.0
isFunded


160000
140000
120000
100000
80000
60000
40000
20000
0
Count
0
1000
2000
3000
4000
5000
loanAmount

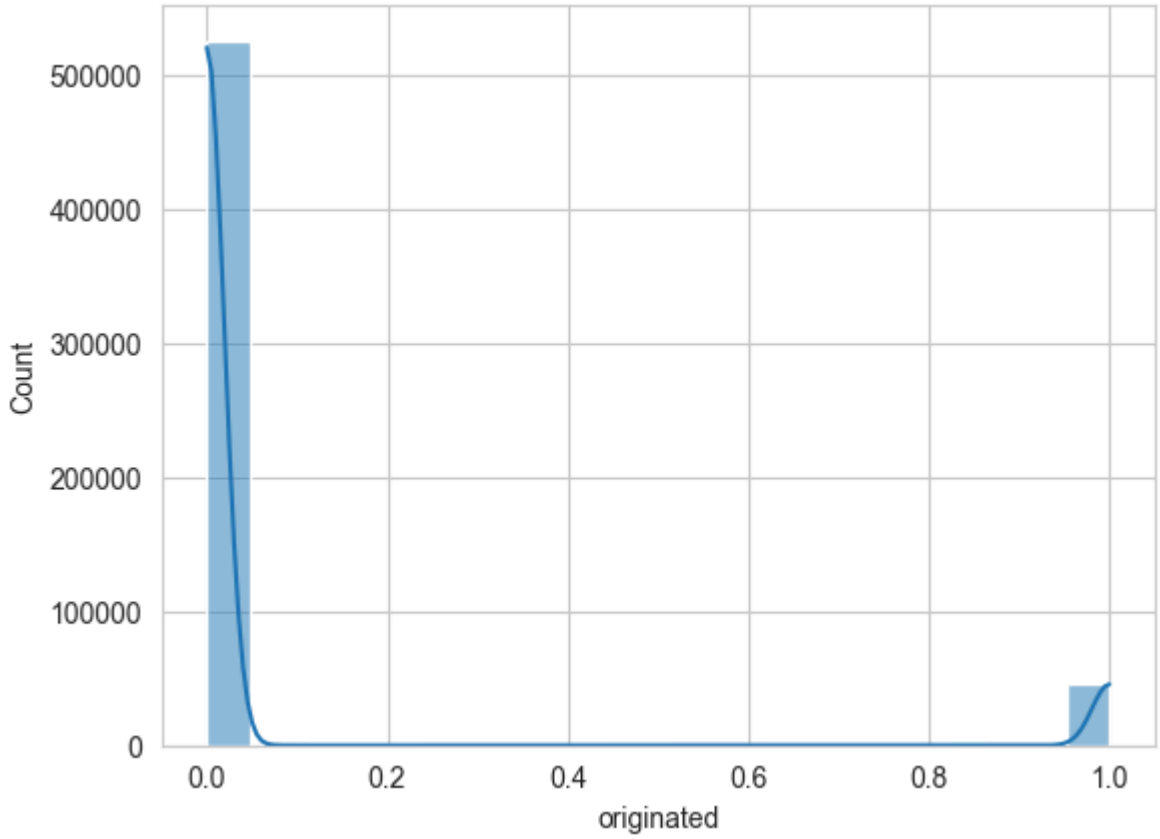

12000
10000
8000
6000
4000
2000
0
Count
0
5000
10000
15000
20000
originallyScheduledPaymentAmount

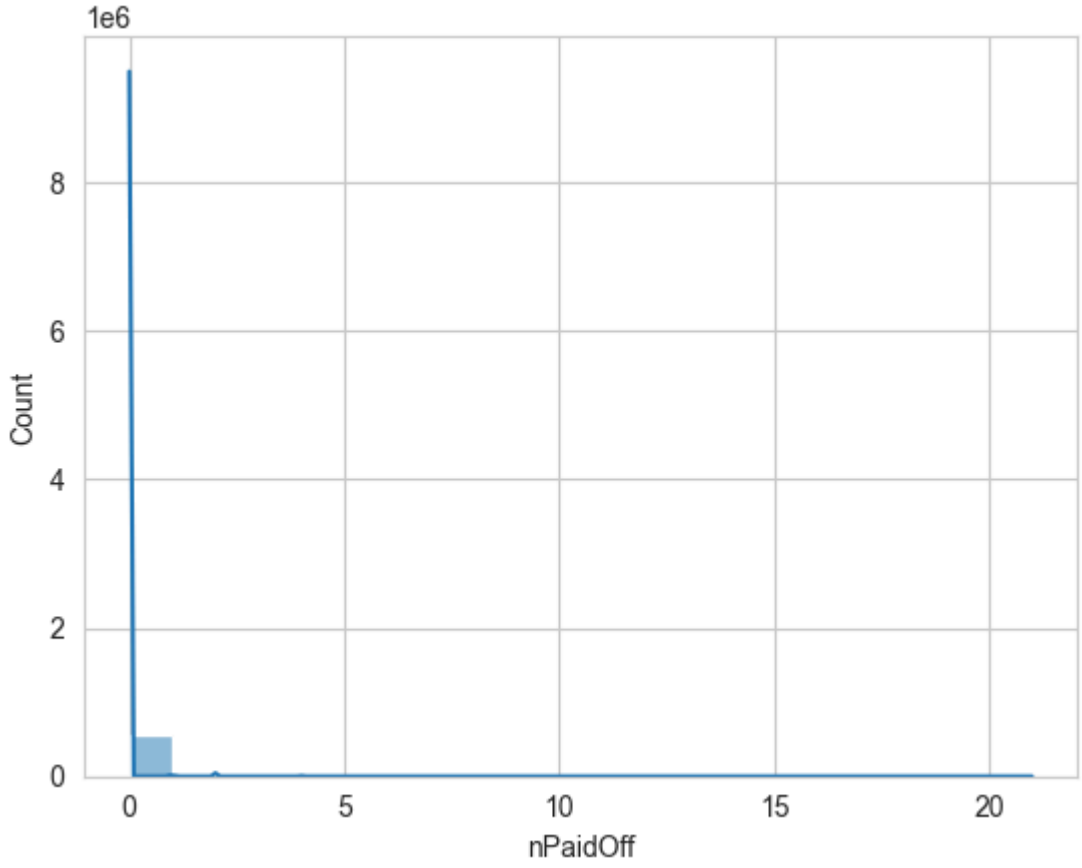

250000
200000
150000
100000
50000
0
Count
0
25
50
75
100
125
150
175
200
leadCost



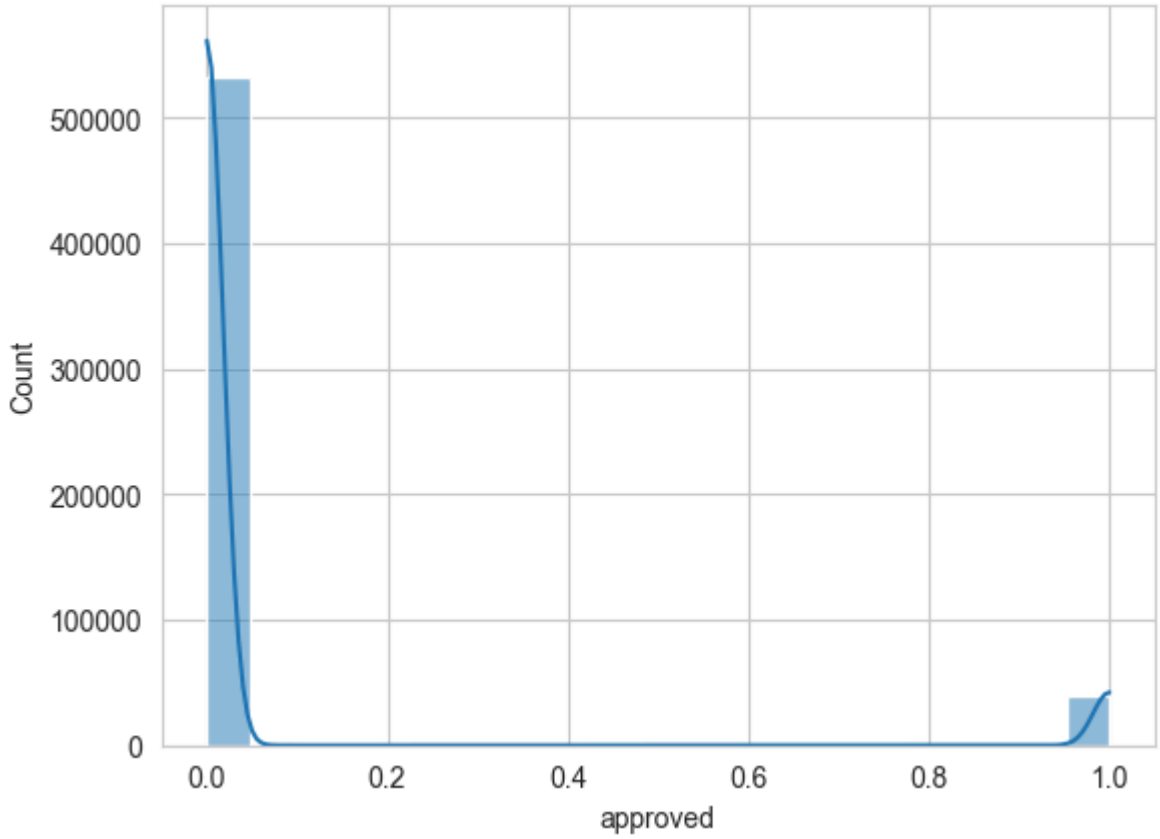


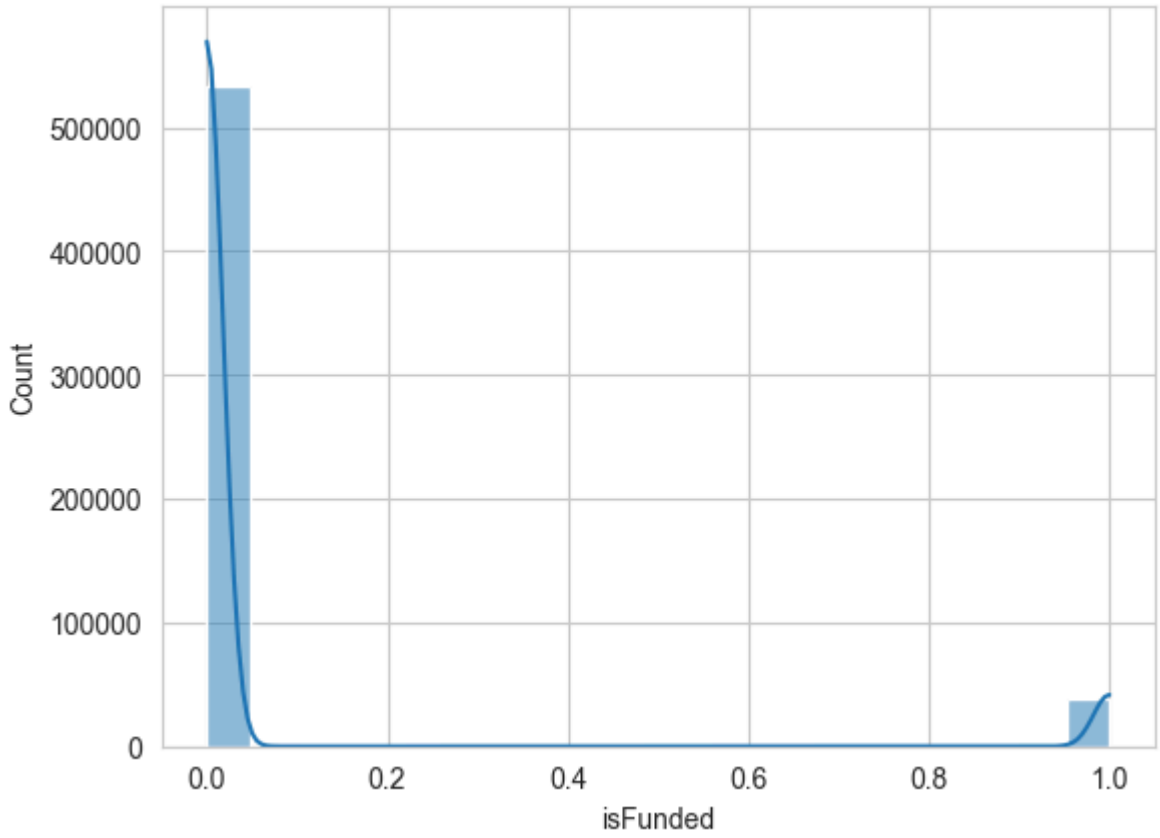

```
<__array_function__ internals>:180: RuntimeWarning: Converting input from bool to
<class 'numpy.uint8'> for compatibility.
```

