Part_I_exploration_template

November 12, 2022

1 Part I - (Prosper Loan Data)

1.1 by (DAISSINTA BAIDI Sammy Salim)

1.2 Introduction

this document exploring dataset that contains 113,937 loans with 81 variables on each loan

1.3 Preliminary Wrangling

```
In [1]: # import all packages and set plots to be embedded inline
        import numpy as np
        import pandas as pd
        import matplotlib.pyplot as plt
        import seaborn as sb
        # suppress warnings from final output
        import warnings
        warnings.simplefilter("ignore")

        %matplotlib inline
```

Load in your dataset and describe its properties through the questions below. Try and motivate your exploration goals through this section.

```
In [2]: loan=pd.read_csv('prosperLoanData.csv')
```

1.4 high-level overview

1.4.1 1-number of rows and lignes in our dataset

```
In [3]: loan.shape
```

```
Out[3]: (113937, 81)
```

1.4.2 2-type of variables of our dataset

```
In [4]: loan.dtypes
```

Out[4]:
```
ListingKey                          object
ListingNumber                        int64
ListingCreationDate                 object
CreditGrade                         object
Term                                 int64
LoanStatus                          object
ClosedDate                          object
BorrowerAPR                        float64
BorrowerRate                       float64
LenderYield                        float64
EstimatedEffectiveYield            float64
EstimatedLoss                      float64
EstimatedReturn                    float64
ProsperRating (numeric)            float64
ProsperRating (Alpha)               object
ProsperScore                       float64
ListingCategory (numeric)            int64
BorrowerState                       object
Occupation                          object
EmploymentStatus                    object
EmploymentStatusDuration           float64
IsBorrowerHomeowner                   bool
CurrentlyInGroup                      bool
GroupKey                            object
DateCreditPulled                    object
CreditScoreRangeLower              float64
CreditScoreRangeUpper              float64
FirstRecordedCreditLine             object
CurrentCreditLines                 float64
OpenCreditLines                    float64
                                    ...
TotalProsperLoans                  float64
TotalProsperPaymentsBilled         float64
OnTimeProsperPayments              float64
ProsperPaymentsLessThanOneMonthLate float64
ProsperPaymentsOneMonthPlusLate    float64
ProsperPrincipalBorrowed           float64
ProsperPrincipalOutstanding        float64
ScorexChangeAtTimeOfListing        float64
LoanCurrentDaysDelinquent            int64
LoanFirstDefaultedCycleNumber      float64
LoanMonthsSinceOrigination           int64
LoanNumber                           int64
LoanOriginalAmount                   int64
LoanOriginationDate                 object
LoanOriginationQuarter              object
MemberKey                           object
MonthlyLoanPayment                 float64
```

```
LP_CustomerPayments                      float64
LP_CustomerPrincipalPayments             float64
LP_InterestandFees                       float64
LP_ServiceFees                           float64
LP_CollectionFees                        float64
LP_GrossPrincipalLoss                    float64
LP_NetPrincipalLoss                      float64
LP_NonPrincipalRecoverypayments          float64
PercentFunded                            float64
Recommendations                            int64
InvestmentFromFriendsCount                 int64
InvestmentFromFriendsAmount              float64
Investors                                  int64
Length: 81, dtype: object
```

1.4.3 3- five's first rows and lignes of our dataset

In [5]:
```
loan.head(10)
```

Out[5]:

	ListingKey	ListingNumber	ListingCreationDate	CreditGrade	Term	LoanStatus	ClosedDate	BorrowerAPR	BorrowerRate	LenderYield	...	LP_ServiceFees	LP_CollectionFees
0	1021339766868145413AB3B	193129	2007-08-26 19:09:29.263000000	C	36	Completed	2009-08-14 00:00:00	0.16516	0.1580	0.1380	...	-133.18	0.0
1	1027360249950330 8B223C1	1209647	2014-02-27 08:28:07.900000000	NaN	36	Current	NaN	0.12016	0.0920	0.0820	...	0.00	0.0
2	0EE9337825851032864889A	81716	2007-01-05 15:00:47.090000000	HR	36	Completed	2009-12-17 00:00:00	0.28269	0.2750	0.2400	...	-24.20	0.0
3	0EF5356002482715299901A	658116	2012-10-22 11:02:35.010000000	NaN	36	Current	NaN	0.12528	0.0974	0.0874	...	-108.01	0.0
4	0F023589499656230C5E3E2	909464	2013-09-14 18:38:39.097000000	NaN	36	Current	NaN	0.24614					
5	0F05359734824199381F61D	1074836	2013-12-14 08:26:37.093000000	NaN	60	Current	NaN	0.15425					
6	0F0A3576754255009D63151	750899	2013-04-12 09:52:56.147000000	NaN	36	Current	NaN	0.31032					
7	0F1035772717087366F9EA7	768193	2013-05-05 06:49:27.493000000	NaN	36	Current	NaN	0.23939					
8	0F043596202561788EA13D5	1023355	2013-12-02 10:43:39.117000000	NaN	36	Current	NaN	0.07620					
9	0F043596202561788EA13D5	1023355	2013-12-02 10:43:39.117000000	NaN	36	Current	NaN	0.07620					

```
4        0.2085       0.1985    ...             -60.27                0.0
5        0.1314       0.1214    ...             -25.33                0.0
6        0.2712       0.2612    ...             -22.95                0.0
7        0.2019       0.1919    ...             -69.21                0.0
8        0.0629       0.0529    ...             -16.77                0.0
9        0.0629       0.0529    ...             -16.77                0.0

   LP_GrossPrincipalLoss  LP_NetPrincipalLoss LP_NonPrincipalRecoverypayments  \
0                    0.0                  0.0                              0.0
1                    0.0                  0.0                              0.0
2                    0.0                  0.0                              0.0
3                    0.0                  0.0                              0.0
4                    0.0                  0.0                              0.0
5                    0.0                  0.0                              0.0
6                    0.0                  0.0                              0.0
7                    0.0                  0.0                              0.0
8                    0.0                  0.0                              0.0
9                    0.0                  0.0                              0.0

   PercentFunded  Recommendations InvestmentFromFriendsCount  \
0            1.0                0                           0
1            1.0                0                           0
2            1.0                0                           0
3            1.0                0                           0
4            1.0                0                           0
5            1.0                0                           0
6            1.0                0                           0
7            1.0                0                           0
8            1.0                0                           0
9            1.0                0                           0

  InvestmentFromFriendsAmount Investors
0                         0.0       258
1                         0.0         1
2                         0.0        41
3                         0.0       158
4                         0.0        20
5                         0.0         1
6                         0.0         1
7                         0.0         1
8                         0.0         1
9                         0.0         1

[10 rows x 81 columns]
```

1.4.4 4-variables of interest

```
In [6]: columns = ['LoanKey', 'Term', 'LoanStatus', 'BorrowerAPR', 'BorrowerRate', 'ListingCateg
                   'BorrowerState', 'Occupation', 'EmploymentStatus', 'LoanOriginalAmount', 'Cre
                   'CreditScoreRangeUpper', 'DebtToIncomeRatio', 'Investors', 'StatedMonthlyInco
                   'IncomeVerifiable', 'ProsperRating (Alpha)','Recommendations']
        loan_clean = loan[columns]

In [7]: loan_clean.head(10)
```

Out[7]:

	LoanKey	Term	LoanStatus	BorrowerAPR	BorrowerRate
0	E33A3400205839220442E84	36	Completed	0.16516	0.1580
1	9E3B37071505919926B1D82	36	Current	0.12016	0.0920
2	6954337960046817851BCB2	36	Completed	0.28269	0.2750
3	A0393664465886295619C51	36	Current	0.12528	0.0974
4	A1803693021888892006 89E	36	Current	0.24614	0.2085
5	C3D63702273952547E79520	60	Current	0.15425	0.1314
6	CE963680102927767790520	36	Current	0.31032	0.2712
7	0C87368108902149313D53B	36	Current	0.23939	0.2019
8	02163700809231365A56A1C	36	Current	0.07620	0.0629
9	02163700809231365A56A1C	36	Current	0.07620	0.0629

	ListingCategory (numeric)	BorrowerState	Occupation	EmploymentStatus
0	0	CO	Other	Self-employed
1	2	CO	Professional	Employed
2	0	GA	Other	Not available
3	16	GA	Skilled Labor	Employed
4	2	MN	Executive	Employed
5	1	NM	Professional	Employed
6	1	KS	Sales - Retail	Employed
7	2	CA	Laborer	Employed
8	7	IL	Food Service	Employed
9	7	IL	Food Service	Employed

	LoanOriginalAmount	CreditScoreRangeLower	CreditScoreRangeUpper
0	9425	640.0	659.0
1	10000	680.0	699.0
2	3001	480.0	499.0
3	10000	800.0	819.0
4	15000	680.0	699.0
5	15000	740.0	759.0
6	3000	680.0	699.0
7	10000	700.0	719.0
8	10000	820.0	839.0
9	10000	820.0	839.0