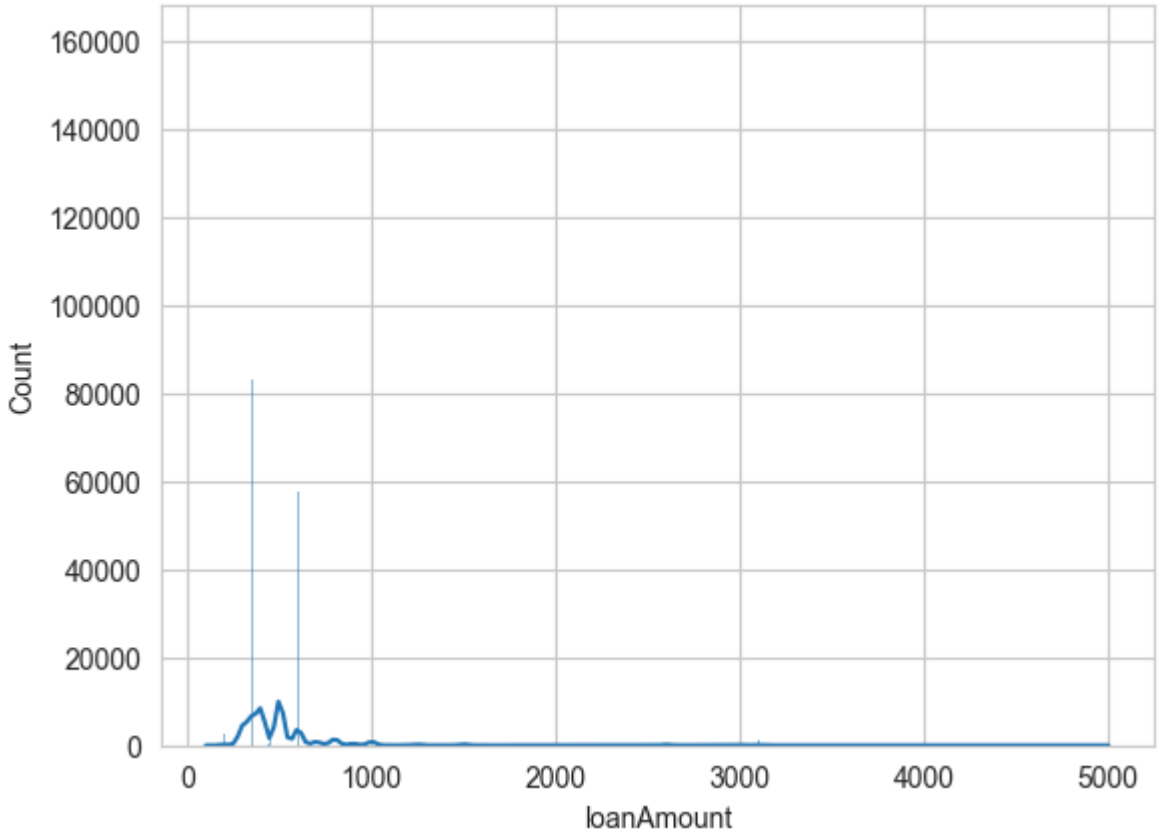


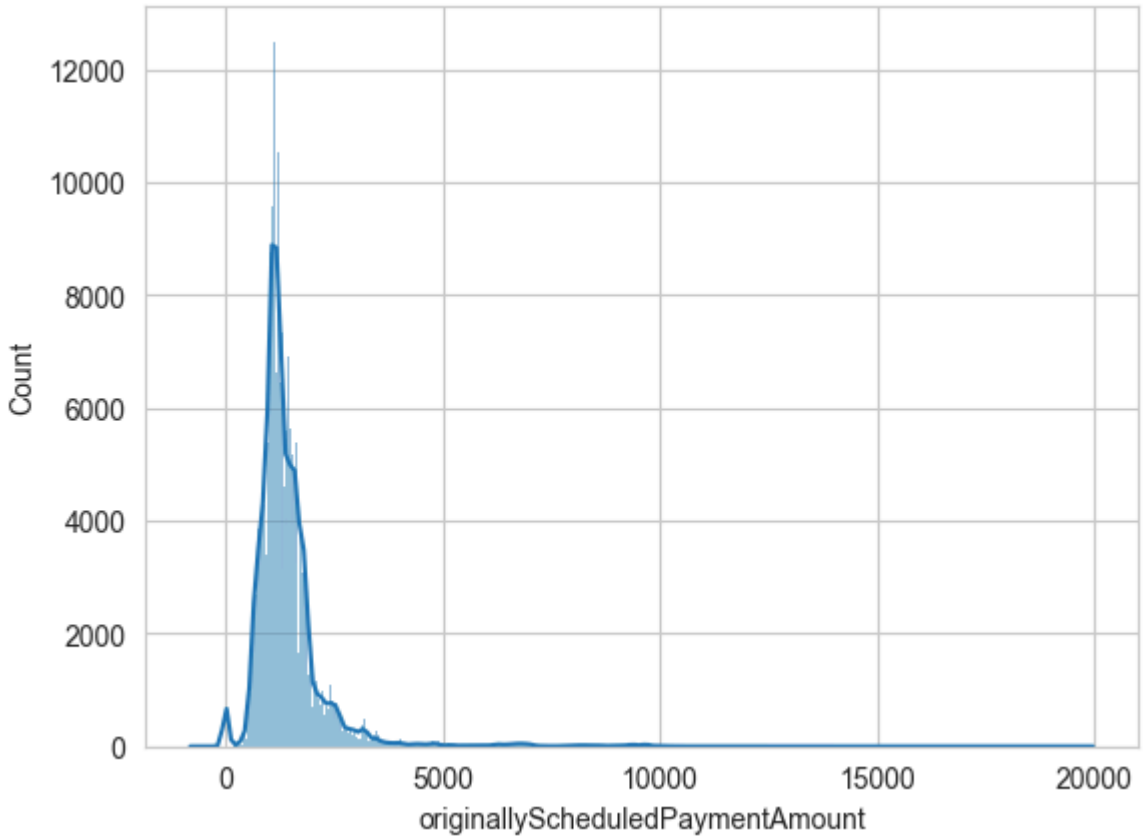


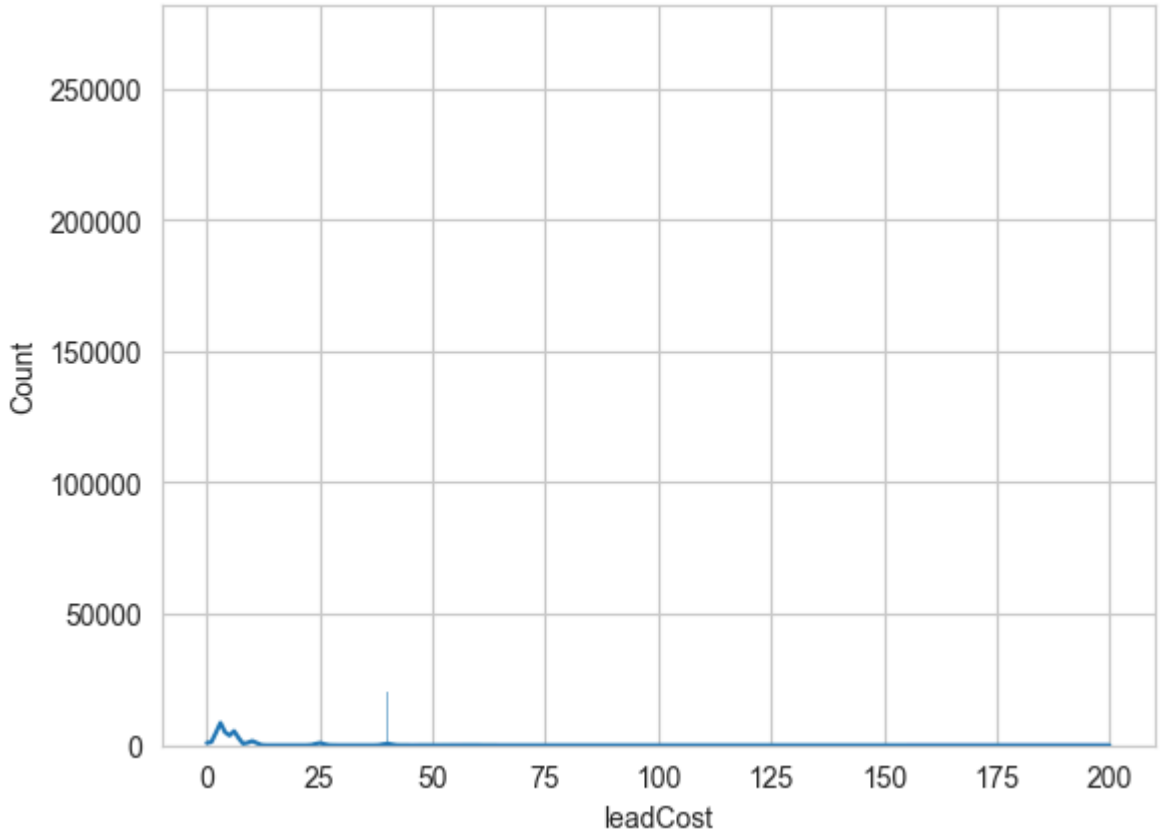

```
<__array_function__ internals>:180: RuntimeWarning: Converting input from bool to
<class 'numpy.uint8'> for compatibility.
```

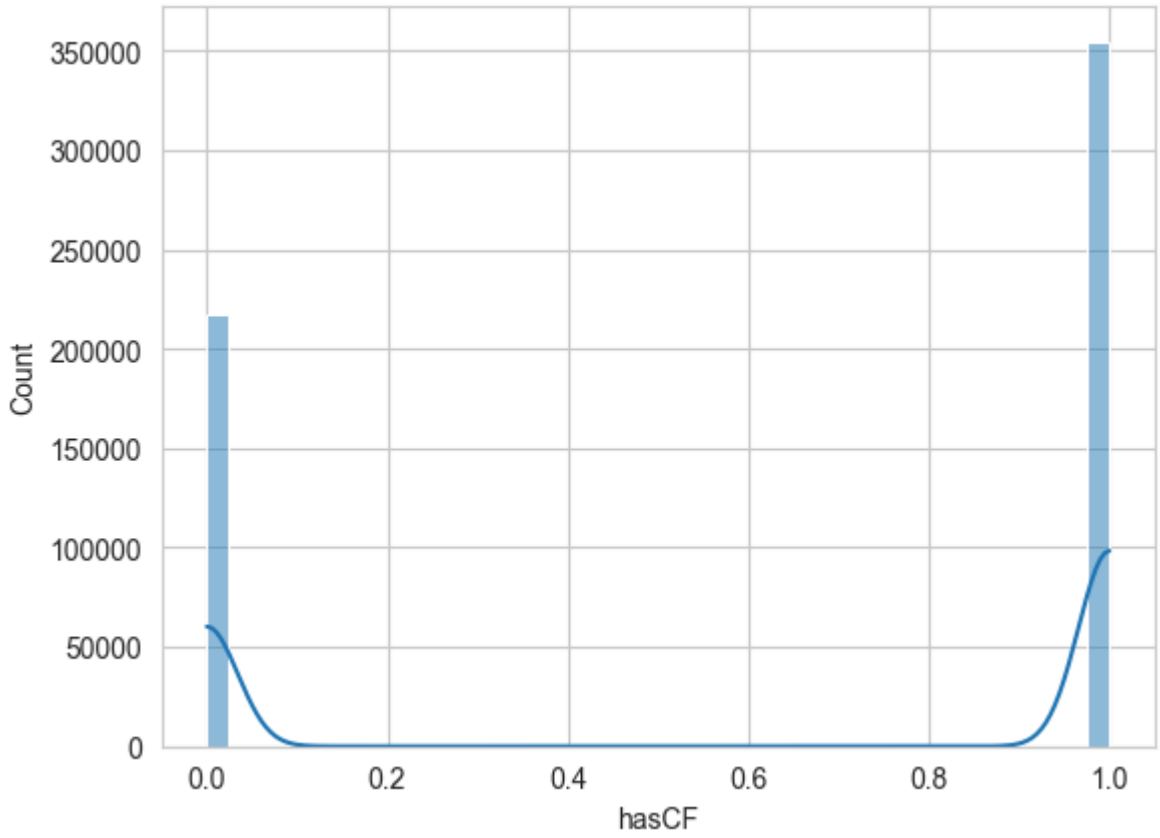

500000
400000
300000
200000
100000
0
Count
0.0
0.2
0.4
0.6
0.8
1.0
approved


500000
400000
300000
200000
100000
0
Count
0.0
0.2
0.4
0.6
0.8
1.0
isFunded


160000
140000
120000
100000
80000
60000
40000
20000
0
Count
0
1000
2000
3000
4000
5000
loanAmount


12000
10000
8000
6000
4000
2000
0
Count
0
5000
10000
15000
20000
originallyScheduledPaymentAmount





In [291…

```
loan.groupby(by=['state'])['loanAmount'].mean()
```

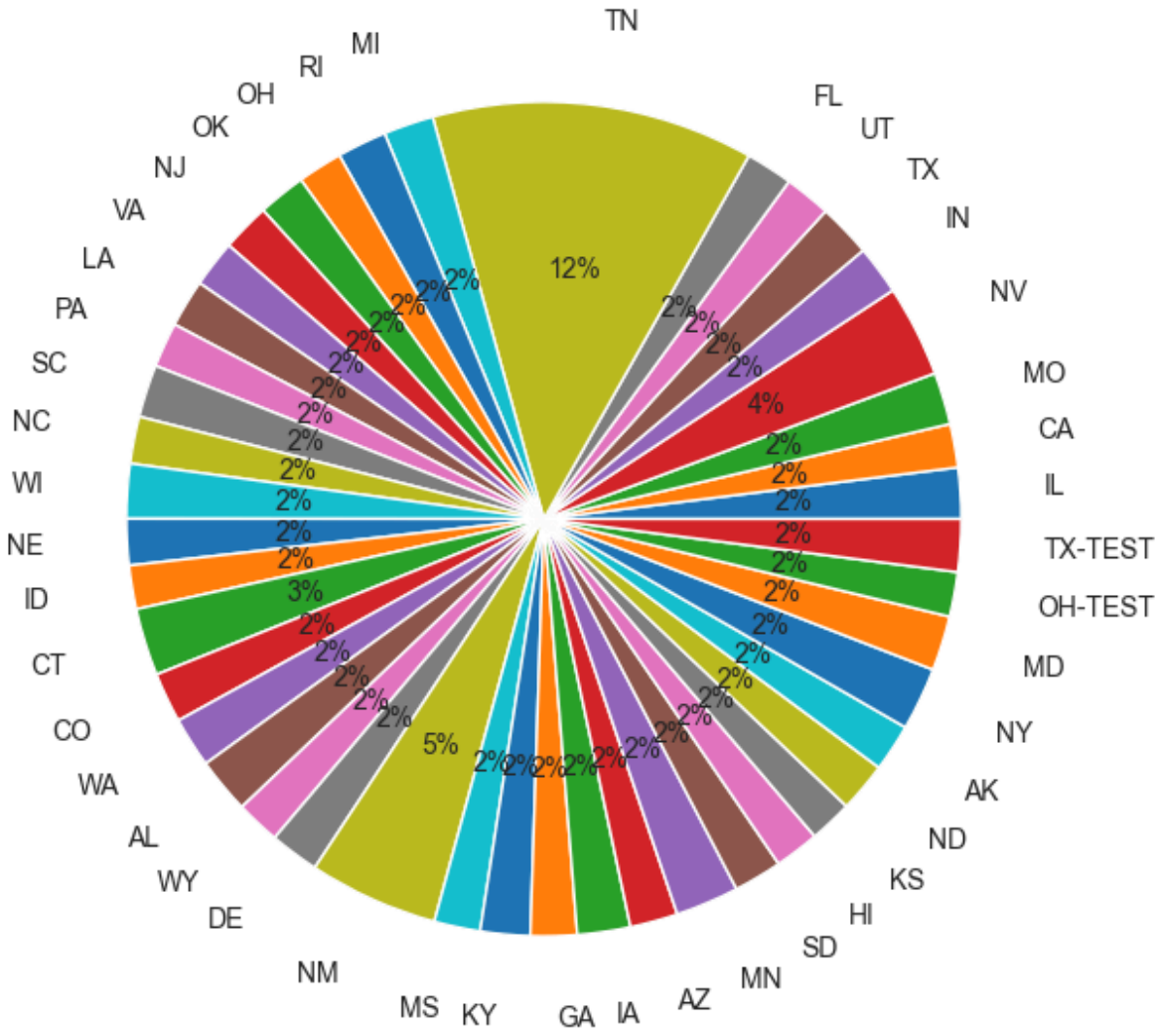
Out[291]:

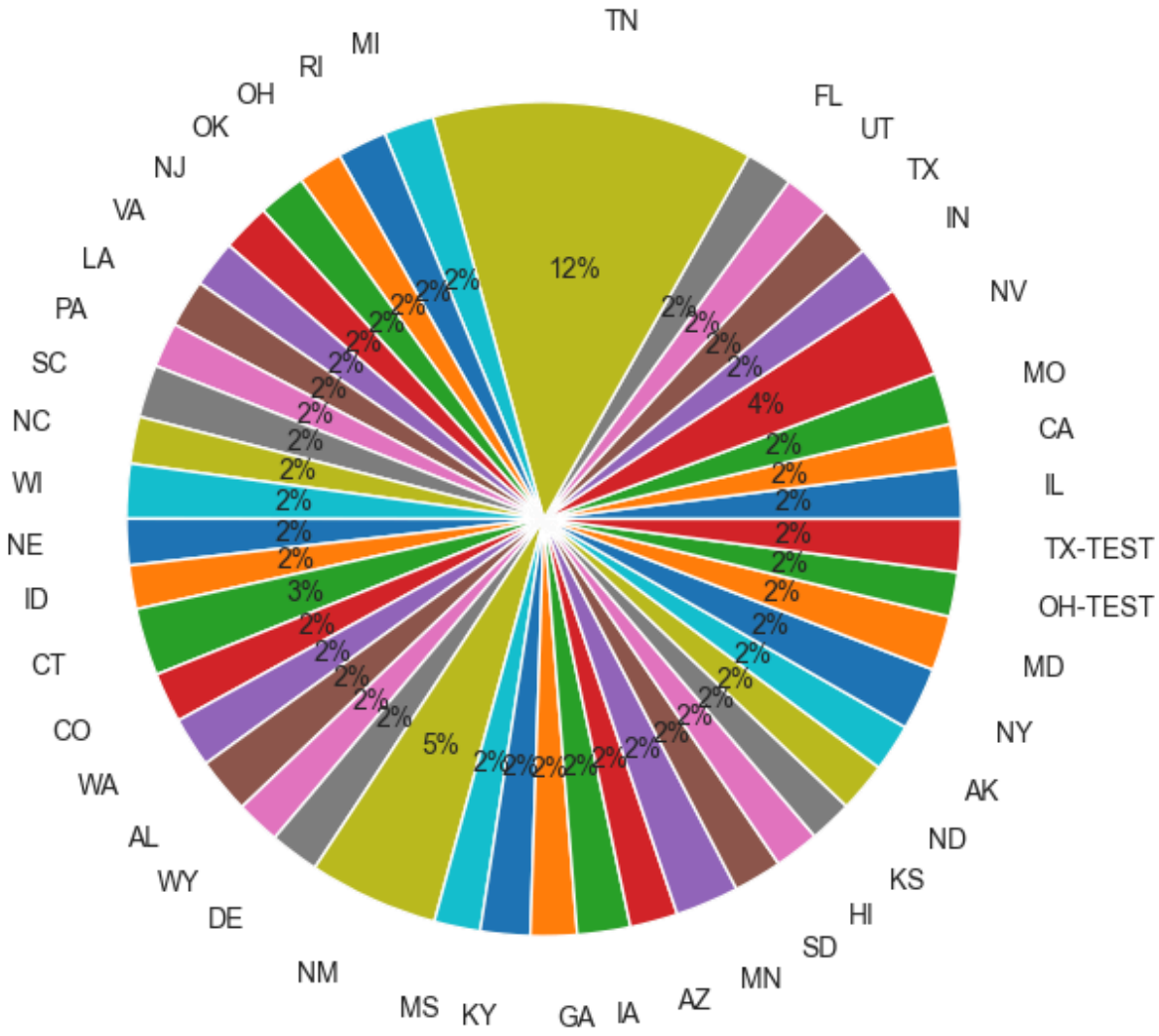
```
state
AK          512.804878
AL          438.151796
AZ          524.643585
CA          922.706079
CO          494.058856
CT          537.488356
DE          467.796128
FL          472.037134
GA         3237.669383
HI          503.282010
IA          498.786127
ID          451.588072
IL          480.973797
IN          486.710881
KS          472.892799
KY          478.496366
LA          451.356475
MD          529.000000
MI          473.040589
MN          549.569841
MO          464.498264
MS          439.160700
NC          678.105561
ND          504.756098
NE          505.717666
NJ          559.076691
NM          445.658915
NV          496.444299
NY         1304.000000
OH          463.834854
OH-TEST     500.000000
OK          465.687393
PA          532.993082
RI          480.685131
SC          638.610974
SD          485.899818
TN          449.952398
TX          433.566002
TX-TEST     500.000000
UT          476.852591
VA          634.820759
WA          551.123278
WI          448.076350
WY          527.939696
Name: loanAmount, dtype: float64
```

Out[291]:
```
state
AK          512.804878
AL          438.151796
AZ          524.643585
CA          922.706079
CO          494.058856
CT          537.488356
DE          467.796128
FL          472.037134
GA         3237.669383
HI          503.282010
IA          498.786127
ID          451.588072
IL          480.973797
IN          486.710881
KS          472.892799
KY          478.496366
LA          451.356475
MD          529.000000
MI          473.040589
MN          549.569841
MO          464.498264
MS          439.160700
NC          678.105561
ND          504.756098
NE          505.717666
NJ          559.076691
NM          445.658915
NV          496.444299
NY         1304.000000
OH          463.834854
OH-TEST     500.000000
OK          465.687393
PA          532.993082
RI          480.685131
SC          638.610974
SD          485.899818
TN          449.952398
TX          433.566002
TX-TEST     500.000000
UT          476.852591
VA          634.820759
WA          551.123278
WI          448.076350
WY          527.939696
Name: loanAmount, dtype: float64
```

The state is GA and NY, Person loan amount quite high person. More possibility is the state is rich state. So the state may property price may high. May people leaving cost high, May possable the state is developed state, Possibility the state is finincial hub on the country.