	DebtToIncomeRatio	Investors	StatedMonthlyIncome	MonthlyLoanPayment
0	0.17	258	3083.333333	330.43
1	0.18	1	6125.000000	318.93

2	0.06	41	2083.333333	123.32
3	0.15	158	2875.000000	321.45
4	0.26	20	9583.333333	563.97
5	0.36	1	8333.333333	342.37
6	0.27	1	2083.333333	122.67
7	0.24	1	3355.750000	372.60
8	0.25	1	3333.333333	305.54
9	0.25	1	3333.333333	305.54

	IncomeVerifiable	ProsperRating (Alpha)	Recommendations
0	True	NaN	0
1	True	A	0
2	True	NaN	0
3	True	A	0
4	True	D	0
5	True	B	0
6	True	E	0
7	True	C	0
8	True	AA	0
9	True	AA	0

1.4.5 5-dupllicated value

In [8]:
```
loan_clean.duplicated().sum()
```

Out[8]: 871

In [9]:
```
loan_clean.drop_duplicates()
```

Out[9]:

	LoanKey	Term	LoanStatus	BorrowerAPR	\
0	E33A3400205839220442E84	36	Completed	0.16516	
1	9E3B37071505919926B1D82	36	Current	0.12016	
2	6954337960046817851BCB2	36	Completed	0.28269	
3	A0393664465886295619C51	36	Current	0.12528	
4	A1803693021888892006 89E	36	Current	0.24614	
5	C3D63702273952547E79520	60	Current	0.15425	
6	CE963680102927767790520	36	Current	0.31032	
7	0C8736810890214931 3D53B	36	Current	0.23939	
8	02163700809231365A56A1C	36	Current	0.07620	
10	7C0836512699736124 60D6D	60	Current	0.27462	
11	F375340302234633830A957	36	Completed	0.15033	
12	209F3701889728853CD17F7	36	Past Due (1-15 days)	0.17969	
13	C7F9368882589826293 56F0	36	Current	0.13138	
14	2BEF3682506622112EC4790	60	Current	0.11695	
15	3EE2364952142596779635D	36	Defaulted	0.35797	
16	2C78368704199057024A715	60	Current	0.30748	
17	51453366538336630763636	36	Chargedoff	0.13202	
18	BC2D367678091765710DFF4	36	Current	0.12528	
19	A02836960150183071E32AD	60	Current	0.24754	

20	CF023700237006812664 3CF	36	Current	0.16732
21	AAE334074114667 42AA4570	36	Completed	0.21488
22	8D313674169912018750992	36	Current	0.35356
23	D0623679715048926AB9F4D	36	Defaulted	0.28032
24	AE413701157050387A7B5B7	36	Current	0.19859
25	E2733695363511227B7135C	36	Current	0.30182
26	3B763675825568665C5122A	60	Completed	0.30748
27	7E583591759296638A02214	36	Completed	0.11296
28	DE4A3697562098725B01EB1	36	Current	0.20268
29	A8B83704317372651543A02	36	Current	0.15223
30	FBCE36430983505912FD996	36	Completed	0.12782
...	...	...	...	...
113907	2B4236970977154499F0DBB	36	Current	0.35356
113908	F6303605142921373247215	36	Completed	0.35858
113909	57403661039471901E1A063	60	Current	0.27554
113910	C5C83703728217652520B9D	36	Current	0.32446
113911	0EA3370036057406813973D	36	Current	0.25330
113912	116E3701974186089374F18	36	Current	0.06726
113913	D5BD3586476598829344AEB	36	Completed	0.12410
113914	063D3366920498906816DA9	36	Defaulted	0.25757
113915	7BC73427049986192BAE704	36	Completed	0.22237
113916	54F136678579880763AA200	36	Current	0.33286
113917	7340369916433687 16AB3D5	36	Current	0.32446
113918	D97D342416183363929B094	36	Defaulted	0.12201
113919	036E3681681503392E3E2AE	36	Current	0.27285
113920	D23D3684001395209B5030A	36	Current	0.35356
113921	30FD3365652573455326F15	36	Completed	NaN
113922	58A834282284173163C9D9D	36	Completed	0.15094
113923	BEA034319304168134 03CE7	36	Completed	0.22378
113924	5AD13664710145110F93CD8	60	Current	0.17317
113925	B5FC3682174533953146478	36	Current	0.31032
113926	E5F33364419370827F04C4C	36	Defaulted	0.29776
113927	895E341956005398355C384	36	Completed	0.07469
113928	73D936216341471895FF2FC	36	Completed	0.22362
113929	F8973687907243662215A6F	36	Completed	0.30285
113930	AF80368651203735984C668	36	Current	0.20053
113931	2AFF3704413774725AD8BAF	60	Current	0.15016
113932	9BD7367919051593140DB62	36	Current	0.22354
113933	62D93634569816897D5A276	36	FinalPaymentInProgress	0.13220
113934	DD1A370200396006300ACA0	60	Current	0.23984
113935	589536350469116027ED11B	60	Completed	0.28408
113936	00AF3704550953269A64E40	36	Current	0.13189

	BorrowerRate	ListingCategory (numeric)	BorrowerState	\
0	0.1580	0	CO	
1	0.0920	2	CO	
2	0.2750	0	GA	
3	0.0974	16	GA	

4	0.2085	2	MN
5	0.1314	1	NM
6	0.2712	1	KS
7	0.2019	2	CA
8	0.0629	7	IL
10	0.2489	1	MD
11	0.1325	0	NaN
12	0.1435	1	AL
13	0.1034	1	AZ
14	0.0949	1	VA
15	0.3177	13	FL
16	0.2809	6	CA
17	0.1250	0	NaN
18	0.0974	1	PA
19	0.2225	1	OR
20	0.1314	1	MN
21	0.2075	0	MI
22	0.3134	1	NY
23	0.2419	15	IL
24	0.1620	1	LA
25	0.2629	15	CA
26	0.2809	1	NY
27	0.0920	1	CO
28	0.1660	2	PA
29	0.1239	1	LA
30	0.0999	20	CA
...	...	...	...
113907	0.3134	1	AZ
113908	0.3220	3	NY
113909	0.2498	2	FL
113910	0.2850	1	TX
113911	0.2155	14	MD
113912	0.0605	2	MN
113913	0.1030	1	WA
113914	0.2500	0	CA
113915	0.2000	7	MA
113916	0.2932	13	FL
113917	0.2850	1	OH
113918	0.1080	4	CA
113919	0.2346	2	IL
113920	0.3134	1	NJ
113921	0.0400	0	NaN
113922	0.1295	5	GA
113923	0.2089	1	MS
113924	0.1499	13	WA
113925	0.2712	1	NY
113926	0.2900	0	CA
113927	0.0679	4	WA

113928	0.1899	3	CO
113929	0.2639	2	FL
113930	0.1639	1	IN
113931	0.1274	3	IL
113932	0.1864	1	IL
113933	0.1110	7	PA
113934	0.2150	1	TX
113935	0.2605	2	GA
113936	0.1039	1	NY

	Occupation	EmploymentStatus	LoanOriginalAmount	\
0	Other	Self-employed	9425	
1	Professional	Employed	10000	
2	Other	Not available	3001	
3	Skilled Labor	Employed	10000	
4	Executive	Employed	15000	
5	Professional	Employed	15000	
6	Sales - Retail	Employed	3000	
7	Laborer	Employed	10000	
8	Food Service	Employed	10000	
10	Fireman	Employed	13500	
11	Waiter/Waitress	Full-time	1000	
12	Sales - Retail	Employed	4000	
13	Construction	Employed	8500	
14	Computer Programmer	Employed	19330	
15	Other	Other	4000	
16	Professional	Full-time	4000	
17	Professional	Not available	10000	
18	Sales - Commission	Employed	15000	
19	Laborer	Employed	6500	
20	Retail Management	Employed	14000	
21	Professional	Full-time	3000	
22	Other	Other	4000	
23	Skilled Labor	Employed	2000	
24	Other	Employed	4000	
25	Engineer - Mechanical	Employed	4000	
26	Sales - Commission	Employed	4000	
27	Executive	Full-time	4000	
28	Military Enlisted	Employed	10000	
29	Other	Employed	35000	
30	Other	Employed	10000	
...	...	...	...	
113907	Sales - Retail	Employed	4000	
113908	Sales - Commission	Employed	7500	
113909	Clerical	Employed	6000	
113910	Executive	Employed	4000	
113911	Other	Employed	10000	
113912	Scientist	Employed	4000	

113913	Analyst	Full-time	8000	
113914	NaN	NaN	3000	
113915	Other	Full-time	3000	
113916	Professional	Employed	4000	
113917	Clerical	Employed	4000	
113918	Social Worker	Full-time	7000	
113919	Other	Other	4000	
113920	Retail Management	Employed	4000	
113921	NaN	NaN	1000	
113922	Other	Full-time	5000	
113923	Clergy	Full-time	8000	
113924	Other	Employed	5000	
113925	Homemaker	Employed	4000	
113926	Other	Not available	3000	
113927	Executive	Full-time	4292	
113928	Other	Full-time	2000	
113929	Accountant/CPA	Employed	2500	
113930	Professional	Employed	3000	
113931	Analyst	Employed	25000	
113932	Food Service Management	Employed	10000	
113933	Professional	Employed	2000	
113934	Other	Employed	10000	
113935	Food Service	Full-time	15000	
113936	Professor	Employed	2000	