In [292…
```
plt.figure(figsize=(7,7))
plt.pie(x = loan.groupby(by=['state'])['loanAmount'].mean(),labels=list(a),autopct=
plt.show()
```


TN
FL
UT
TX
IN
NV
MO
CA
IL
TX-TEST
OH-TEST
MD
NY
AK
ND
KS
HI
SD
MN
AZ
IA
GA
KY
MS
NM
DE
WY
AL
WA
CO
CT
ID
NE
WI
NC
SC
PA
LA
VA
NJ
OK
OH
RI
MI
12%
5%
4%
3%
2%


TN
FL
UT
TX
IN
NV
MO
CA
IL
TX-TEST
OH-TEST
MD
NY
AK
ND
KS
HI
SD
MN
AZ
IA
GA
KY
MS
NM
DE
WY
AL
WA
CO
CT
ID
NE
WI
NC
SC
PA
LA
VA
NJ
OK
OH
RI
MI
12%
5%

In [293…

```
loan
```

Out[293]:

	loanId	anon_ssn	payFrequency	apr	originated	nPaidOff
0	LL-I-07399092	beff4989be82aab4a5b47679216942fd	B	360.0	False	0.0
1	LL-I-06644937	464f5d9ae4fa09ece4048d949191865c	B	199.0	True	0.0
2	LL-I-10707532	3c174ae9e2505a5f9ddbff9843281845	B	590.0	False	0.0
3	LL-I-02272596	9be6f443bb97db7e95fa0c281d34da91	B	360.0	False	0.0
4	LL-I-09542882	63b5494f60b5c19c827c7b068443752c	B	590.0	False	0.0
...	...	...	...	...	...	...
577677	LL-I-12122269	801262d04720d32040612759857f4147	B	590.0	False	0.0
577678	LL-I-16183462	e37750de9d99a67e0fa96a51e86fdf5b	S	490.0	False	0.0
577679	LL-I-06962710	d7e55e85266208ac4c353f42ebcde5ca	B	590.0	False	0.0
577680	LL-I-01253468	c3b35307cb36116bf59574f9138d3dad	B	550.0	False	0.0
577681	LL-I-04733921	dc0a43b16c037ee5d0142daebb5db83a	I	590.0	False	0.0

571867 rows × 17 columns

Out[293]:

	loanId	anon_ssn	payFrequency	apr	originated	nPaidOff
0	LL-I-07399092	beff4989be82aab4a5b47679216942fd	B	360.0	False	0.0
1	LL-I-06644937	464f5d9ae4fa09ece4048d949191865c	B	199.0	True	0.0
2	LL-I-10707532	3c174ae9e2505a5f9ddbff9843281845	B	590.0	False	0.0
3	LL-I-02272596	9be6f443bb97db7e95fa0c281d34da91	B	360.0	False	0.0
4	LL-I-09542882	63b5494f60b5c19c827c7b068443752c	B	590.0	False	0.0
...	...	...	...	...	...	...
577677	LL-I-12122269	801262d04720d32040612759857f4147	B	590.0	False	0.0
577678	LL-I-16183462	e37750de9d99a67e0fa96a51e86fdf5b	S	490.0	False	0.0
577679	LL-I-06962710	d7e55e85266208ac4c353f42ebcde5ca	B	590.0	False	0.0
577680	LL-I-01253468	c3b35307cb36116bf59574f9138d3dad	B	550.0	False	0.0
577681	LL-I-04733921	dc0a43b16c037ee5d0142daebb5db83a	I	590.0	False	0.0

571867 rows × 17 columns

In [294…

```
loan.dtypes
```

Out[294]:

```
loanId                             object
anon_ssn                           object
payFrequency                       object
apr                               float64
originated                           bool
nPaidOff                          float64
approved                             bool
isFunded                            int64
loanStatus                         object
loanAmount                        float64
originallyScheduledPaymentAmount  float64
state                              object
leadType                           object
leadCost                            int64
hasCF                               int64
Application_Year                   object
Application_Month                  object
dtype: object
```

Out[294]:
```
loanId                                object
anon_ssn                              object
payFrequency                          object
apr                                  float64
originated                              bool
nPaidOff                             float64
approved                                bool
isFunded                               int64
loanStatus                            object
loanAmount                           float64
originallyScheduledPaymentAmount     float64
state                                 object
leadType                              object
leadCost                               int64
hasCF                                  int64
Application_Year                      object
Application_Month                     object
dtype: object
```

In [295…
```
loan[loan.isna()==True].columns
```

Out[295]:
```
Index(['loanId', 'anon_ssn', 'payFrequency', 'apr', 'originated', 'nPaidOff',
       'approved', 'isFunded', 'loanStatus', 'loanAmount',
       'originallyScheduledPaymentAmount', 'state', 'leadType', 'leadCost',
       'hasCF', 'Application_Year', 'Application_Month'],
      dtype='object')
```

Out[295]:
```
Index(['loanId', 'anon_ssn', 'payFrequency', 'apr', 'originated', 'nPaidOff',
       'approved', 'isFunded', 'loanStatus', 'loanAmount',
       'originallyScheduledPaymentAmount', 'state', 'leadType', 'leadCost',
       'hasCF', 'Application_Year', 'Application_Month'],
      dtype='object')
```

In [296…
```
loan["payFrequency"].value_counts()
```

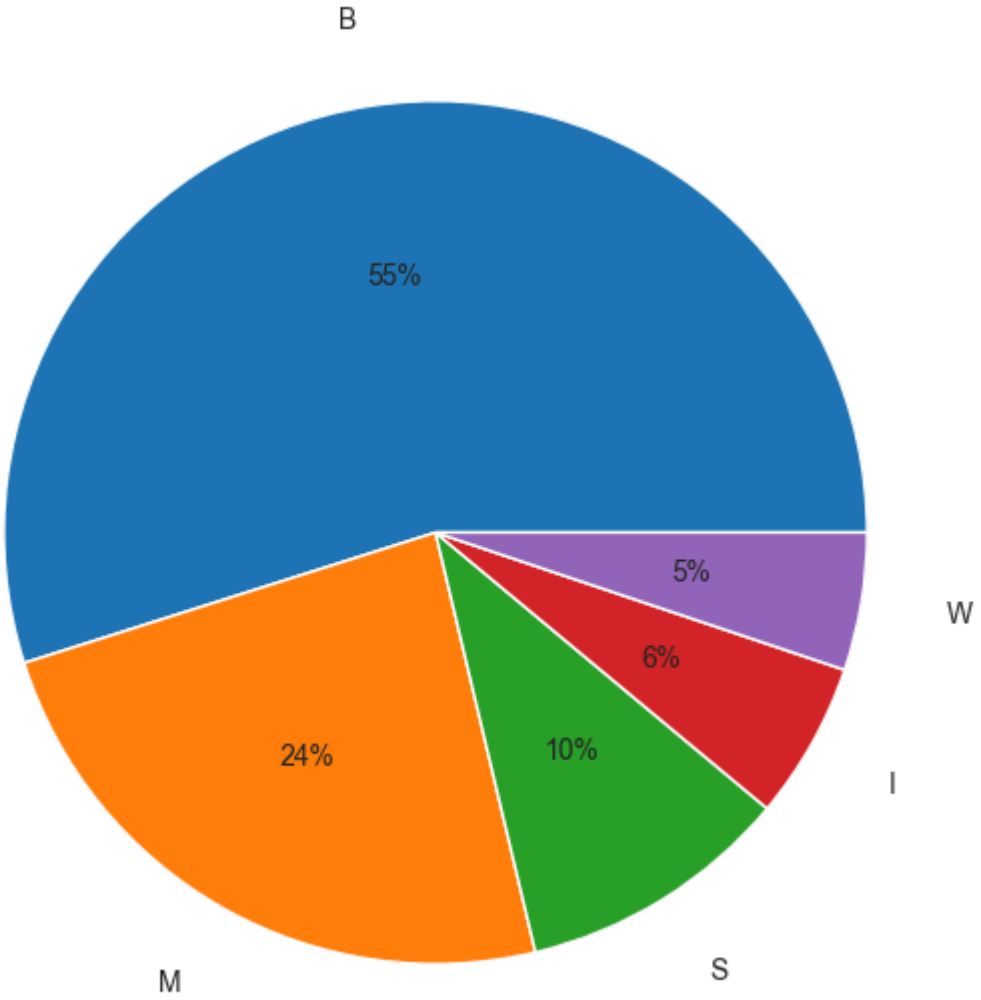
Out[296]:
```
payFrequency
B    313900
W    136288
M     58501
I     33687
S     29491
Name: count, dtype: int64
```

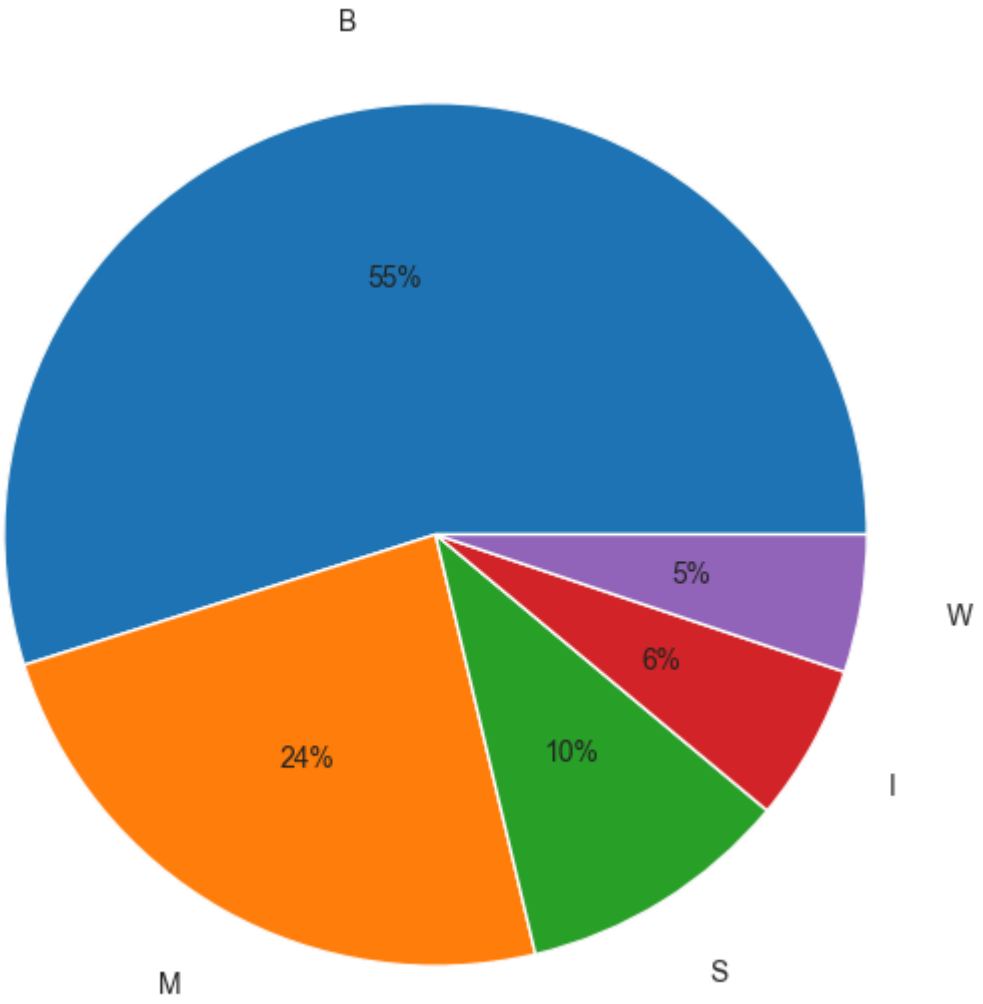
Out[296]:
```
payFrequency
B    313900
W    136288
M     58501
I     33687
S     29491
Name: count, dtype: int64
```

1. we can say that biweekly payments coustome is more which is 55 parcentile, so people are more prefer the biweekly payments. If people more like biweekly then it is more covinent to add addin option which upsealing.

2. the option **semi monthly** is less all of them. so the we can make it more atractive using offers.

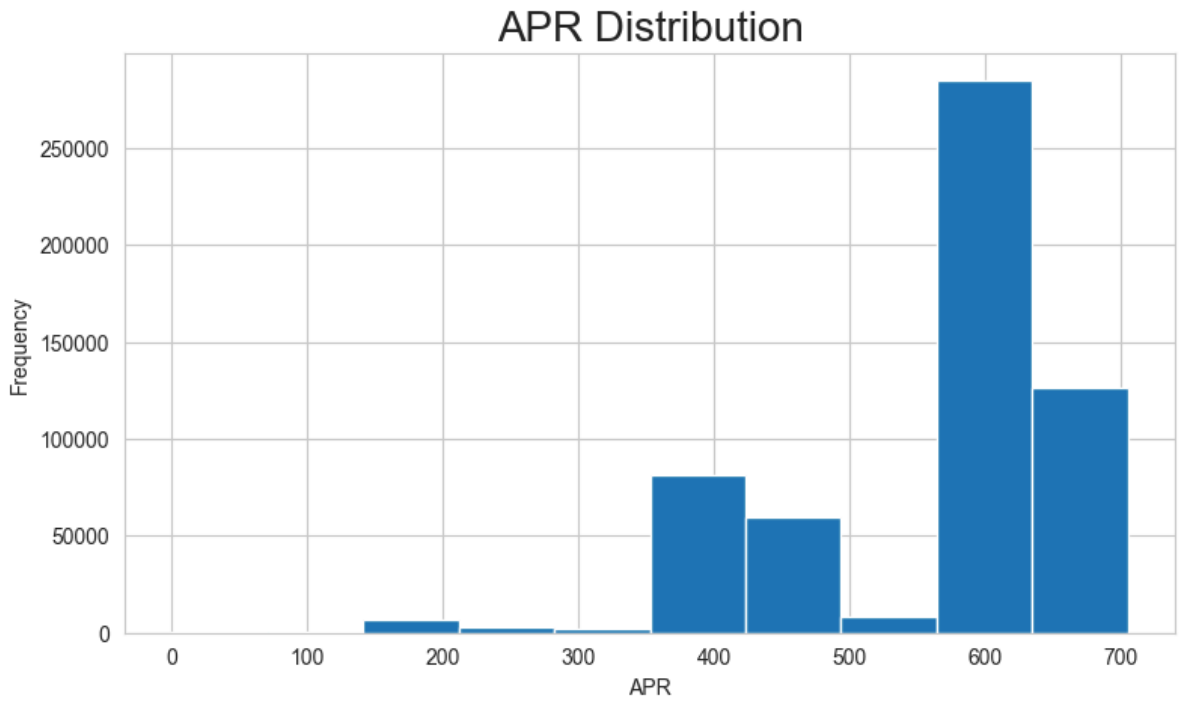
In [297…
```
plt.figure(figsize=(7,7))
plt.pie(x = loan["payFrequency"].value_counts(),labels=list(loan["payFrequency"].ur
plt.show()
```

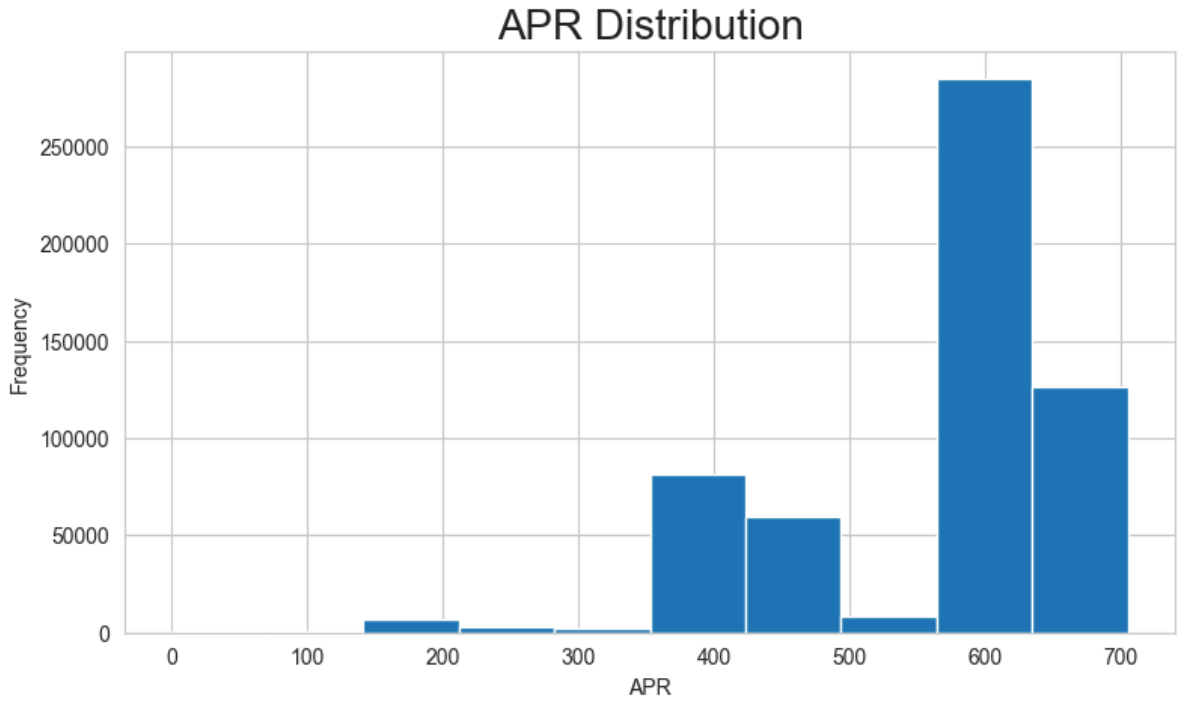

B
55%
W
5%
I
6%
S
10%
M
24%



In [298…

```
plt.figure(figsize=(9,5))
plt.hist(x = 'apr',data=loan)
plt.title('APR Distribution',size = 20)
plt.xlabel('APR')
plt.ylabel('Frequency')
plt.show()
```





we can say from the apr distribution data, 600 value is more popular compare to other options

In [299…
```
loan["loanStatus"].unique()
```

Out[299]:
```
array(['Withdrawn Application', 'Paid Off Loan', 'Rejected', 'New Loan',
       'Internal Collection', 'CSR Voided New Loan',
       'External Collection', 'Returned Item', 'Customer Voided New Loan',
       'Credit Return Void', 'Pending Paid Off', 'Charged Off Paid Off',
       'Settled Bankruptcy', 'Settlement Paid Off', 'Charged Off',
       'Pending Rescind', 'Customver Voided New Loan',
       'Pending Application', 'Voided New Loan',
       'Pending Application Fee', 'Settlement Pending Paid Off'],
      dtype=object)
```

Out[299]:
```
array(['Withdrawn Application', 'Paid Off Loan', 'Rejected', 'New Loan',
       'Internal Collection', 'CSR Voided New Loan',
       'External Collection', 'Returned Item', 'Customer Voided New Loan',
       'Credit Return Void', 'Pending Paid Off', 'Charged Off Paid Off',
       'Settled Bankruptcy', 'Settlement Paid Off', 'Charged Off',
       'Pending Rescind', 'Customver Voided New Loan',
       'Pending Application', 'Voided New Loan',
       'Pending Application Fee', 'Settlement Pending Paid Off'],
      dtype=object)
```

In [300…
```
crosstabdata=pd.crosstab(index = loan['approved'], columns = loan['state'],margins=
crosstabdata
```

Out[300]:

state	AK	AL	AZ	CA	CO	CT	DE	FL	GA	HI	…	SD	TN	TX	T
approved															
False	254	4059	2715	20745	3138	1099	801	23968	1573	565	…	2107	31409	46384	
True	33	256	621	1676	464	361	77	1844	145	52	…	89	1352	3164	
All	287	4315	3336	22421	3602	1460	878	25812	1718	617	…	2196	32761	49548	

3 rows × 45 columns

Out[300]:

state	AK	AL	AZ	CA	CO	CT	DE	FL	GA	HI	…	SD	TN	TX	T
approved															
False	254	4059	2715	20745	3138	1099	801	23968	1573	565	…	2107	31409	46384	
True	33	256	621	1676	464	361	77	1844	145	52	…	89	1352	3164	
All	287	4315	3336	22421	3602	1460	878	25812	1718	617	…	2196	32761	49548	

3 rows × 45 columns

we can compare the value of chancess of approve with respect to the state.

In [301…
```
crosstabdata.index
```

Out[301]:
```
Index([False, True, 'All'], dtype='object', name='approved')
```

Out[301]:
```
Index([False, True, 'All'], dtype='object', name='approved')
```

In [302…
```
loan["leadType"].value_counts()
```

Out[302]:
```
leadType
bvMandatory      472031
lead              71445
organic           21365
prescreen          4420
rc_returning       2040
california          479
repeat               24
express              22
instant-offer        21
lionpay              20
Name: count, dtype: int64
```

Out[302]:
```
leadType
bvMandatory      472031
lead              71445
organic           21365
prescreen          4420
rc_returning       2040
california          479
repeat               24
express              22
instant-offer        21
lionpay              20
Name: count, dtype: int64
```

from the dataset **bvMandatory** is more leadType compayer to others

In [303…
```
loan["state"].value_counts()
```

Out[303]:
```
state
OH         88240
IL         66205
TX         49548
MO         48955
WI         40095
MI         34387
TN         32761
NC         26724
FL         25812
IN         25550
SC         23420
CA         22421
NV         11427
PA          9685
VA          9172
NJ          7615
UT          6716
AL          4315
MS          3771
CO          3602
LA          3421
AZ          3336
NM          3225
KY          2889
SD          2196
MN          2069
OK          1753
GA          1718
WY          1708
CT          1460
WA          1379
KS          1222
IA          1038
DE           878
ID           721
RI           686
NE           634
HI           617
AK           287
ND           205
NY             1
MD             1
OH-TEST        1
TX-TEST        1
Name: count, dtype: int64
```

Out[303]:
```
state
OH         88240
IL         66205
TX         49548
MO         48955
WI         40095
MI         34387
TN         32761
NC         26724
FL         25812
IN         25550
SC         23420
CA         22421
NV         11427
PA          9685
VA          9172
NJ          7615
UT          6716
AL          4315
MS          3771
CO          3602
LA          3421
AZ          3336
NM          3225
KY          2889
SD          2196
MN          2069
OK          1753
GA          1718
WY          1708
CT          1460
WA          1379
KS          1222
IA          1038
DE           878
ID           721
RI           686
NE           634
HI           617
AK           287
ND           205
NY             1
MD             1
OH-TEST        1
TX-TEST        1
Name: count, dtype: int64
```

In [304…
```
loan
```

Out[304]:

	loanId	anon_ssn	payFrequency	apr	originated	nPaidOff
0	LL-I-07399092	beff4989be82aab4a5b47679216942fd	B	360.0	False	0.0
1	LL-I-06644937	464f5d9ae4fa09ece4048d949191865c	B	199.0	True	0.0
2	LL-I-10707532	3c174ae9e2505a5f9ddbff9843281845	B	590.0	False	0.0
3	LL-I-02272596	9be6f443bb97db7e95fa0c281d34da91	B	360.0	False	0.0
4	LL-I-09542882	63b5494f60b5c19c827c7b068443752c	B	590.0	False	0.0
...	...	...	...	...	...	...
577677	LL-I-12122269	801262d04720d32040612759857f4147	B	590.0	False	0.0
577678	LL-I-16183462	e37750de9d99a67e0fa96a51e86fdf5b	S	490.0	False	0.0
577679	LL-I-06962710	d7e55e85266208ac4c353f42ebcde5ca	B	590.0	False	0.0
577680	LL-I-01253468	c3b35307cb36116bf59574f9138d3dad	B	550.0	False	0.0
577681	LL-I-04733921	dc0a43b16c037ee5d0142daebb5db83a	I	590.0	False	0.0