	CreditScoreRangeLower	CreditScoreRangeUpper	DebtToIncomeRatio	\
0	640.0	659.0	0.17000	
1	680.0	699.0	0.18000	
2	480.0	499.0	0.06000	
3	800.0	819.0	0.15000	
4	680.0	699.0	0.26000	
5	740.0	759.0	0.36000	
6	680.0	699.0	0.27000	
7	700.0	719.0	0.24000	
8	820.0	839.0	0.25000	
10	640.0	659.0	0.12000	
11	640.0	659.0	0.27000	
12	680.0	699.0	0.18000	
13	740.0	759.0	0.09000	
14	740.0	759.0	0.20000	
15	700.0	719.0	0.49000	
16	640.0	659.0	0.15000	
17	760.0	779.0	0.12000	
18	740.0	759.0	0.24000	
19	680.0	699.0	0.41000	
20	660.0	679.0	0.20000	
21	620.0	639.0	0.09000	
22	700.0	719.0	9.20000	

23	680.0	699.0	0.39000
24	660.0	679.0	0.16000
25	680.0	699.0	0.12000
26	660.0	679.0	0.11000
27	700.0	719.0	0.26000
28	720.0	739.0	0.12000
29	740.0	759.0	0.32000
30	740.0	759.0	0.11000
...	...	...	...
113907	640.0	659.0	0.40000
113908	700.0	719.0	NaN
113909	800.0	819.0	0.24000
113910	640.0	659.0	0.18000
113911	660.0	679.0	0.29000
113912	800.0	819.0	0.20000
113913	780.0	799.0	0.25000
113914	520.0	539.0	0.05000
113915	620.0	639.0	0.26000
113916	660.0	679.0	0.15000
113917	660.0	679.0	0.40000
113918	740.0	759.0	0.39000
113919	680.0	699.0	0.22000
113920	740.0	759.0	0.25000
113921	NaN	NaN	0.23284
113922	640.0	659.0	0.18000
113923	700.0	719.0	0.57000
113924	640.0	659.0	0.17000
113925	680.0	699.0	0.28000
113926	540.0	559.0	0.07000
113927	760.0	779.0	0.06000
113928	740.0	759.0	0.27000
113929	660.0	679.0	0.05000
113930	680.0	699.0	0.20000
113931	800.0	819.0	0.28000
113932	700.0	719.0	0.13000
113933	700.0	719.0	0.11000
113934	700.0	719.0	0.51000
113935	680.0	699.0	0.48000
113936	680.0	699.0	0.23000

	Investors	StatedMonthlyIncome	MonthlyLoanPayment	IncomeVerifiable	\
0	258	3083.333333	330.43	True	
1	1	6125.000000	318.93	True	
2	41	2083.333333	123.32	True	
3	158	2875.000000	321.45	True	
4	20	9583.333333	563.97	True	
5	1	8333.333333	342.37	True	
6	1	2083.333333	122.67	True	

7	1	3355.750000	372.60	True
8	1	3333.333333	305.54	True
10	19	7500.000000	395.37	True
11	53	1666.666667	33.81	True
12	1	2416.666667	137.39	True
13	171	5833.333333	275.63	True
14	371	10833.333333	415.37	True
15	10	5500.000000	173.71	True
16	8	8291.666667	124.76	True
17	85	5833.333333	334.54	True
18	303	6250.000000	482.18	True
19	1	3075.000000	180.45	True
20	1	5166.666667	472.66	True
21	53	3750.000000	112.64	True
22	94	118.333333	172.76	True
23	30	2500.000000	78.67	True
24	1	2333.333333	141.02	True
25	3	6974.000000	161.78	True
26	37	3885.916667	124.76	True
27	121	6666.666667	0.00	True
28	1	3600.000000	354.54	True
29	1	10416.666667	1169.03	True
30	30	3750.000000	322.62	True
...	...	...	...	...
113907	3	2166.666667	172.76	True
113908	98	2833.333333	327.49	False
113909	9	3333.333333	176.04	True
113910	1	18756.000000	166.54	True
113911	1	3333.333333	379.58	True
113912	1	2500.000000	121.78	True
113913	265	9750.000000	259.27	True
113914	1	2400.000000	119.28	True
113915	135	4416.666667	1.57	True
113916	87	4583.333333	168.32	True
113917	1	1916.666667	166.54	True
113918	173	5583.333333	228.51	True
113919	1	2500.000000	155.80	True
113920	1	5208.333333	172.76	True
113921	1	12500.000000	29.52	True
113922	145	5250.000000	168.35	True
113923	270	3966.666667	300.95	True
113924	83	3208.333333	118.92	True
113925	1	2333.333333	163.56	True
113926	39	5416.666667	125.72	True
113927	194	10333.333333	132.11	True
113928	25	2333.333333	73.30	True
113929	26	4333.333333	101.25	True
113930	52	6250.000000	106.05	True

113931	1	8146.666667	565.50	True
113932	1	4333.333333	364.74	True
113933	22	8041.666667	65.57	True
113934	119	2875.000000	273.35	True
113935	274	3875.000000	449.55	True
113936	1	4583.333333	64.90	True

	ProsperRating (Alpha)	Recommendations
0	NaN	0
1	A	0
2	NaN	0
3	A	0
4	D	0
5	B	0
6	E	0
7	C	0
8	AA	0
10	C	0
11	NaN	0
12	B	0
13	A	0
14	A	0
15	HR	0
16	E	0
17	NaN	0
18	A	0
19	D	0
20	B	0
21	NaN	0
22	HR	0
23	D	0
24	C	0
25	E	0
26	E	0
27	A	0
28	C	0
29	A	0
30	A	0
...	...	...
113907	HR	0
113908	E	0
113909	C	0
113910	E	0
113911	D	0
113912	AA	0
113913	A	0
113914	NaN	0
113915	NaN	1

```
113916                    E                0
113917                    E                0
113918                  NaN                0
113919                    D                0
113920                   HR                0
113921                  NaN                0
113922                  NaN                0
113923                  NaN                0
113924                    A                1
113925                    E                0
113926                  NaN                0
113927                  NaN                2
113928                    C                0
113929                    E                0
113930                    B                0
113931                    B                0
113932                    C                0
113933                    A                0
113934                    D                0
113935                    C                0
113936                    A                0

[113066 rows x 19 columns]
```

1.4.6 6-missing value

```
In [10]: loan_clean.isnull().sum()
```

```
Out[10]: LoanKey                         0
         Term                            0
         LoanStatus                      0
         BorrowerAPR                    25
         BorrowerRate                    0
         ListingCategory (numeric)       0
         BorrowerState                5515
         Occupation                   3588
         EmploymentStatus             2255
         LoanOriginalAmount              0
         CreditScoreRangeLower         591
         CreditScoreRangeUpper         591
         DebtToIncomeRatio            8554
         Investors                       0
         StatedMonthlyIncome             0
         MonthlyLoanPayment              0
         IncomeVerifiable                0
         ProsperRating (Alpha)       29084
         Recommendations                 0
         dtype: int64
```

```
In [11]: loan_clean.BorrowerState.value_counts()
```

```
Out[11]: CA    14717
         TX     6842
         NY     6729
         FL     6720
         IL     5921
         GA     5008
         OH     4197
         MI     3593
         VA     3278
         NJ     3097
         NC     3084
         WA     3048
         PA     2972
         MD     2821
         MO     2615
         MN     2318
         MA     2242
         CO     2210
         IN     2078
         AZ     1901
         WI     1842
         OR     1817
         TN     1737
         AL     1679
         CT     1627
         SC     1122
         NV     1090
         KS     1062
         KY      983
         OK      971
         LA      954
         UT      877
         AR      855
         MS      787
         NE      674
         ID      599
         NH      551
         NM      472
         RI      435
         HI      409
         WV      391
         DC      382
         MT      330
         DE      300
         VT      207
         AK      200
```

```
SD      189
IA      186
WY      150
ME      101
ND       52
Name: BorrowerState, dtype: int64
```

```
In [12]: loan_clean.Occupation.value_counts()
```

```
Out[12]: Other                                28617
         Professional                         13628
         Computer Programmer                   4478
         Executive                             4311
         Teacher                               3759
         Administrative Assistant              3688
         Analyst                               3602
         Sales - Commission                    3446
         Accountant/CPA                        3233
         Clerical                              3164
         Sales - Retail                        2797
         Skilled Labor                         2746
         Retail Management                     2602
         Nurse (RN)                            2489
         Construction                          1790
         Truck Driver                          1675
         Laborer                               1595
         Police Officer/Correction Officer     1578
         Civil Service                         1457
         Engineer - Mechanical                 1406
         Military Enlisted                     1272
         Food Service Management               1239
         Engineer - Electrical                 1125
         Food Service                          1123
         Medical Technician                    1117
         Attorney                              1046
         Tradesman - Mechanic                   951
         Social Worker                          741
         Postal Service                         627
         Professor                              557
                                               ...
         Scientist                              372
         Military Officer                       346
         Bus Driver                             316
         Principal                              312
         Teacher's Aide                         276
         Pharmacist                             257
         Student - College Graduate Student     245
         Landscaping                            236
```

```
Engineer - Chemical                     225
Investor                                214
Architect                               213
Pilot - Private/Commercial              199
Clergy                                  196
Student - College Senior                188
Car Dealer                              180
Chemist                                 145
Psychologist                            145
Biologist                               125
Religious                               124
Flight Attendant                        123
Homemaker                               120
Tradesman - Carpenter                   120
Student - College Junior                112
Tradesman - Plumber                     102
Student - College Sophomore              69
Dentist                                  68
Student - College Freshman               41
Student - Community College              28
Judge                                    22
Student - Technical School               16
Name: Occupation, Length: 67, dtype: int64
```

```
In [13]: loan_clean.EmploymentStatus.value_counts()
```

```
Out[13]: Employed         67322
         Full-time        26355
         Self-employed     6134
         Not available     5347
         Other             3806
         Part-time         1088
         Not employed       835
         Retired            795
         Name: EmploymentStatus, dtype: int64
```

```
In [14]: loan_clean.rename(columns={'ProsperRating (Alpha)': 'ProsperRating_Alpha'},inplace=True
         loan_clean.ProsperRating_Alpha.value_counts()
```

```
Out[14]: C     18345
         B     15581
         A     14551
         D     14274
         E      9795
         HR     6935
         AA     5372
         Name: ProsperRating_Alpha, dtype: int64
```