571867 rows × 17 columns

Out[304]:

	loanId	anon_ssn	payFrequency	apr	originated	nPaidOff
0	LL-I-07399092	beff4989be82aab4a5b47679216942fd	B	360.0	False	0.0
1	LL-I-06644937	464f5d9ae4fa09ece4048d949191865c	B	199.0	True	0.0
2	LL-I-10707532	3c174ae9e2505a5f9ddbff9843281845	B	590.0	False	0.0
3	LL-I-02272596	9be6f443bb97db7e95fa0c281d34da91	B	360.0	False	0.0
4	LL-I-09542882	63b5494f60b5c19c827c7b068443752c	B	590.0	False	0.0
...	...	...	...	...	...	...
577677	LL-I-12122269	801262d04720d32040612759857f4147	B	590.0	False	0.0
577678	LL-I-16183462	e37750de9d99a67e0fa96a51e86fdf5b	S	490.0	False	0.0
577679	LL-I-06962710	d7e55e85266208ac4c353f42ebcde5ca	B	590.0	False	0.0
577680	LL-I-01253468	c3b35307cb36116bf59574f9138d3dad	B	550.0	False	0.0
577681	LL-I-04733921	dc0a43b16c037ee5d0142daebb5db83a	I	590.0	False	0.0

571867 rows × 17 columns

In [305…

```
loan.dtypes
```

Out[305]:

```
loanId                             object
anon_ssn                           object
payFrequency                       object
apr                               float64
originated                           bool
nPaidOff                          float64
approved                             bool
isFunded                            int64
loanStatus                         object
loanAmount                        float64
originallyScheduledPaymentAmount  float64
state                              object
leadType                           object
leadCost                            int64
hasCF                               int64
Application_Year                   object
Application_Month                  object
dtype: object
```

Out[305]:
```
loanId                                object
anon_ssn                              object
payFrequency                          object
apr                                  float64
originated                              bool
nPaidOff                             float64
approved                                bool
isFunded                               int64
loanStatus                            object
loanAmount                           float64
originallyScheduledPaymentAmount     float64
state                                 object
leadType                              object
leadCost                               int64
hasCF                                  int64
Application_Year                      object
Application_Month                     object
dtype: object
```

In [306…
```
loan.value_counts("originated")
```

Out[306]:
```
originated
False    525882
True      45985
Name: count, dtype: int64
```

Out[306]:
```
originated
False    525882
True      45985
Name: count, dtype: int64
```

In [307…
```
vc=variables.columns
```

In [308…
```
vc
```

Out[308]:
```
Index(['.underwritingdataclarity.clearfraud.clearfraudinquiry.thirtydaysago',
       '.underwritingdataclarity.clearfraud.clearfraudinquiry.twentyfourhoursago',
       '.underwritingdataclarity.clearfraud.clearfraudinquiry.oneminuteago',
       '.underwritingdataclarity.clearfraud.clearfraudinquiry.onehourago',
       '.underwritingdataclarity.clearfraud.clearfraudinquiry.ninetydaysago',
       '.underwritingdataclarity.clearfraud.clearfraudinquiry.sevendaysago',
       '.underwritingdataclarity.clearfraud.clearfraudinquiry.tenminutesago',
       '.underwritingdataclarity.clearfraud.clearfraudinquiry.fifteendaysago',
       '.underwritingdataclarity.clearfraud.clearfraudinquiry.threesixtyfivedaysag
o',
       '.underwritingdataclarity.clearfraud.clearfraudindicator.inquiryonfilecurre
ntaddressconflict',
       '.underwritingdataclarity.clearfraud.clearfraudindicator.totalnumberoffraud
indicators',
       '.underwritingdataclarity.clearfraud.clearfraudindicator.telephonenumberinc
onsistentwithaddress',
       '.underwritingdataclarity.clearfraud.clearfraudindicator.inquiryageyoungert
hanssnissuedate',
       '.underwritingdataclarity.clearfraud.clearfraudindicator.onfileaddresscauti
ous',
       '.underwritingdataclarity.clearfraud.clearfraudindicator.inquiryaddressnonr
esidential',
       '.underwritingdataclarity.clearfraud.clearfraudindicator.onfileaddresshighr
isk',
       '.underwritingdataclarity.clearfraud.clearfraudindicator.ssnreportedmorefre
quentlyforanother',
       '.underwritingdataclarity.clearfraud.clearfraudindicator.currentaddressrepo
rtedbytradeopenlt90days',
       '.underwritingdataclarity.clearfraud.clearfraudindicator.inputssninvalid',
       '.underwritingdataclarity.clearfraud.clearfraudindicator.inputssnissuedatec
annotbeverified',
       '.underwritingdataclarity.clearfraud.clearfraudindicator.inquiryaddresscaut
ious',
       '.underwritingdataclarity.clearfraud.clearfraudindicator.morethan3inquiries
inthelast30days',
       '.underwritingdataclarity.clearfraud.clearfraudindicator.onfileaddressnonre
sidential',
       '.underwritingdataclarity.clearfraud.clearfraudindicator.creditestablishedp
riortossnissuedate',
       '.underwritingdataclarity.clearfraud.clearfraudindicator.driverlicenseforma
tinvalid',
       '.underwritingdataclarity.clearfraud.clearfraudindicator.inputssnrecordedas
deceased',
       '.underwritingdataclarity.clearfraud.clearfraudindicator.inquiryaddresshigh
risk',
       '.underwritingdataclarity.clearfraud.clearfraudindicator.inquirycurrentaddr
essnotonfile',
       '.underwritingdataclarity.clearfraud.clearfraudindicator.bestonfilessnissue
datecannotbeverified',
       '.underwritingdataclarity.clearfraud.clearfraudindicator.highprobabilityssn
belongstoanother',
       '.underwritingdataclarity.clearfraud.clearfraudindicator.maxnumberofssnswit
hanybankaccount',
       '.underwritingdataclarity.clearfraud.clearfraudindicator.bestonfilessnrecor
dedasdeceased',
       '.underwritingdataclarity.clearfraud.clearfraudindicator.currentaddressrepo
rtedbynewtradeonly',
       '.underwritingdataclarity.clearfraud.clearfraudindicator.creditestablishedb
eforeage18',
       '.underwritingdataclarity.clearfraud.clearfraudindicator.telephonenumberinc
onsistentwithstate',
       '.underwritingdataclarity.clearfraud.clearfraudindicator.driverlicenseincon
sistentwithonfile',
       '.underwritingdataclarity.clearfraud.clearfraudindicator.workphonepreviousl
```

```
ylistedascellphone',
       '.underwritingdataclarity.clearfraud.clearfraudindicator.workphonepreviousl
ylistedashomephone',
       '.underwritingdataclarity.clearfraud.clearfraudidentityverification.ssnname
match',
       '.underwritingdataclarity.clearfraud.clearfraudidentityverification.nameadd
ressmatch',
       '.underwritingdataclarity.clearfraud.clearfraudidentityverification.phonema
tchtype',
       '.underwritingdataclarity.clearfraud.clearfraudidentityverification.ssnname
reasoncodedescription',
       '.underwritingdataclarity.clearfraud.clearfraudidentityverification.phonema
tchresult',
       '.underwritingdataclarity.clearfraud.clearfraudidentityverification.nameadd
ressreasoncodedescription',
       '.underwritingdataclarity.clearfraud.clearfraudidentityverification.phonema
tchtypedescription',
       '.underwritingdataclarity.clearfraud.clearfraudidentityverification.overall
matchresult',
       '.underwritingdataclarity.clearfraud.clearfraudidentityverification.phonety
pe',
       '.underwritingdataclarity.clearfraud.clearfraudidentityverification.ssndobr
easoncode',
       '.underwritingdataclarity.clearfraud.clearfraudidentityverification.ssnname
reasoncode',
       '.underwritingdataclarity.clearfraud.clearfraudidentityverification.nameadd
ressreasoncode',
       '.underwritingdataclarity.clearfraud.clearfraudidentityverification.ssndobm
atch',
       '.underwritingdataclarity.clearfraud.clearfraudidentityverification.overall
matchreasoncode',
       'clearfraudscore', 'underwritingid'],
      dtype='object')
```

Out[308]:
```
Index(['.underwritingdataclarity.clearfraud.clearfraudinquiry.thirtydaysago',
       '.underwritingdataclarity.clearfraud.clearfraudinquiry.twentyfourhoursago',
       '.underwritingdataclarity.clearfraud.clearfraudinquiry.oneminuteago',
       '.underwritingdataclarity.clearfraud.clearfraudinquiry.onehourago',
       '.underwritingdataclarity.clearfraud.clearfraudinquiry.ninetydaysago',
       '.underwritingdataclarity.clearfraud.clearfraudinquiry.sevendaysago',
       '.underwritingdataclarity.clearfraud.clearfraudinquiry.tenminutesago',
       '.underwritingdataclarity.clearfraud.clearfraudinquiry.fifteendaysago',
       '.underwritingdataclarity.clearfraud.clearfraudinquiry.threesixtyfivedaysag
o',
       '.underwritingdataclarity.clearfraud.clearfraudindicator.inquiryonfilecurre
ntaddressconflict',
       '.underwritingdataclarity.clearfraud.clearfraudindicator.totalnumberoffraud
indicators',
       '.underwritingdataclarity.clearfraud.clearfraudindicator.telephonenumberinc
onsistentwithaddress',
       '.underwritingdataclarity.clearfraud.clearfraudindicator.inquiryageyoungert
hanssnissuedate',
       '.underwritingdataclarity.clearfraud.clearfraudindicator.onfileaddresscauti
ous',
       '.underwritingdataclarity.clearfraud.clearfraudindicator.inquiryaddressnonr
esidential',
       '.underwritingdataclarity.clearfraud.clearfraudindicator.onfileaddresshighr
isk',
       '.underwritingdataclarity.clearfraud.clearfraudindicator.ssnreportedmorefre
quentlyforanother',
       '.underwritingdataclarity.clearfraud.clearfraudindicator.currentaddressrepo
rtedbytradeopenlt90days',
       '.underwritingdataclarity.clearfraud.clearfraudindicator.inputssninvalid',
       '.underwritingdataclarity.clearfraud.clearfraudindicator.inputssnissuedatec
annotbeverified',
       '.underwritingdataclarity.clearfraud.clearfraudindicator.inquiryaddresscaut
ious',
       '.underwritingdataclarity.clearfraud.clearfraudindicator.morethan3inquiries
inthelast30days',
       '.underwritingdataclarity.clearfraud.clearfraudindicator.onfileaddressnonre
sidential',
       '.underwritingdataclarity.clearfraud.clearfraudindicator.creditestablishedp
riortossnissuedate',
       '.underwritingdataclarity.clearfraud.clearfraudindicator.driverlicenseforma
tinvalid',
       '.underwritingdataclarity.clearfraud.clearfraudindicator.inputssnrecordedas
deceased',
       '.underwritingdataclarity.clearfraud.clearfraudindicator.inquiryaddresshigh
risk',
       '.underwritingdataclarity.clearfraud.clearfraudindicator.inquirycurrentaddr
essnotonfile',
       '.underwritingdataclarity.clearfraud.clearfraudindicator.bestonfilessnissue
datecannotbeverified',
       '.underwritingdataclarity.clearfraud.clearfraudindicator.highprobabilityssn
belongstoanother',
       '.underwritingdataclarity.clearfraud.clearfraudindicator.maxnumberofssnswit
hanybankaccount',
       '.underwritingdataclarity.clearfraud.clearfraudindicator.bestonfilessnrecor
dedasdeceased',
       '.underwritingdataclarity.clearfraud.clearfraudindicator.currentaddressrepo
rtedbynewtradeonly',
       '.underwritingdataclarity.clearfraud.clearfraudindicator.creditestablishedb
eforeage18',
       '.underwritingdataclarity.clearfraud.clearfraudindicator.telephonenumberinc
onsistentwithstate',
       '.underwritingdataclarity.clearfraud.clearfraudindicator.driverlicenseincon
sistentwithonfile',
       '.underwritingdataclarity.clearfraud.clearfraudindicator.workphonepreviousl
```

```
ylistedascellphone',
       '.underwritingdataclarity.clearfraud.clearfraudindicator.workphonepreviousl
ylistedashomephone',
       '.underwritingdataclarity.clearfraud.clearfraudidentityverification.ssnname
match',
       '.underwritingdataclarity.clearfraud.clearfraudidentityverification.nameadd
ressmatch',
       '.underwritingdataclarity.clearfraud.clearfraudidentityverification.phonema
tchtype',
       '.underwritingdataclarity.clearfraud.clearfraudidentityverification.ssnname
reasoncodedescription',
       '.underwritingdataclarity.clearfraud.clearfraudidentityverification.phonema
tchresult',
       '.underwritingdataclarity.clearfraud.clearfraudidentityverification.nameadd
ressreasoncodedescription',
       '.underwritingdataclarity.clearfraud.clearfraudidentityverification.phonema
tchtypedescription',
       '.underwritingdataclarity.clearfraud.clearfraudidentityverification.overall
matchresult',
       '.underwritingdataclarity.clearfraud.clearfraudidentityverification.phonety
pe',
       '.underwritingdataclarity.clearfraud.clearfraudidentityverification.ssndobr
easoncode',
       '.underwritingdataclarity.clearfraud.clearfraudidentityverification.ssnname
reasoncode',
       '.underwritingdataclarity.clearfraud.clearfraudidentityverification.nameadd
ressreasoncode',
       '.underwritingdataclarity.clearfraud.clearfraudidentityverification.ssndobm
atch',
       '.underwritingdataclarity.clearfraud.clearfraudidentityverification.overall
matchreasoncode',
       'clearfraudscore', 'underwritingid'],
      dtype='object')
```

In [309…

```
variables.columns=variables.columns.str.split(".").str[-1]
```

In [310…

```
variables
```

Out[310]:

	thirtydaysago	twentyfourhoursago	oneminuteago	onehourago	ninetydaysago	sevendays
0	8.0	2.0	2.0	2.0	8.0	
1	5.0	2.0	2.0	2.0	11.0	
2	9.0	4.0	2.0	3.0	10.0	
3	3.0	2.0	2.0	2.0	9.0	
4	5.0	5.0	2.0	2.0	6.0	
...	...	...	...	...	...	
49747	2.0	2.0	2.0	2.0	2.0	
49748	6.0	4.0	1.0	4.0	11.0	
49749	4.0	4.0	1.0	4.0	4.0	
49750	3.0	3.0	2.0	2.0	3.0	
49751	5.0	3.0	2.0	2.0	6.0	

49752 rows × 54 columns

Out[310]:

	thirtydaysago	twentyfourhoursago	oneminuteago	onehourago	ninetydaysago	sevendays
0	8.0	2.0	2.0	2.0	8.0	
1	5.0	2.0	2.0	2.0	11.0	
2	9.0	4.0	2.0	3.0	10.0	
3	3.0	2.0	2.0	2.0	9.0	
4	5.0	5.0	2.0	2.0	6.0	
...	...	...	...	...	...	
49747	2.0	2.0	2.0	2.0	2.0	
49748	6.0	4.0	1.0	4.0	11.0	
49749	4.0	4.0	1.0	4.0	4.0	
49750	3.0	3.0	2.0	2.0	3.0	
49751	5.0	3.0	2.0	2.0	6.0	

49752 rows × 54 columns

In [311…

```
result=pd.merge(loan, payment, how="outer", on="loanId")
```

In [312…

```
result
```

Out[312]:

	loanId	anon_ssn	payFrequency	apr	originated	nPaidOff
0	LL-I-07399092	beff4989be82aab4a5b47679216942fd	B	360.0	False	0.0
1	LL-I-06644937	464f5d9ae4fa09ece4048d949191865c	B	199.0	True	0.0
2	LL-I-06644937	464f5d9ae4fa09ece4048d949191865c	B	199.0	True	0.0
3	LL-I-06644937	464f5d9ae4fa09ece4048d949191865c	B	199.0	True	0.0
4	LL-I-06644937	464f5d9ae4fa09ece4048d949191865c	B	199.0	True	0.0
...	...	...	...	...	...	...
1221295	LP-I-00000073	NaN	NaN	NaN	NaN	NaN
1221296	LP-I-00000073	NaN	NaN	NaN	NaN	NaN
1221297	LP-I-00000073	NaN	NaN	NaN	NaN	NaN
1221298	LP-I-00000073	NaN	NaN	NaN	NaN	NaN
1221299	LP-I-00000073	NaN	NaN	NaN	NaN	NaN

1221300 rows × 25 columns

Out[312]:

	loanId	anon_ssn	payFrequency	apr	originated	nPaidOff
0	LL-I-07399092	beff4989be82aab4a5b47679216942fd	B	360.0	False	0.0
1	LL-I-06644937	464f5d9ae4fa09ece4048d949191865c	B	199.0	True	0.0
2	LL-I-06644937	464f5d9ae4fa09ece4048d949191865c	B	199.0	True	0.0
3	LL-I-06644937	464f5d9ae4fa09ece4048d949191865c	B	199.0	True	0.0
4	LL-I-06644937	464f5d9ae4fa09ece4048d949191865c	B	199.0	True	0.0
...	...	...	...	...	...	...
1221295	LP-I-00000073	NaN	NaN	NaN	NaN	NaN
1221296	LP-I-00000073	NaN	NaN	NaN	NaN	NaN
1221297	LP-I-00000073	NaN	NaN	NaN	NaN	NaN
1221298	LP-I-00000073	NaN	NaN	NaN	NaN	NaN
1221299	LP-I-00000073	NaN	NaN	NaN	NaN	NaN

1221300 rows × 25 columns

In [313…
```
payment.isna().sum()
```

Out[313]:
```
loanId                0
installmentIndex      0
isCollection          0
paymentDate           0
principal             0
fees                  0
paymentAmount         0
paymentStatus        164057
paymentReturnCode    657831
dtype: int64
```

Out[313]:
```
loanId                0
installmentIndex      0
isCollection          0
paymentDate           0
principal             0
fees                  0
paymentAmount         0
paymentStatus        164057
paymentReturnCode    657831
dtype: int64
```

In [314…
```
loan.isna().sum()
```

Out[314]:
```
loanId                                0
anon_ssn                              0
payFrequency                          0
apr                                   0
originated                            0
nPaidOff                              0
approved                              0
isFunded                              0
loanStatus                            0
loanAmount                            0
originallyScheduledPaymentAmount      0
state                                 0
leadType                              0
leadCost                              0
hasCF                                 0
Application_Year                      0
Application_Month                     0
dtype: int64
```

Out[314]:
```
loanId                                0
anon_ssn                              0
payFrequency                          0
apr                                   0
originated                            0
nPaidOff                              0
approved                              0
isFunded                              0
loanStatus                            0
loanAmount                            0
originallyScheduledPaymentAmount      0
state                                 0
leadType                              0
leadCost                              0
hasCF                                 0
Application_Year                      0
Application_Month                     0
dtype: int64
```

In [315…
```
result.isna().sum()
```

Out[315]:
```
loanId                                0
anon_ssn                            383
payFrequency                        383
apr                                 383
originated                          383
nPaidOff                            383
approved                            383
isFunded                            383
loanStatus                          383
loanAmount                          383
originallyScheduledPaymentAmount    383
state                               383
leadType                            383
leadCost                            383
hasCF                               383
Application_Year                    383
Application_Month                   383
installmentIndex                 531936
isCollection                     531936
paymentDate                      531936
principal                        531936
fees                             531936
paymentAmount                    531936
paymentStatus                    695993
paymentReturnCode               1189767
dtype: int64
```

Out[315]:
```
loanId                                0
anon_ssn                            383
payFrequency                        383
apr                                 383
originated                          383
nPaidOff                            383
approved                            383
isFunded                            383
loanStatus                          383
loanAmount                          383
originallyScheduledPaymentAmount    383
state                               383
leadType                            383
leadCost                            383
hasCF                               383
Application_Year                    383
Application_Month                   383
installmentIndex                 531936
isCollection                     531936
paymentDate                      531936
principal                        531936
fees                             531936
paymentAmount                    531936
paymentStatus                    695993
paymentReturnCode               1189767
dtype: int64
```

In [316…
```
result.shape
```

Out[316]:
```
(1221300, 25)
```

Out[316]:
```
(1221300, 25)
```

In [317…
```
result.drop("paymentReturnCode",axis=1,inplace=True)
```

In [318…
```
result.columns
```

Out[318]:
```
Index(['loanId', 'anon_ssn', 'payFrequency', 'apr', 'originated', 'nPaidOff',
       'approved', 'isFunded', 'loanStatus', 'loanAmount',
       'originallyScheduledPaymentAmount', 'state', 'leadType', 'leadCost',
       'hasCF', 'Application_Year', 'Application_Month', 'installmentIndex',
       'isCollection', 'paymentDate', 'principal', 'fees', 'paymentAmount',
       'paymentStatus'],
      dtype='object')
```

Out[318]:
```
Index(['loanId', 'anon_ssn', 'payFrequency', 'apr', 'originated', 'nPaidOff',
       'approved', 'isFunded', 'loanStatus', 'loanAmount',
       'originallyScheduledPaymentAmount', 'state', 'leadType', 'leadCost',
       'hasCF', 'Application_Year', 'Application_Month', 'installmentIndex',
       'isCollection', 'paymentDate', 'principal', 'fees', 'paymentAmount',
       'paymentStatus'],
      dtype='object')
```

In [319…
```
result
```

Out[319]:

	loanId	anon_ssn	payFrequency	apr	originated	nPaidOff
0	LL-I-07399092	beff4989be82aab4a5b47679216942fd	B	360.0	False	0.0
1	LL-I-06644937	464f5d9ae4fa09ece4048d949191865c	B	199.0	True	0.0
2	LL-I-06644937	464f5d9ae4fa09ece4048d949191865c	B	199.0	True	0.0
3	LL-I-06644937	464f5d9ae4fa09ece4048d949191865c	B	199.0	True	0.0
4	LL-I-06644937	464f5d9ae4fa09ece4048d949191865c	B	199.0	True	0.0
...	...	...	...	...	...	...
1221295	LP-I-00000073	NaN	NaN	NaN	NaN	NaN
1221296	LP-I-00000073	NaN	NaN	NaN	NaN	NaN
1221297	LP-I-00000073	NaN	NaN	NaN	NaN	NaN
1221298	LP-I-00000073	NaN	NaN	NaN	NaN	NaN
1221299	LP-I-00000073	NaN	NaN	NaN	NaN	NaN

1221300 rows × 24 columns

Out[319]:

	loanId	anon_ssn	payFrequency	apr	originated	nPaidOff
0	LL-I-07399092	beff4989be82aab4a5b47679216942fd	B	360.0	False	0.0
1	LL-I-06644937	464f5d9ae4fa09ece4048d949191865c	B	199.0	True	0.0
2	LL-I-06644937	464f5d9ae4fa09ece4048d949191865c	B	199.0	True	0.0
3	LL-I-06644937	464f5d9ae4fa09ece4048d949191865c	B	199.0	True	0.0
4	LL-I-06644937	464f5d9ae4fa09ece4048d949191865c	B	199.0	True	0.0
...	...	...	...	...	...	...
1221295	LP-I-00000073	NaN	NaN	NaN	NaN	NaN
1221296	LP-I-00000073	NaN	NaN	NaN	NaN	NaN
1221297	LP-I-00000073	NaN	NaN	NaN	NaN	NaN
1221298	LP-I-00000073	NaN	NaN	NaN	NaN	NaN
1221299	LP-I-00000073	NaN	NaN	NaN	NaN	NaN