1.4.7 6-Manage the empty value

```
In [15]: loan_clean.Occupation.fillna('Other', inplace=True)
         loan_clean.EmploymentStatus.fillna('Other', inplace=True)
         loan_clean=loan_clean.dropna()

In [16]: loan_clean.info()

<class 'pandas.core.frame.DataFrame'>
Int64Index: 77557 entries, 1 to 113936
Data columns (total 19 columns):
LoanKey                      77557 non-null object
Term                         77557 non-null int64
LoanStatus                   77557 non-null object
BorrowerAPR                  77557 non-null float64
BorrowerRate                 77557 non-null float64
ListingCategory (numeric)    77557 non-null int64
BorrowerState                77557 non-null object
Occupation                   77557 non-null object
EmploymentStatus             77557 non-null object
LoanOriginalAmount           77557 non-null int64
CreditScoreRangeLower        77557 non-null float64
CreditScoreRangeUpper        77557 non-null float64
DebtToIncomeRatio            77557 non-null float64
Investors                    77557 non-null int64
StatedMonthlyIncome          77557 non-null float64
MonthlyLoanPayment           77557 non-null float64
IncomeVerifiable             77557 non-null bool
ProsperRating_Alpha          77557 non-null object
Recommendations              77557 non-null int64
dtypes: bool(1), float64(7), int64(5), object(6)
memory usage: 11.3+ MB


In [17]: loan_clean.rename(columns={'ListingCategory (numeric)': 'ListingCategory_numeric'},inpl

In [18]: loan_clean.shape

Out[18]: (77557, 19)
```

1.4.8 What is the structure of your dataset?

The cleaned dataset consists of 77557 loan and 19 features

1.4.9 What is/are the main feature(s) of interest in your dataset?

the main feature for us is the LoanStatus it will be interest to now what affect the LonStatus

1.4.10 What features in the dataset do you think will help support your investigation into your feature(s) of interest?

the features in dataset that we think will help investigation of feature of interest are:

- Term
- LoanStatus
- BorrowerAPR
- BorrowerRate
- ListingCategory_numeric
- BorrowerState
- Occupation
- EmploymentStatus
- LoanOriginalAmount
- CreditScoreRangeLower
- CreditScoreRangeUpper
- DebtToIncomeRatio
- Investors
- StatedMonthlyIncome
- MonthlyLoanPayment
- IncomeVerifiable
- ProsperRating_Alpha
- Recommendations

1.5 Univariate Exploration

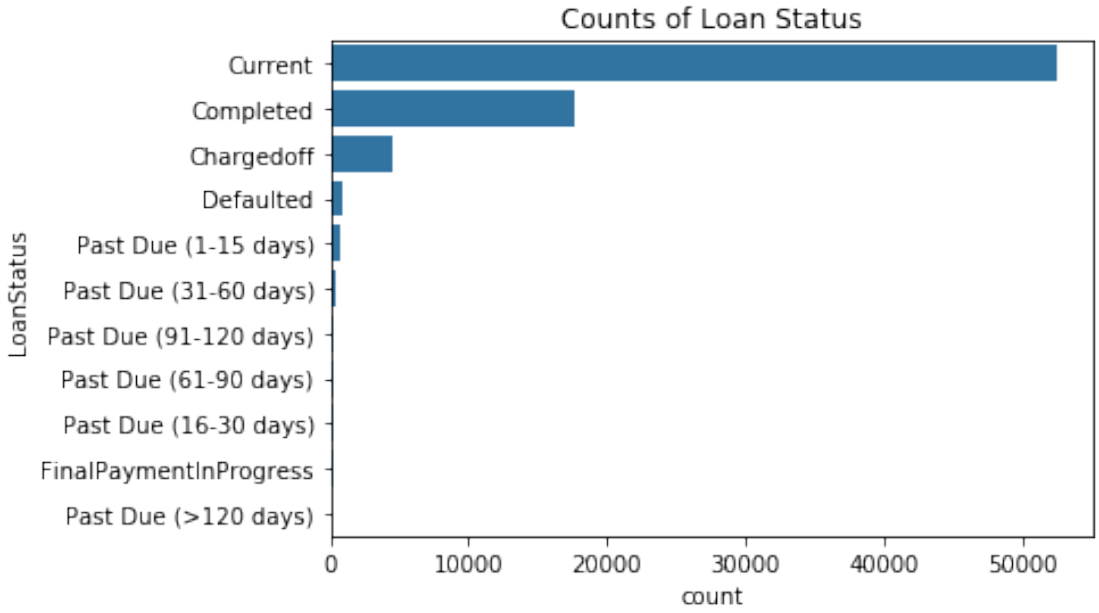
```
In [19]: #### a function to plot countplot
         def plot(dataset,yfeature,order,color,title):
             sb.countplot(data = dataset, y =yfeature, color=color, order=order)
             plt.ylabel(yfeature)
             plt.title(title)
             plt.show()
```

1.5.1 question 1

we firstly begin with our features of interest: what is the distribution of LoanStatus?

1.5.2 visualisation

```
In [20]: type_order = loan_clean['LoanStatus'].value_counts().index
         color=sb.color_palette()[0]
         plot(loan_clean,'LoanStatus',type_order,color,'Counts of Loan Status')
```

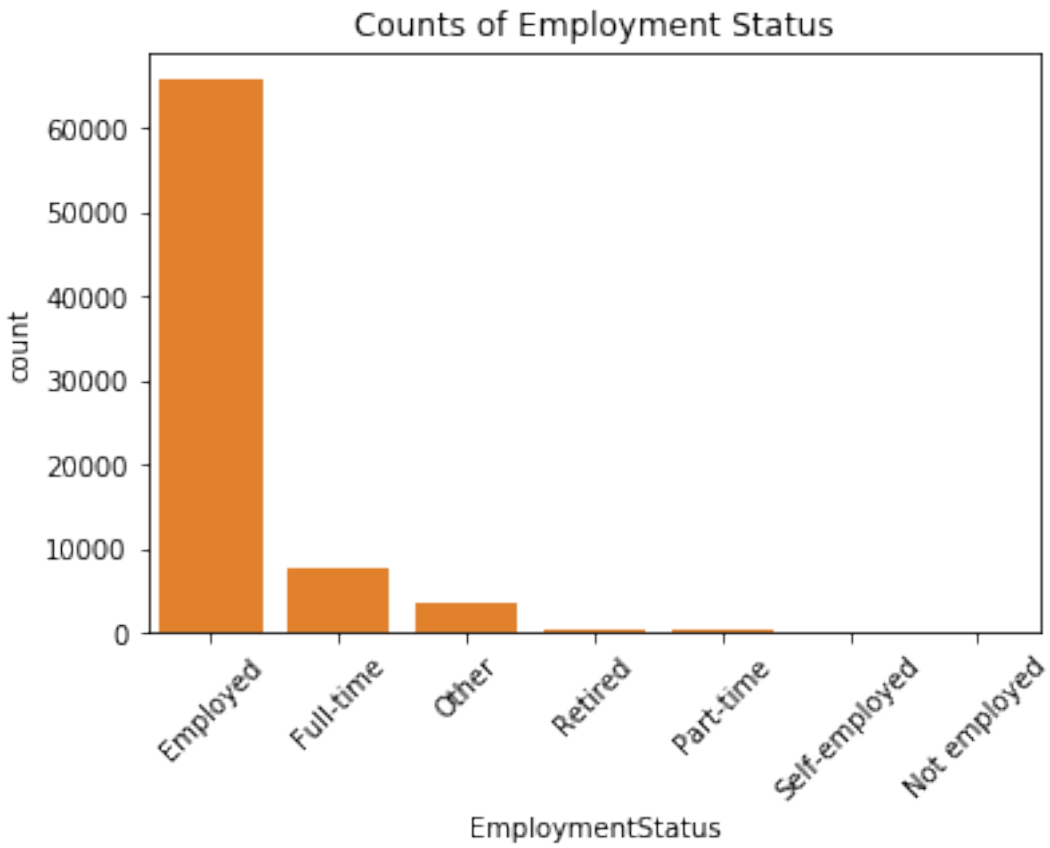


1.5.3 observation

from the distribution of LoanStatus feature we can observe that the Loan Status must hight is Current