1221300 rows × 24 columns

In [320…

```
result.isna().sum()
```

Out[320]:

```
loanId                                0
anon_ssn                            383
payFrequency                        383
apr                                 383
originated                          383
nPaidOff                            383
approved                            383
isFunded                            383
loanStatus                          383
loanAmount                          383
originallyScheduledPaymentAmount    383
state                               383
leadType                            383
leadCost                            383
hasCF                               383
Application_Year                    383
Application_Month                   383
installmentIndex                 531936
isCollection                     531936
paymentDate                      531936
principal                        531936
fees                             531936
paymentAmount                    531936
paymentStatus                    695993
dtype: int64
```

Out[320]:
```
loanId                                0
anon_ssn                            383
payFrequency                        383
apr                                 383
originated                          383
nPaidOff                            383
approved                            383
isFunded                            383
loanStatus                          383
loanAmount                          383
originallyScheduledPaymentAmount    383
state                               383
leadType                            383
leadCost                            383
hasCF                               383
Application_Year                    383
Application_Month                   383
installmentIndex                 531936
isCollection                     531936
paymentDate                      531936
principal                        531936
fees                             531936
paymentAmount                    531936
paymentStatus                    695993
dtype: int64
```

In [321…
```
result.dropna(subset=list(result.columns[(result.isna().sum()>0)==True]), axis=0, i
```

In [322…
```
result.isna().sum()
```

Out[322]:
```
loanId                            0
anon_ssn                          0
payFrequency                      0
apr                               0
originated                        0
nPaidOff                          0
approved                          0
isFunded                          0
loanStatus                        0
loanAmount                        0
originallyScheduledPaymentAmount  0
state                             0
leadType                          0
leadCost                          0
hasCF                             0
Application_Year                  0
Application_Month                 0
installmentIndex                  0
isCollection                      0
paymentDate                       0
principal                         0
fees                              0
paymentAmount                     0
paymentStatus                     0
dtype: int64
```

Out[322]:

loanId	0
anon_ssn	0
payFrequency	0
apr	0
originated	0
nPaidOff	0
approved	0
isFunded	0
loanStatus	0
loanAmount	0
originallyScheduledPaymentAmount	0
state	0
leadType	0
leadCost	0
hasCF	0
Application_Year	0
Application_Month	0
installmentIndex	0
isCollection	0
paymentDate	0
principal	0
fees	0
paymentAmount	0
paymentStatus	0

dtype: int64

In [323…

```
result.iloc[:,2:].dtypes
```

Out[323]:

payFrequency	object
apr	float64
originated	object
nPaidOff	float64
approved	object
isFunded	float64
loanStatus	object
loanAmount	float64
originallyScheduledPaymentAmount	float64
state	object
leadType	object
leadCost	float64
hasCF	float64
Application_Year	object
Application_Month	object
installmentIndex	float64
isCollection	object
paymentDate	object
principal	float64
fees	float64
paymentAmount	float64
paymentStatus	object

dtype: object

Out[323]:
```
payFrequency                        object
apr                                float64
originated                          object
nPaidOff                           float64
approved                            object
isFunded                           float64
loanStatus                          object
loanAmount                         float64
originallyScheduledPaymentAmount   float64
state                               object
leadType                            object
leadCost                           float64
hasCF                              float64
Application_Year                    object
Application_Month                   object
installmentIndex                   float64
isCollection                        object
paymentDate                         object
principal                          float64
fees                               float64
paymentAmount                      float64
paymentStatus                       object
dtype: object
```

In [324…
```
result[["originated","approved","isCollection"]]=result[["originated","approved","i
```

In [325…
```
payFrequency_map={"B":0,"W":1,"S":2,"M":3,"I":4}
loanStatus_map={"External Collection":0,
                "Paid Off Loan":1,
                "Internal Collection":2,
                "New Loan":3,
                "Settlement Paid Off":4,
                "Credit Return Void":5,
                "Customer Voided New Loan":6,
                "Settled Bankruptcy":7,
                "Returned Item":8,
                "Charged Off Paid Off":9,
                "Pending Paid Off":10,
                "CSR Voided New Loan":11,
                "Pending Rescind":12,
                "Withdrawn Application":13,
                "Voided New Loan":14,
                "Charged Off":15,
                "Settlement Pending Paid Off":16,
                "Customver Voided New Loan":17
                }
leadType_map={"bvMandatory":0,"lead":1,"organic":2,"prescreen":3,"rc_returning":4,'
paymentStatus_map={"Cancelled":0,"Checked":1,"Rejected":2,"Pending":3,"Skipped":4,'

result['payFrequency']=result['payFrequency'].map(payFrequency_map)
result['loanStatus']=result['loanStatus'].map(loanStatus_map)
result['leadType']=result['leadType'].map(leadType_map)
result['paymentStatus']=result['paymentStatus'].map(paymentStatus_map)
```

In [326…
```
state_map={"OH":0,"IL":1,"WI":2,"TX":3,"CA":4,"MI":5,"IN":6,"MO":7, "NC":8,"FL":9,'
                   "PA":12,"NJ":13,"NV":14,"AZ":15,"VA":16,"CT":17,"MN":18,"KY":19,
                   "UT":24,"GA":25,"KS":26,"OK":27,"NM":28,"IA":29,"CO":30,"SD":31,
                   "ID":36,"HI":37,"AK":38,"ND":39,"MD":40}
result['state']=result['state'].map(state_map)
```

In [327…
```
result["state"].value_counts().index
```

Out[327]:
```
Index([ 0,  1,  2,  3,  4,  5,  6,  7,  8,  9, 10, 11, 12, 13, 14, 15, 16, 17,
       18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35,
       36, 37, 38, 39, 40],
      dtype='int64', name='state')
```
Out[327]:
```
Index([ 0,  1,  2,  3,  4,  5,  6,  7,  8,  9, 10, 11, 12, 13, 14, 15, 16, 17,
       18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35,
       36, 37, 38, 39, 40],
      dtype='int64', name='state')
```

In [328…
```
result1=result.copy()
```

In [329…
```
result.drop(["loanId","anon_ssn","paymentDate"], axis=1,inplace=True)
```

In [330…
```
array=result.iloc[:,2:].values
```

In [331…
```
array
```

Out[331]:
```
array([[1, 0.0, 1, ..., 114.49, 266.47, 1],
       [1, 0.0, 1, ..., 217.39, 266.47, 1],
       [1, 0.0, 1, ..., 213.64, 266.47, 1],
       ...,
       [1, 0.0, 1, ..., 11.64, 39.23, 0],
       [1, 0.0, 1, ..., 8.22, 39.23, 0],
       [1, 0.0, 1, ..., 4.39, 39.85, 0]], dtype=object)
```
Out[331]:
```
array([[1, 0.0, 1, ..., 114.49, 266.47, 1],
       [1, 0.0, 1, ..., 217.39, 266.47, 1],
       [1, 0.0, 1, ..., 213.64, 266.47, 1],
       ...,
       [1, 0.0, 1, ..., 11.64, 39.23, 0],
       [1, 0.0, 1, ..., 8.22, 39.23, 0],
       [1, 0.0, 1, ..., 4.39, 39.85, 0]], dtype=object)
```

In [332…
```
stscaler = StandardScaler().fit(array)
X = stscaler.transform(array)
```

In [333…
```
X
```

Out[333]:
```
array([[ 0.01287362, -0.35787597,  0.00861893, ...,  0.82271761,
         1.40712609,  0.53208   ],
       [ 0.01287362, -0.35787597,  0.00861893, ...,  2.58846395,
         1.40712609,  0.53208   ],
       [ 0.01287362, -0.35787597,  0.00861893, ...,  2.5241146 ,
         1.40712609,  0.53208   ],
       ...,
       [ 0.01287362, -0.35787597,  0.00861893, ..., -0.94217074,
        -0.6426671 , -0.81065643],
       [ 0.01287362, -0.35787597,  0.00861893, ..., -1.00085735,
        -0.6426671 , -0.81065643],
       [ 0.01287362, -0.35787597,  0.00861893, ..., -1.06657949,
        -0.63707446, -0.81065643]])
```
Out[333]:
```
array([[ 0.01287362, -0.35787597,  0.00861893, ...,  0.82271761,
         1.40712609,  0.53208   ],
       [ 0.01287362, -0.35787597,  0.00861893, ...,  2.58846395,
         1.40712609,  0.53208   ],
       [ 0.01287362, -0.35787597,  0.00861893, ...,  2.5241146 ,
         1.40712609,  0.53208   ],
       ...,
       [ 0.01287362, -0.35787597,  0.00861893, ..., -0.94217074,
        -0.6426671 , -0.81065643],
       [ 0.01287362, -0.35787597,  0.00861893, ..., -1.00085735,
        -0.6426671 , -0.81065643],
       [ 0.01287362, -0.35787597,  0.00861893, ..., -1.06657949,
        -0.63707446, -0.81065643]])
```

Used DBSCAN to make cluster of DataSet

In [334…
```
dbscan = DBSCAN(eps=0.7, min_samples=7)
dbscan.fit(X)
```

Out[334]:
```
DBSCAN
DBSCAN(eps=0.7, min_samples=7)
```

Out[334]:
```
DBSCAN
DBSCAN(eps=0.7, min_samples=7)
```

In [335…
```
dbscan.labels_
```

Out[335]:
```
array([ 0,  0,  0, ..., 12, 12, 12], dtype=int64)
```

Out[335]:
```
array([ 0,  0,  0, ..., 12, 12, 12], dtype=int64)
```

In [336…
```
cl=pd.DataFrame(dbscan.labels_,columns=['cluster'])
```

In [337…
```
cl
```

Out[337]:

	cluster
0	0
1	0
2	0
3	0
4	0
…	…
525032	12
525033	12
525034	12
525035	12
525036	12

525037 rows × 1 columns

Out[337]:

	cluster
0	0
1	0
2	0
3	0
4	0
...	...
525032	12
525033	12
525034	12
525035	12
525036	12

525037 rows × 1 columns

In [338...
```
DBSCAN_data=pd.concat([result1,cl],axis=1)
```

In [339...
```
DBSCAN_data
```

Out[339]:

	loanId	anon_ssn	payFrequency	apr	originated	nPaidOff
1	LL-I-06644937	464f5d9ae4fa09ece4048d949191865c	0.0	199.0	1.0	0.0
2	LL-I-06644937	464f5d9ae4fa09ece4048d949191865c	0.0	199.0	1.0	0.0
3	LL-I-06644937	464f5d9ae4fa09ece4048d949191865c	0.0	199.0	1.0	0.0
4	LL-I-06644937	464f5d9ae4fa09ece4048d949191865c	0.0	199.0	1.0	0.0
5	LL-I-06644937	464f5d9ae4fa09ece4048d949191865c	0.0	199.0	1.0	0.0
...	...	...	...	...	...	...
525032	NaN	NaN	NaN	NaN	NaN	NaN
525033	NaN	NaN	NaN	NaN	NaN	NaN
525034	NaN	NaN	NaN	NaN	NaN	NaN
525035	NaN	NaN	NaN	NaN	NaN	NaN
525036	NaN	NaN	NaN	NaN	NaN	NaN

824111 rows × 25 columns

Out[339]:

	loanId	anon_ssn	payFrequency	apr	originated	nPaidOff
1	LL-I-06644937	464f5d9ae4fa09ece4048d949191865c	0.0	199.0	1.0	0.0
2	LL-I-06644937	464f5d9ae4fa09ece4048d949191865c	0.0	199.0	1.0	0.0
3	LL-I-06644937	464f5d9ae4fa09ece4048d949191865c	0.0	199.0	1.0	0.0
4	LL-I-06644937	464f5d9ae4fa09ece4048d949191865c	0.0	199.0	1.0	0.0
5	LL-I-06644937	464f5d9ae4fa09ece4048d949191865c	0.0	199.0	1.0	0.0
...	...	...	...	...	...	...
525032	NaN	NaN	NaN	NaN	NaN	NaN
525033	NaN	NaN	NaN	NaN	NaN	NaN
525034	NaN	NaN	NaN	NaN	NaN	NaN
525035	NaN	NaN	NaN	NaN	NaN	NaN
525036	NaN	NaN	NaN	NaN	NaN	NaN

824111 rows × 25 columns

In [339…