1.5.4 question 2 what is the distribution of Employment Status

```
In [21]: type_order = loan_clean['EmploymentStatus'].value_counts().index
         sb.countplot(data = loan_clean, x ='EmploymentStatus', color=sb.color_palette()[1], ord
         plt.ylabel('count')
         plt.title('Counts of Employment Status')
         plt.xticks(rotation = 45)
         plt.show()
```



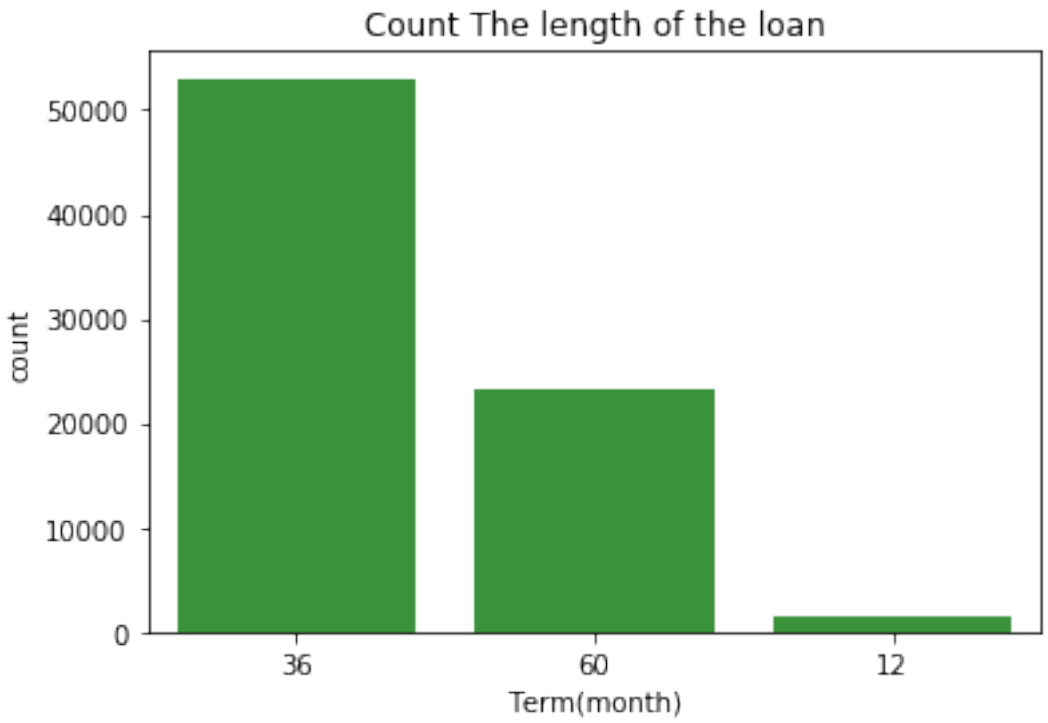
1.5.5 observation

the most borrowers are Employed

1.5.6 question 3

what is the distribution of Term loan?

```
In [22]: type_order = loan_clean['Term'].value_counts().index
         sb.countplot(data = loan_clean, x ='Term', color=sb.color_palette()[2], order=type_orde
         plt.ylabel('count')
         plt.xlabel('Term(month)')
         plt.title(' Count The length of the loan ')
         plt.show()
```

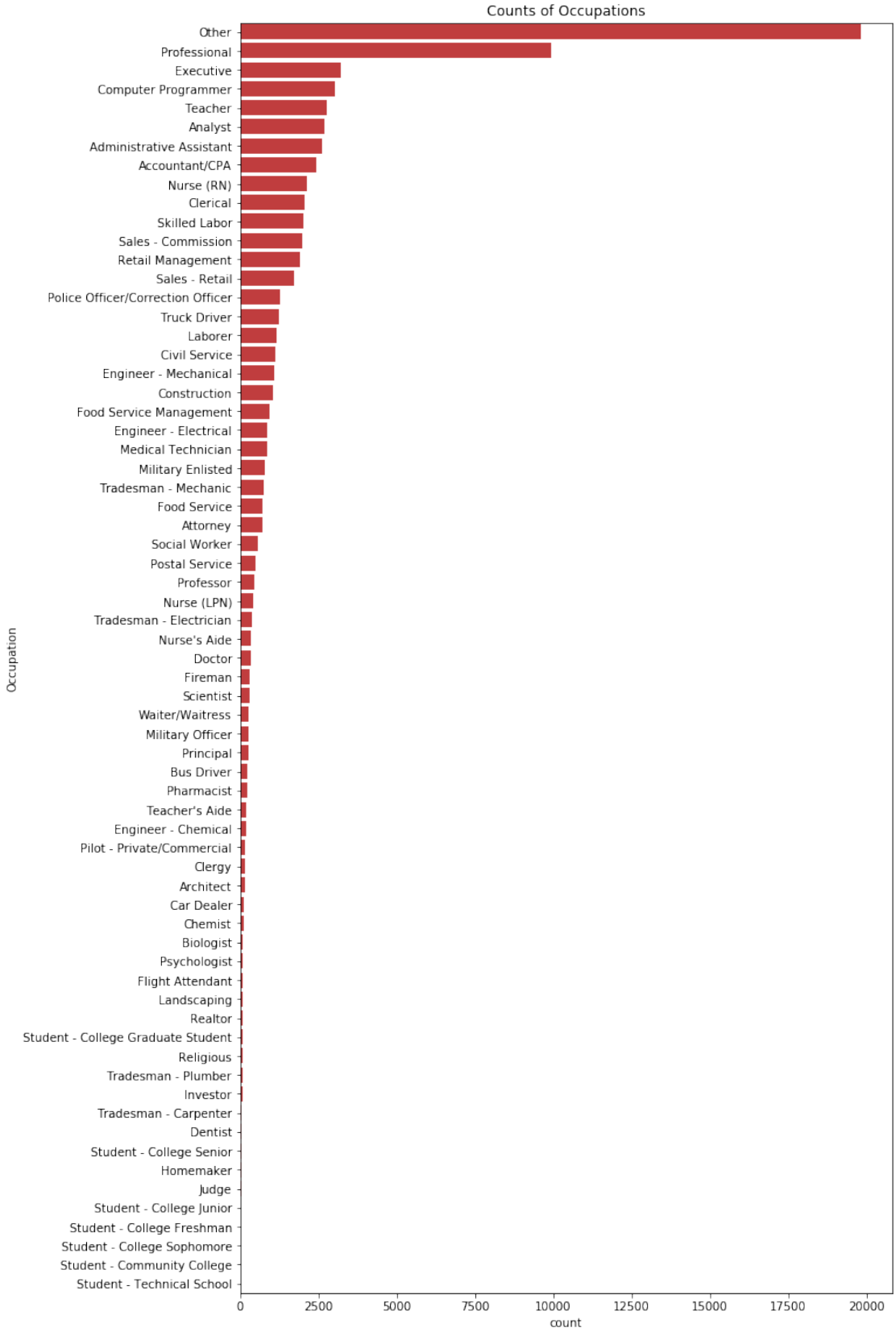


1.5.7 observation

the most length of the loan are 36 month

1.5.8 question 4 what is the distribution of ocupation of borrower?

```
In [23]: plt.subplots(figsize = [10,20])
         color=sb.color_palette()[3]
         type_order = loan_clean['Occupation'].value_counts().index
         plot(loan_clean,'Occupation',type_order,color,'Counts of Occupations')
```

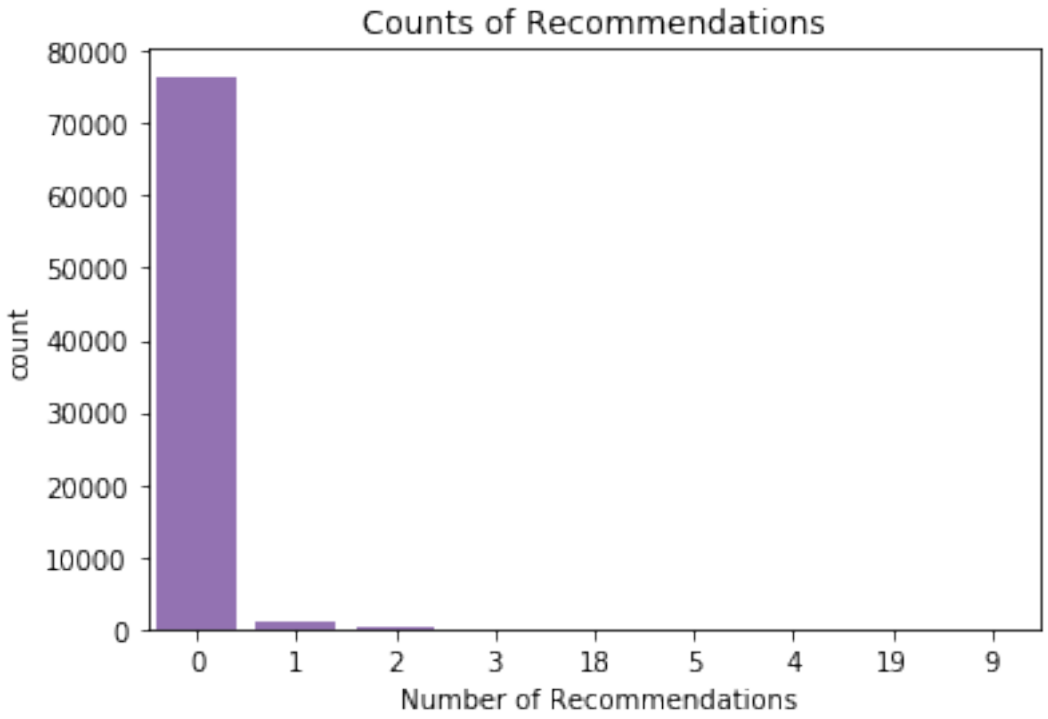

Counts of Occupations
Other
Professional
Executive
Computer Programmer
Teacher
Analyst
Administrative Assistant
Accountant/CPA
Nurse (RN)
Clerical
Skilled Labor
Sales - Commission
Retail Management
Sales - Retail
Police Officer/Correction Officer
Truck Driver
Laborer
Civil Service
Engineer - Mechanical
Construction
Food Service Management
Engineer - Electrical
Medical Technician
Military Enlisted
Tradesman - Mechanic
Food Service
Attorney
Social Worker
Postal Service
Professor
Nurse (LPN)
Tradesman - Electrician
Nurse's Aide
Doctor
Fireman
Scientist
Waiter/Waitress
Military Officer
Principal
Bus Driver
Pharmacist
Teacher's Aide
Engineer - Chemical
Pilot - Private/Commercial
Clergy
Architect
Car Dealer
Chemist
Biologist
Psychologist
Flight Attendant
Landscaping
Realtor
Student - College Graduate Student
Religious
Tradesman - Plumber
Investor
Tradesman - Carpenter
Dentist
Student - College Senior
Homemaker
Judge
Student - College Junior
Student - College Freshman
Student - College Sophomore
Student - Community College
Student - Technical School
Occupation
0
2500
5000
7500
10000
12500
15000
17500
20000
count

1.5.9 observation

because Other and professional which are the two first occupation not give us usefull information we ignore it and we observe that: executive, computer programmer, teacher ,analyst, administrative assistant are the five first occupation

1.5.10 question 5 what is recommandations distribution of borrower?

```
In [24]: order = loan_clean['Recommendations'].value_counts().index
         sb.countplot(data = loan_clean, x ='Recommendations', color=sb.color_palette()[4], orde
         plt.xlabel(' Number of Recommendations')
         plt.title('Counts of Recommendations')
         plt.show()
```



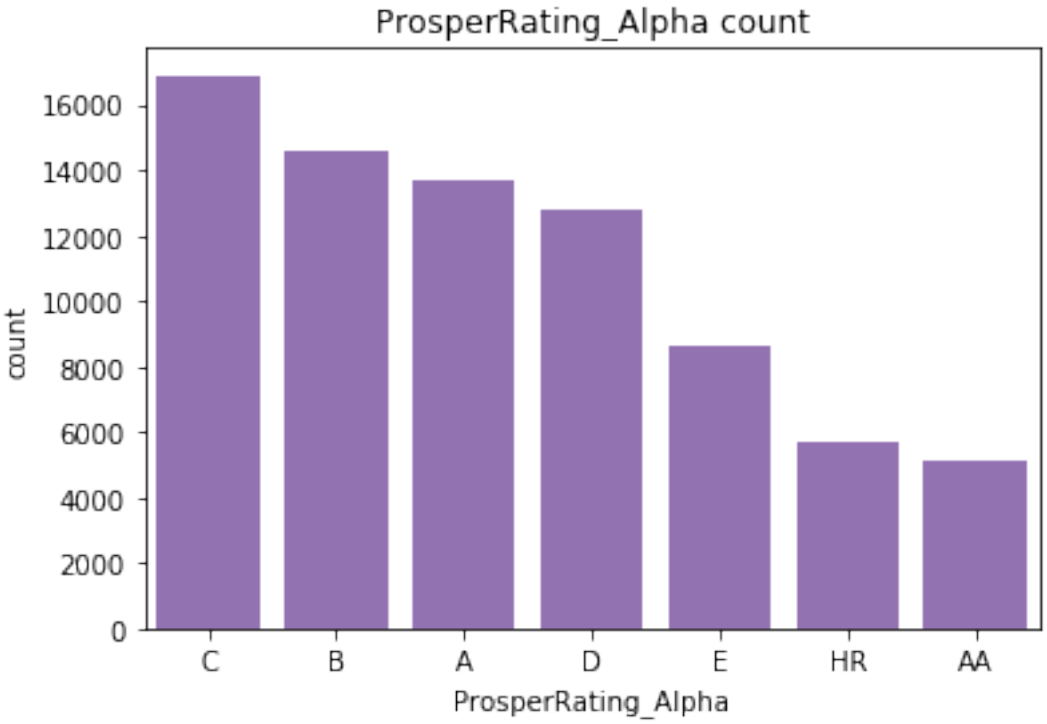
1.5.11 observation

we look that most of the borrowers have 0 recommandation at the time the listing was created.

1.5.12 question 6 what is the distribution rosperScore ?

```
In [25]: order =loan_clean['ProsperRating_Alpha'].value_counts().index
         sb.countplot(data=loan_clean, x='ProsperRating_Alpha', color=sb.color_palette()[4], ord
```

```
plt.xlabel('ProsperRating_Alpha');
plt.title('ProsperRating_Alpha count');
```



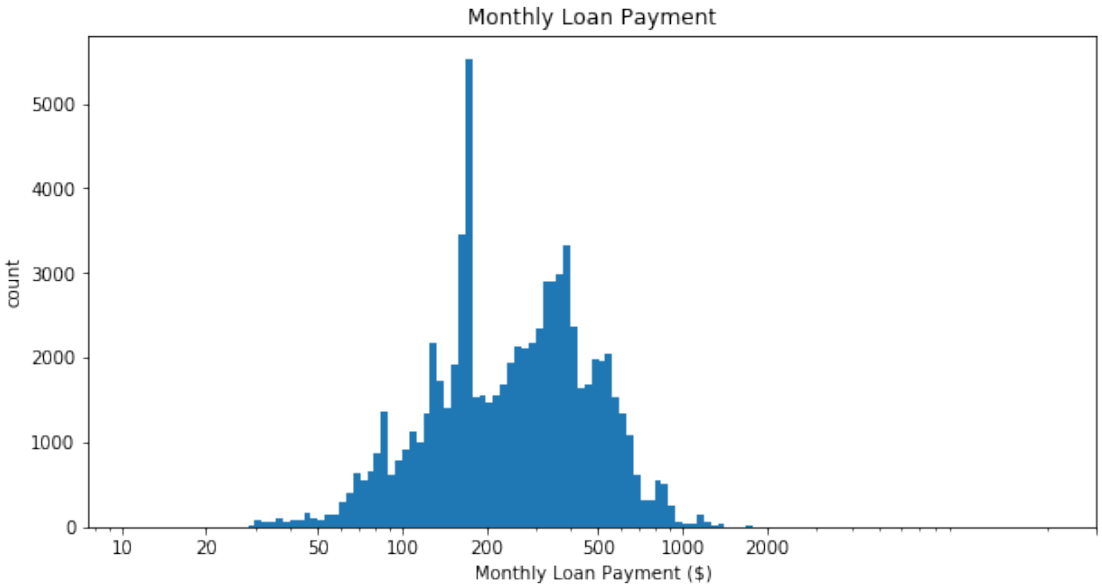
1.5.13 observation

we fine that the rating of most common borrowers is between C to D

1.5.14 question 7 what is the distribution of Monthly Loan Payment

```
In [26]: bins = 10 ** np.arange(1, np.log10(loan_clean['MonthlyLoanPayment'].max())+0.025, 0.025

         plt.figure(figsize=[10, 5])
         plt.hist(data = loan_clean, x = 'MonthlyLoanPayment', bins = bins)
         plt.xscale('log')
         plt.xticks([10, 20, 50, 100, 200, 500, 1000, 2000, 30000], ['10', '20','50', '100', '20
         plt.xlabel('Monthly Loan Payment ($)')
         plt.ylabel('count')
         plt.title('Monthly Loan Payment')
         plt.show()
```



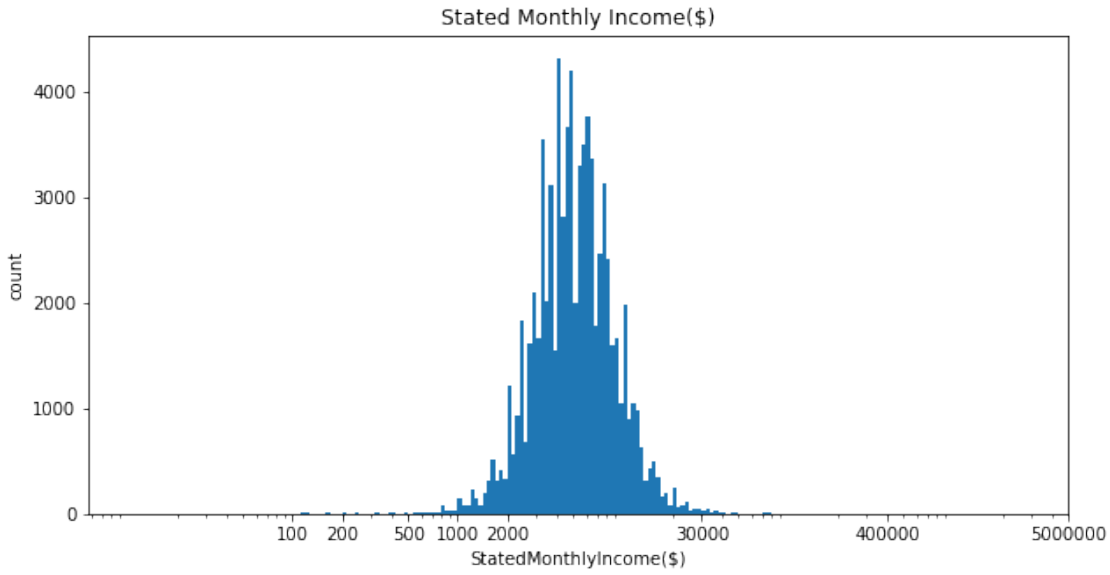
1.5.15 observation

we can see that the higths monthly payment are between 100 and 1000 but the most higth is between 100 and 200

1.5.16 question 8 what is the distribution of StatedMonthlyIncome?

```
In [27]: bins = 10 ** np.arange(1, np.log10(loan_clean['StatedMonthlyIncome'].max())+0.025, 0.02

         plt.figure(figsize=[10, 5])
         plt.hist(data = loan_clean, x = 'StatedMonthlyIncome', bins = bins)
         plt.xscale('log')
         plt.xticks([100, 200, 500, 1e3, 2e3, 3e4, 4e5, 5e6], ['100', '200', '500', '1000', '200
         plt.xlabel('StatedMonthlyIncome($)')
         plt.ylabel('count')
         plt.title('Stated Monthly Income($)')
         plt.show()
```



1.5.17 observation

the Stated Monthly Income of borrowers is most between 1000 and 30000

1.5.18 Discuss the distribution(s) of your variable(s) of interest. Were there any unusual points? Did you need to perform any transformations?

- the Stated Monthly Income of borrowers is most between 1000 and 30000
- we can see that the higths monthly payment are between 100 and 1000 but the most higth is between 100 and 200
- we fine that the rating of most common borrowers is between C to D
- we look that most of the borrowers have 0 recommandation at the time the listing was created.
- because Other and professional which are the two first occupation not give us usefull information we ignore it and we observe that: executive, computer programmer, teacher ,analyst, administrative assistant are the five first occupation
- the most length of the loan are 36 month
- the most borrowers are Employed
- from the distribution of LoanStatus feature we can observe that the Loan Status must hight is Current

1.5.19 Of the features you investigated, were there any unusual distributions? Did you perform any operations on the data to tidy, adjust, or change the form of the data? If so, why did you do this?

I have made any operation on my data after features investigqtion

1.6 Bivariate Exploration

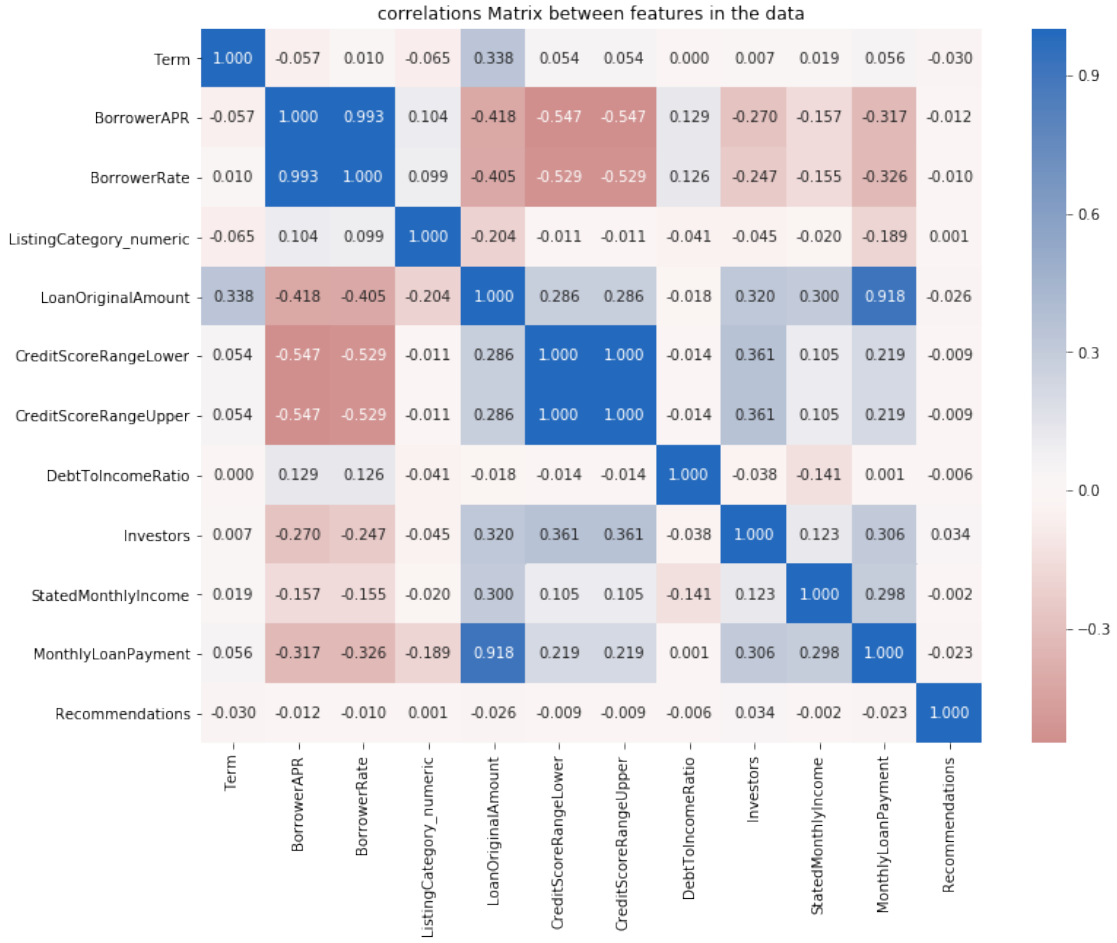
1.6.1 question 9 what is the correlations between pairwaises of numerics variables in the dataset

```
In [28]: numeric_var = loan_clean.select_dtypes(include='number').columns

In [29]: numeric_var

Out[29]: Index(['Term', 'BorrowerAPR', 'BorrowerRate', 'ListingCategory_numeric',
                'LoanOriginalAmount', 'CreditScoreRangeLower', 'CreditScoreRangeUpper',
                'DebtToIncomeRatio', 'Investors', 'StatedMonthlyIncome',
                'MonthlyLoanPayment', 'Recommendations'],
               dtype='object')

In [30]: plt.figure(figsize = [12, 9])
         sb.heatmap(loan_clean[numeric_var].corr(), annot = True, fmt = '.3f',
                    cmap = 'vlag_r', center = 0)
         plt.title("correlations Matrix between features in the data")
         plt.show()
```

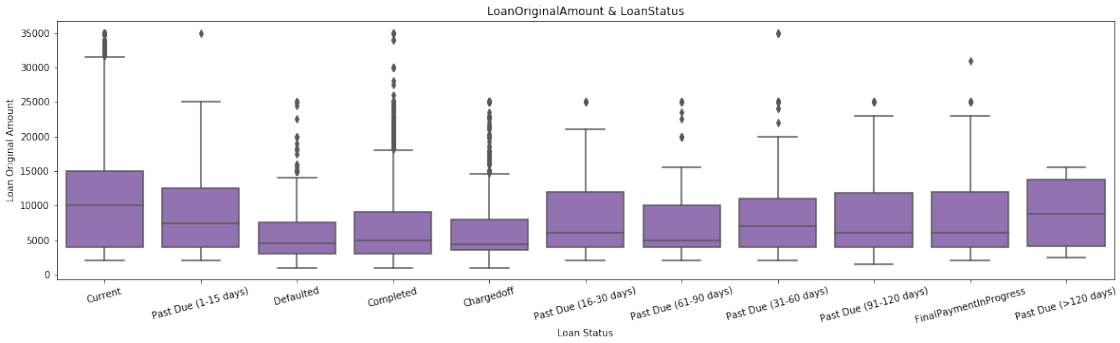


1.6.2 observation

we see strong positive correlation between creditScoreRangeLower and CreditScoreRangeUper, between LoanOrginalAmount and MonthtlyPayement, between BorrowerAPR and BorrowerRate

1.6.3 question 10 what is the relation between Loan Orginal Amount and Loan Statut?

```
In [31]: plt.figure(figsize = [20, 5])
         sb.boxplot(data=loan_clean,y='LoanOriginalAmount',x='LoanStatus',color=sb.color_palette
         plt.title('LoanOriginalAmount & LoanStatus');
         plt.ylabel('Loan Original Amount');
         plt.xlabel('Loan Status');
         plt.xticks(rotation=15);
```

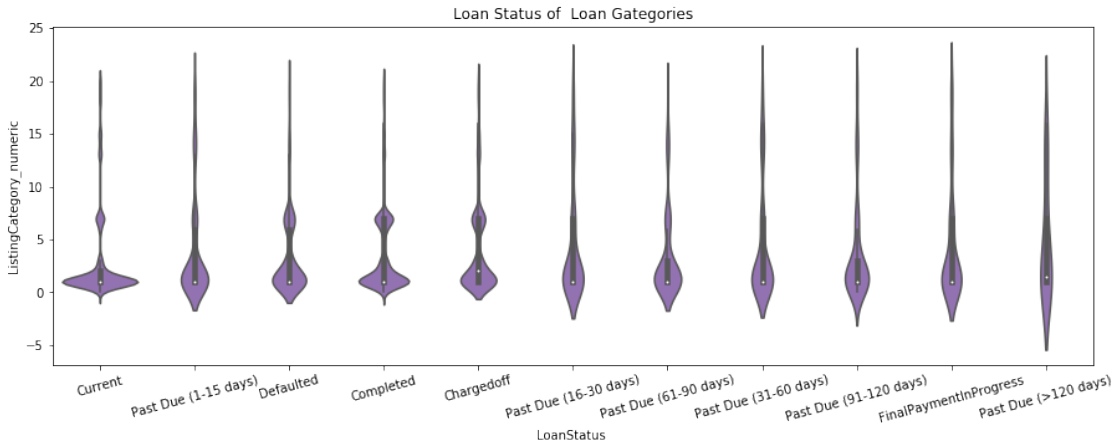


1.6.4 observation

we note that completed, defaulted and chargedoff have the lower IQR

1.6.5 question 11 what is the loan statut of loan categories?

```
In [32]: plt.figure(figsize = [15, 5])
         sb.violinplot(data=loan_clean,x='LoanStatus', y='ListingCategory_numeric',color=sb.colo
         plt.title('Loan Status of  Loan Gategories')
         plt.xlabel('LoanStatus ');
         plt.ylabel('ListingCategory_numeric');
         plt.xticks(rotation=15);
```



1.6.6 observation

we see that all Loan Statut have almost the same Listing Category

1.6.7 Talk about some of the relationships you observed in this part of the investigation. How did the feature(s) of interest vary with other features in the dataset?

- we see that all Loan Statut have almost the same Listing Category
- we note that defaulted and chargedoff have the lower IQR
- we see strong positive correlation between creditScoreRangeLower and CreditScoreRange-Uper, between LoanOrginalAmount and MonthtlyPayement, between BorrowerAPR and BorrowerRate

1.6.8 Did you observe any interesting relationships between the other features (not the main feature(s) of interest)?

we see strong positive correlation between creditScoreRangeLower and CreditScoreRangeUper, between LoanOrginalAmount and MonthtlyPayement, between BorrowerAPR and BorrowerRate

1.7 Multivariate Exploration

1.7.1 question 12 what correlation can we have between Loan Original Amount , Loan Original Amount and Stated Monthly Income?

```
In [ ]: plt.figure(figsize = [10, 5])
        plt.scatter(data=loan_clean,x='LoanOriginalAmount',y = 'Term',c='StatedMonthlyIncome',cm
        plt.colorbar(label = 'StatedMonthlyIncome');
        plt.xlabel('Loan Original Amount')
        plt.ylabel('Term')
        plt.title('Term vs LoanOriginalAmount  vs StatedMonthlyIncome');
```

1.7.2 observation

we can see that there are negative correlation between Loan Original Amount , Term and Stated Monthly Income and
most of Loan Original Amount have StatedMonthlyIncome below of 100000$

1.7.3 question 13 relation between Employment Status, Loan Status, EmploymentStatus

```
In [ ]: plt.figure(figsize=[12,10])
        sb.boxplot(data=loan_clean, x="LoanStatus", y="BorrowerAPR", hue="EmploymentStatus");
        plt.xticks(rotation = 90);
        plt.xlabel('Loan Status');
        plt.ylabel('BorrowerAPR');
        plt.title('Loan Status Vs BorrowerAPR VS Employment Status');
```

1.7.4 observation

For each category of loan status, the lowest APR is generaly for Employed and Full-time. the highest APR is generaly Not employed and self employed

question 14 relation between DebtToIncomeRatio, LoanOriginalAmount and LoanStatus

```
In [ ]: plt.figure(figsize = [10, 5])
        plt.scatter(data=loan_clean,x='DebtToIncomeRatio',y = 'LoanStatus',c='LoanOriginalAmount
        plt.colorbar(label = 'LoanOriginalAmount');
        plt.xlabel('DebtToIncomeRatio')
        plt.ylabel('LoanStatus')
        plt.title('LoanOriginalAmount vs LoanOriginalAmount  vs StatedMonthlyIncome');
```

1.7.5 observation

there are negative correlation between Debt To Income Ratio and Loan Status

1.7.6 question 15

```
In [ ]: fig = plt.figure(figsize = [20,8])
        ax = sb.pointplot(data = loan_clean, x = 'LoanStatus', y = 'BorrowerAPR', hue = 'Employm
                          dodge = 0.5)
        plt.ylabel('BorrowerAPR')
        plt.xticks(rotation=15)
        plt.title('EmploymentStatus vs BorrowerAPR in each LoanStatus')
        plt.legend(ncol=2)
        plt.show();
```

1.7.7 observation

the past due loan have the most higth BorrowerAPR

1.7.8 Talk about some of the relationships you observed in this part of the investigation. Were there features that strengthened each other in terms of looking at your feature(s) of interest?

- the past due loan have the most higth BorrowerAPR
- there are negative correlation between Debt To Income Ratio and Loan Status
- For each category of loan status, the lowest APR is generaly for Employed and Full-time. the highest APR is generaly Not employed and self employed
- we can see that there are negative correlation between Loan Original Amount , Term and Stated Monthly Income and most of Loan Original Amount have StatedMonthlyIncome below of 100000($)
-

1.7.9 Were there any interesting or surprising interactions between features?- we firstly begin with our features of interest: what is the distribution of LoanStatus?

none

1.8 Conclusions

after ours exploration we find this features in ours dataset: - the Stated Monthly Income of borrowers is most between 1000 and 30000 - we can see that the higths monthly payment are between 100 and 1000 but the most higth is between 100 and 200 - we fine that the rating of most common borrowers is between C to D - we look that most of the borrowers have 0 recommandation at the time the listing was created. - because Other and professional which are the two first occupation not give us usefull information we ignore it and we observe that: executive, computer programmer, teacher ,analyst, administrative assistant are the five first occupation - the most length of the loan are 36 month - the most borrowers are Employed - from the distribution of LoanStatus feature we can observe that the Loan Status must hight is Current - the past due loan have the most higth BorrowerAPR - there are negative correlation between Debt To Income Ratio and Loan Status - For each category of loan status, the lowest APR is generaly for Employed and Full-time. the highest APR is generaly Not employed and self employed - we can see that there are negative correlation between Loan Original Amount , Term and Stated Monthly Income and most of Loan Original Amount have StatedMonthlyIncome below of 100000($) - we see that all Loan Statut have almost the same Listing Category - we note that defaulted and chargedoff have the lower IQR - we see strong positive correlation between creditScoreRangeLower and CreditScoreRangeUper, between LoanOrginalAmount and MonthtlyPayement, between BorrowerAPR and BorrowerRate

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

In [ ]: