Prosper_Loan_Data_Analyses

January 18, 2021

1 Loan Data from Prosper

by Yara Aldajjani

1.1 Contents:

Introduction This dataset is financial dataset and this is related to the loan, borrowers, lenders, interest rates and stuffs like that. Prosper or Prosper Marketplace Inc. is a San Francisco, California based company specializing in loans at low interest rates to the borrowers. In this dataset, we are using the data from the Posper to analyse it and trying to find the pattern in the Prosper data. This may be tedious because of the sheer size of the dataset and the complicated nature of all the financial datasets. We are using R, an advanced high level programming language of the analysis with some of its most popular graphic package ggplot. The codes are not visible in the HTML/PDF export for the simlicity but the codes can be reviewed from the RMD file.

The dataset is comprised of 81 variables and contains 113937 entries. The variable that are explored in the dataset are the following Term : Amount of month customers opted for loan

[2]:
```
# Download the needed libraries
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline
```

Data Wrangling

Data Gathering The dataset contains 113,937 loans with 81 variables on each loan, such as borrower rate (or interest rate), loan amount and so on. The data can be found here:https://s3.amazonaws.com/udacity-hosted-downloads/ud651/prosperLoanData.csv

[3]:
```
# Loading the loans dataset
Loan_Data = pd.read_csv('prosperLoanData.csv')
```

[4]:
```
# Load the loans dataset's dictionary
Loan_Data_Dict= pd.read_csv('ProsperLoanDataDictionary.csv')
```

Data Assessment **Visual Assessment**: **Loan_Data**:

[5]:
```
# To print all the columns in order to have a better look for the visual␣
↪assessment
pd.set_option('display.max_columns', None)
```

[6]:
```
# To prevent the characters truncation
pd.set_option('max_colwidth', None)
```

[7]:
```
Loan_Data
```

[7]:

	ListingKey	ListingNumber	ListingCreationDate	CreditGrade	Term	LoanStatus	ClosedDate
0	1021339766868145413AB3B	193129	2007-08-26 19:09:29.263000000	C	36	Completed	2009-08-14 00:00:00
1	1027360249950330 8B223C1	1209647	2014-02-27 08:28:07.900000000	NaN	36	Current	NaN
2	0EE9337825851032864889A	81716	2007-01-05 15:00:47.090000000	HR	36	Completed	2009-12-17 00:00:00
3	0EF5356002482715299901A	658116	2012-10-22 11:02:35.010000000	NaN	36	Current	NaN
4	0F023589499656230C5E3E2	909464	2013-09-14 18:38:39.097000000	NaN	36	Current	NaN
…	…	…	…	…	…	…	…
113932	E6D93576552482716 9606C	753087	2013-04-14 05:55:02.663000000	NaN	36	Current	NaN
113933	E6DB353036033497292EE43	537216	2011-11-03 20:42:55.333000000	NaN	36	FinalPaymentInProgress	NaN
113934	E6E135961700520296 92BB1	1069178	2013-12-13 05:49:12.703000000	NaN	60	Current	NaN
113935	E6EB3531504622671970D9E	539056	2011-11-14 13:18:26.597000000	NaN	60	Completed	2013-08-13 00:00:00
113936	E6ED3600409833199F711B7	1140093	2014-01-15 09:27:37.657000000	NaN	36	Current	NaN

```
        BorrowerAPR  BorrowerRate  LenderYield  EstimatedEffectiveYield  \
0           0.16516        0.1580       0.1380                      NaN
1           0.12016        0.0920       0.0820                  0.07960
2           0.28269        0.2750       0.2400                      NaN
3           0.12528        0.0974       0.0874                  0.08490
4           0.24614        0.2085       0.1985                  0.18316
…                 …             …            …                        …
113932      0.22354        0.1864       0.1764                  0.16490
113933      0.13220        0.1110       0.1010                  0.10070
113934      0.23984        0.2150       0.2050                  0.18828
113935      0.28408        0.2605       0.2505                  0.24450
113936      0.13189        0.1039       0.0939                  0.09071

        EstimatedLoss  EstimatedReturn  ProsperRating (numeric)  \
0                 NaN              NaN                      NaN
1              0.0249          0.05470                      6.0
2                 NaN              NaN                      NaN
3              0.0249          0.06000                      6.0
4              0.0925          0.09066                      3.0
…                 …                …                        …
113932         0.0699          0.09500                      4.0
113933         0.0200          0.08070                      6.0
113934         0.1025          0.08578                      3.0
113935         0.0850          0.15950                      4.0
113936         0.0299          0.06081                      6.0

       ProsperRating (Alpha)  ProsperScore  ListingCategory (numeric)  \
0                        NaN           NaN                          0
1                          A           7.0                          2
2                        NaN           NaN                          0
3                          A           9.0                         16
4                          D           4.0                          2
…                        …             …                          …
113932                     C           5.0                          1
113933                     A           8.0                          7
113934                     D           3.0                          1
113935                     C           5.0                          2
113936                     A           7.0                          1

       BorrowerState              Occupation EmploymentStatus  \
0                 CO                   Other    Self-employed
1                 CO            Professional         Employed
2                 GA                   Other    Not available
3                 GA           Skilled Labor         Employed
4                 MN               Executive         Employed
…                 …                       …                …
```

```
113932           IL  Food Service Management         Employed
113933           PA              Professional         Employed
113934           TX                     Other         Employed
113935           GA              Food Service        Full-time
113936           NY                 Professor         Employed

        EmploymentStatusDuration  IsBorrowerHomeowner  CurrentlyInGroup  \
0                            2.0                 True              True
1                           44.0                False             False
2                            NaN                False              True
3                          113.0                 True             False
4                           44.0                 True             False
…                            …                    …                 …
113932                     246.0                 True             False
113933                      21.0                 True             False
113934                      84.0                 True             False
113935                      94.0                 True             False
113936                     244.0                False             False

                     GroupKey               DateCreditPulled  \
0                         NaN  2007-08-26 18:41:46.780000000
1                         NaN            2014-02-27 08:28:14
2       783C3371218786870A73D20  2007-01-02 14:09:10.060000000
3                         NaN            2012-10-22 11:02:32
4                         NaN            2013-09-14 18:38:44
…                         …                              …
113932                    NaN            2013-04-14 05:54:58
113933                    NaN            2011-11-03 20:42:53
113934                    NaN            2013-12-13 05:49:15
113935                    NaN            2011-11-14 13:18:24
113936                    NaN            2014-01-15 09:27:40

        CreditScoreRangeLower  CreditScoreRangeUpper FirstRecordedCreditLine  \
0                       640.0                  659.0     2001-10-11 00:00:00
1                       680.0                  699.0     1996-03-18 00:00:00
2                       480.0                  499.0     2002-07-27 00:00:00
3                       800.0                  819.0     1983-02-28 00:00:00
4                       680.0                  699.0     2004-02-20 00:00:00
…                       …                      …                         …
113932                  700.0                  719.0     1997-09-01 00:00:00
113933                  700.0                  719.0     1992-01-17 00:00:00
113934                  700.0                  719.0     2002-02-25 00:00:00
113935                  680.0                  699.0     1993-12-01 00:00:00
113936                  680.0                  699.0     1995-01-01 00:00:00

        CurrentCreditLines  OpenCreditLines  TotalCreditLinespast7years  \
0                      5.0              4.0                        12.0
```

```
1                    14.0            14.0                        29.0
2                     NaN             NaN                         3.0
3                     5.0             5.0                        29.0
4                    19.0            19.0                        49.0
…                     …               …                           …
113932                9.0             9.0                        41.0
113933               14.0            13.0                        39.0
113934               12.0            12.0                        25.0
113935               11.0            11.0                        22.0
113936               10.0             9.0                        44.0

        OpenRevolvingAccounts  OpenRevolvingMonthlyPayment  \
0                           1                         24.0
1                          13                        389.0
2                           0                          0.0
3                           7                        115.0
4                           6                        220.0
…                           …                          …
113932                      9                        209.0
113933                      9                        495.0
113934                      9                        521.0
113935                      7                        488.0
113936                      8                        289.0

        InquiriesLast6Months  TotalInquiries  CurrentDelinquencies  \
0                        3.0             3.0                   2.0
1                        3.0             5.0                   0.0
2                        0.0             1.0                   1.0
3                        0.0             1.0                   4.0
4                        1.0             9.0                   0.0
…                         …               …                     …
113932                   0.0             0.0                   0.0
113933                   1.0             4.0                   1.0
113934                   1.0             2.0                   0.0
113935                   1.0             4.0                   0.0
113936                   0.0             1.0                   1.0

        AmountDelinquent  DelinquenciesLast7Years  PublicRecordsLast10Years  \
0                  472.0                      4.0                       0.0
1                    0.0                      0.0                       1.0
2                    NaN                      0.0                       0.0
3                10056.0                     14.0                       0.0
4                    0.0                      0.0                       0.0
…                     …                        …                         …
113932               0.0                      7.0                       1.0
113933            5062.0                      4.0                       0.0
113934               0.0                      0.0                       0.0
```

```
113935              0.0                     0.0                     1.0
113936            257.0                     3.0                     1.0

        PublicRecordsLast12Months  RevolvingCreditBalance  \
0                             0.0                     0.0
1                             0.0                  3989.0
2                             NaN                     NaN
3                             0.0                  1444.0
4                             0.0                  6193.0
…                             …                       …
113932                        0.0                  7714.0
113933                        0.0                 15743.0
113934                        0.0                 22147.0
113935                        0.0                 11956.0
113936                        0.0                  6166.0

        BankcardUtilization  AvailableBankcardCredit  TotalTrades  \
0                      0.00                   1500.0         11.0
1                      0.21                  10266.0         29.0
2                       NaN                     NaN          NaN
3                      0.04                  30754.0         26.0
4                      0.81                    695.0         39.0
…                         …                       …            …
113932                 0.80                   1886.0         37.0
113933                 0.69                   6658.0         39.0
113934                 0.73                   7853.0         25.0
113935                 0.69                   4137.0         19.0
113936                 0.80                    675.0         36.0

        TradesNeverDelinquent (percentage)  TradesOpenedLast6Months  \
0                                     0.81                      0.0
1                                     1.00                      2.0
2                                      NaN                      NaN
3                                     0.76                      0.0
4                                     0.95                      2.0
…                                        …                        …
113932                                0.83                      3.0
113933                                0.92                      0.0
113934                                1.00                      0.0
113935                                0.80                      1.0
113936                                0.75                      0.0

        DebtToIncomeRatio     IncomeRange  IncomeVerifiable  \
0                    0.17  $25,000-49,999              True
1                    0.18  $50,000-74,999              True
2                    0.06   Not displayed              True
3                    0.15  $25,000-49,999              True
```

```
4                    0.26       $100,000+              True
…                     …            …                  …
113932               0.13  $50,000-74,999              True
113933               0.11  $75,000-99,999              True
113934               0.51  $25,000-49,999              True
113935               0.48  $25,000-49,999              True
113936               0.23  $50,000-74,999              True

        StatedMonthlyIncome                  LoanKey  TotalProsperLoans  \
0              3083.333333  E33A3400205839220442E84                NaN
1              6125.000000  9E3B37071505919926B1D82                NaN
2              2083.333333  695433796004681785 1BCB2              NaN
3              2875.000000  A0393664465886295619C51               NaN
4              9583.333333  A1803693021888892006 89E              1.0
…                    …                      …                  …
113932         4333.333333  9BD7367919051593140DB62               NaN
113933         8041.666667  62D93634569816897D5A276               3.0
113934         2875.000000  DD1A370200396006300ACA0               NaN
113935         3875.000000  589536350469116027ED11B               1.0
113936         4583.333333  00AF3704550953269A64E40                NaN

        TotalProsperPaymentsBilled  OnTimeProsperPayments  \
0                              NaN                    NaN
1                              NaN                    NaN
2                              NaN                    NaN
3                              NaN                    NaN
4                             11.0                   11.0
…                              …                      …
113932                         NaN                    NaN
113933                        60.0                   60.0
113934                         NaN                    NaN
113935                        16.0                   16.0
113936                         NaN                    NaN

        ProsperPaymentsLessThanOneMonthLate  ProsperPaymentsOneMonthPlusLate  \
0                                       NaN                              NaN
1                                       NaN                              NaN
2                                       NaN                              NaN
3                                       NaN                              NaN
4                                       0.0                              0.0
…                                        …                                …
113932                                  NaN                              NaN
113933                                  0.0                              0.0
113934                                  NaN                              NaN
113935                                  0.0                              0.0
113936                                  NaN                              NaN
```

```
        ProsperPrincipalBorrowed  ProsperPrincipalOutstanding  \
0                            NaN                          NaN
1                            NaN                          NaN
2                            NaN                          NaN
3                            NaN                          NaN
4                        11000.0                      9947.90
…                            …                            …
113932                       NaN                          NaN
113933                   33501.0                      4815.42
113934                       NaN                          NaN
113935                    5000.0                      3264.37
113936                       NaN                          NaN

        ScorexChangeAtTimeOfListing  LoanCurrentDaysDelinquent  \
0                               NaN                          0
1                               NaN                          0
2                               NaN                          0
3                               NaN                          0
4                               NaN                          0
…                               …                            …
113932                          NaN                          0
113933                        -26.0                          0
113934                          NaN                          0
113935                         -4.0                          0
113936                          NaN                          0

        LoanFirstDefaultedCycleNumber  LoanMonthsSinceOrigination  LoanNumber  \
0                                 NaN                          78       19141
1                                 NaN                           0      134815
2                                 NaN                          86        6466
3                                 NaN                          16       77296
4                                 NaN                           6      102670
…                                 …                            …           …
113932                            NaN                          11       88485
113933                            NaN                          28       55801
113934                            NaN                           3      123122
113935                            NaN                          28       56401
113936                            NaN                           2      127508

        LoanOriginalAmount  LoanOriginationDate LoanOriginationQuarter  \
0                     9425  2007-09-12 00:00:00                Q3 2007
1                    10000  2014-03-03 00:00:00                Q1 2014
2                     3001  2007-01-17 00:00:00                Q1 2007
3                    10000  2012-11-01 00:00:00                Q4 2012
4                    15000  2013-09-20 00:00:00                Q3 2013
…                        …                    …                    …
113932               10000  2013-04-22 00:00:00                Q2 2013
```

```
113933               2000  2011-11-07 00:00:00               Q4 2011
113934              10000  2013-12-23 00:00:00               Q4 2013
113935              15000  2011-11-21 00:00:00               Q4 2011
113936               2000  2014-01-21 00:00:00               Q1 2014

                    MemberKey  MonthlyLoanPayment  LP_CustomerPayments  \
0       1F3E3376408759268057EDA              330.43          11396.1400
1       1D13370546739025387B2F4              318.93              0.0000
2       5F7033715035555618FA612              123.32           4186.6300
3       9ADE356069835475068C6D2              321.45           5143.2000
4       36CE356043264555721F06C              563.97           2819.8500
…                           …                   …                   …
113932  2EC435768441332602FDC15              364.74           3647.4000
113933  55C4336679182766893E4FC               65.57           2330.5500
113934  0FE0370029359765342FDB5              273.35            546.7000
113935  A3383486182227278262 1C8             449.55          21122.5600
113936  CE1E3704648000761C9F724               64.90             64.3307

        LP_CustomerPrincipalPayments  LP_InterestandFees  LP_ServiceFees  \
0                            9425.00           1971.1400         -133.18
1                               0.00              0.0000            0.00
2                            3001.00           1185.6300          -24.20
3                            4091.09           1052.1100         -108.01
4                            1563.22           1256.6300          -60.27
…                              …                 …                …
113932                       2238.38           1409.0200          -75.58
113933                       1997.16            333.3900          -30.05
113934                        183.15            363.5500          -16.91
113935                      15000.00           6122.5600         -235.05
113936                         47.25             17.0807           -1.70

        LP_CollectionFees  LP_GrossPrincipalLoss  LP_NetPrincipalLoss  \
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
…                       …                      …                    …
113932                0.0                    0.0                  0.0
113933                0.0                    0.0                  0.0
113934                0.0                    0.0                  0.0
113935                0.0                    0.0                  0.0
113936                0.0                    0.0                  0.0

        LP_NonPrincipalRecoverypayments  PercentFunded  Recommendations  \
0                                   0.0            1.0                0
1                                   0.0            1.0                0
```

```
2                                    0.0            1.0                 0
3                                    0.0            1.0                 0
4                                    0.0            1.0                 0
…                                    …              …                   …
113932                               0.0            1.0                 0
113933                               0.0            1.0                 0
113934                               0.0            1.0                 0
113935                               0.0            1.0                 0
113936                               0.0            1.0                 0

        InvestmentFromFriendsCount  InvestmentFromFriendsAmount  Investors
0                                0                          0.0        258
1                                0                          0.0          1
2                                0                          0.0         41
3                                0                          0.0        158
4                                0                          0.0         20
…                                …                          …          …
113932                           0                          0.0          1
113933                           0                          0.0         22
113934                           0                          0.0        119
113935                           0                          0.0        274
113936                           0                          0.0          1

[113937 rows x 81 columns]
```

Loan_Data_Dict:

[8]:
```
# To print all the rows in order to have a better look for the visual assessment
pd.set_option("max_rows", None)
```

[9]:
```
Loan_Data_Dict
```

[9]:
```
                            Variable  \
0                          ListingKey
1                       ListingNumber
2                  ListingCreationDate
3                          CreditGrade
4                                 Term
5                           LoanStatus
6                           ClosedDate
7                          BorrowerAPR
8                         BorrowerRate
9                           LenderYield
10              EstimatedEffectiveYield
11                        EstimatedLoss
12                      EstimatedReturn
13              ProsperRating (numeric)
```

```
14                ProsperRating (Alpha)
15                         ProsperScore
16                       ListingCategory
17                         BorrowerState
18                            Occupation
19                      EmploymentStatus
20              EmploymentStatusDuration
21                   IsBorrowerHomeowner
22                      CurrentlyInGroup
23                               GroupKey
24                      DateCreditPulled
25                CreditScoreRangeLower
26                CreditScoreRangeUpper
27              FirstRecordedCreditLine
28                   CurrentCreditLines
29                      OpenCreditLines
30            TotalCreditLinespast7years
31                 OpenRevolvingAccounts
32           OpenRevolvingMonthlyPayment
33                  InquiriesLast6Months
34                        TotalInquiries
35                  CurrentDelinquencies
36                      AmountDelinquent
37              DelinquenciesLast7Years
38              PublicRecordsLast10Years
39             PublicRecordsLast12Months
40                RevolvingCreditBalance
41                   BankcardUtilization
42              AvailableBankcardCredit
43                           TotalTrades
44              TradesNeverDelinquent
45              TradesOpenedLast6Months
46                    DebtToIncomeRatio
47                           IncomeRange
48                     IncomeVerifiable
49                  StatedMonthlyIncome
50                              LoanKey
51                    TotalProsperLoans
52           TotalProsperPaymentsBilled
53                 OnTimeProsperPayments
54  ProsperPaymentsLessThanOneMonthLate
55      ProsperPaymentsOneMonthPlusLate
56              ProsperPrincipalBorrowed
57           ProsperPrincipalOutstanding
58           ScorexChangeAtTimeOfListing
59             LoanCurrentDaysDelinquent
60         LoanFirstDefaultedCycleNumber
```

61	LoanMonthsSinceOrigination	
62	LoanNumber	
63	LoanOriginalAmount	
64	LoanOriginationDate	
65	LoanOriginationQuarter	
66	MemberKey	
67	MonthlyLoanPayment	
68	LP_CustomerPayments	
69	LP_CustomerPrincipalPayments	
70	LP_InterestandFees	
71	LP_ServiceFees	
72	LP_CollectionFees	
73	LP_GrossPrincipalLoss	
74	LP_NetPrincipalLoss	
75	LP_NonPrincipalRecoverypayments	
76	PercentFunded	
77	Recommendations	
78	InvestmentFromFriendsCount	
79	InvestmentFromFriendsAmount	
80	Investors	

	Description
0	Unique key for each listing, same value as the 'key' used in the listing object in the API.
1	The number that uniquely identifies the listing to the public as displayed on the website.
2	The date the listing was created.
3	The Credit rating that was assigned at the time the listing went live. Applicable for listings pre-2009 period and will only be populated for those listings.
4	The length of the loan expressed in months.
5	The current status of the loan: Cancelled, Chargedoff, Completed, Current, Defaulted, FinalPaymentInProgress, PastDue. The PastDue status will be accompanied by a delinquency bucket.
6	Closed date is applicable for Cancelled, Completed, Chargedoff and Defaulted loan statuses.
7	The Borrower's Annual Percentage Rate (APR) for the loan.
8	The Borrower's interest rate for this loan.

9
The Lender yield on the loan. Lender yield is equal to the interest rate on the loan less the servicing fee.
10
Effective yield is equal to the borrower interest rate (i) minus the servicing fee rate, (ii) minus estimated uncollected interest on charge-offs, (iii) plus estimated collected late fees. Applicable for loans originated after July 2009.
11
Estimated loss is the estimated principal loss on charge-offs. Applicable for loans originated after July 2009.
12
The estimated return assigned to the listing at the time it was created. Estimated return is the difference between the Estimated Effective Yield and the Estimated Loss Rate. Applicable for loans originated after July 2009.
13
The Prosper Rating assigned at the time the listing was created: 0 - N/A, 1 - HR, 2 - E, 3 - D, 4 - C, 5 - B, 6 - A, 7 - AA. Applicable for loans originated after July 2009.
14
The Prosper Rating assigned at the time the listing was created between AA - HR. Applicable for loans originated after July 2009.
15
A custom risk score built using historical Prosper data. The score ranges from 1-10, with 10 being the best, or lowest risk score. Applicable for loans originated after July 2009.
16 The category of the listing that the borrower selected when posting their listing: 0 - Not Available, 1 - Debt Consolidation, 2 - Home Improvement, 3 - Business, 4 - Personal Loan, 5 - Student Use, 6 - Auto, 7- Other, 8 - Baby&Adoption, 9 - Boat, 10 - Cosmetic Procedure, 11 - Engagement Ring, 12 - Green Loans, 13 - Household Expenses, 14 - Large Purchases, 15 - Medical/Dental, 16 - Motorcycle, 17 - RV, 18 - Taxes, 19 - Vacation, 20 - Wedding Loans
17
The two letter abbreviation of the state of the address of the borrower at the time the Listing was created.
18
The Occupation selected by the Borrower at the time they created the listing.
19
The employment status of the borrower at the time they posted the listing.
20
The length in months of the employment status at the time the listing was created.
21
A Borrower will be classified as a homowner if they have a mortgage on their credit profile or provide documentation confirming they are a homeowner.
22
Specifies whether or not the Borrower was in a group at the time the listing was created.

23
The Key of the group in which the Borrower is a member of. Value will be null if the borrower does not have a group affiliation.
24
The date the credit profile was pulled.
25
The lower value representing the range of the borrower's credit score as provided by a consumer credit rating agency.
26
The upper value representing the range of the borrower's credit score as provided by a consumer credit rating agency.
27
The date the first credit line was opened.
28
Number of current credit lines at the time the credit profile was pulled.
29
Number of open credit lines at the time the credit profile was pulled.
30
Number of credit lines in the past seven years at the time the credit profile was pulled.
31
Number of open revolving accounts at the time the credit profile was pulled.
32
Monthly payment on revolving accounts at the time the credit profile was pulled.
33
Number of inquiries in the past six months at the time the credit profile was pulled.
34
Total number of inquiries at the time the credit profile was pulled.
35
Number of accounts delinquent at the time the credit profile was pulled.
36
Dollars delinquent at the time the credit profile was pulled.
37
Number of delinquencies in the past 7 years at the time the credit profile was pulled.
38
Number of public records in the past 10 years at the time the credit profile was pulled.
39
Number of public records in the past 12 months at the time the credit profile was pulled.
40
Dollars of revolving credit at the time the credit profile was pulled.
41
The percentage of available revolving credit that is utilized at the time the credit profile was pulled.

42
The total available credit via bank card at the time the credit profile was pulled.
43
Number of trade lines ever opened at the time the credit profile was pulled.
44
Number of trades that have never been delinquent at the time the credit profile was pulled.
45
Number of trades opened in the last 6 months at the time the credit profile was pulled.
46
The debt to income ratio of the borrower at the time the credit profile was pulled. This value is Null if the debt to income ratio is not available. This value is capped at 10.01 (any debt to income ratio larger than 1000% will be returned as 1001%).
47
The income range of the borrower at the time the listing was created.
48
The borrower indicated they have the required documentation to support their income.
49
The monthly income the borrower stated at the time the listing was created.
50
Unique key for each loan. This is the same key that is used in the API.
51
Number of Prosper loans the borrower at the time they created this listing. This value will be null if the borrower had no prior loans.
52
Number of on time payments the borrower made on Prosper loans at the time they created this listing. This value will be null if the borrower had no prior loans.
53
Number of on time payments the borrower had made on Prosper loans at the time they created this listing. This value will be null if the borrower has no prior loans.
54
Number of payments the borrower made on Prosper loans that were less than one month late at the time they created this listing. This value will be null if the borrower had no prior loans.
55
Number of payments the borrower made on Prosper loans that were greater than one month late at the time they created this listing. This value will be null if the borrower had no prior loans.
56
Total principal borrowed on Prosper loans at the time the listing was created. This value will be null if the borrower had no prior loans.

57
Principal outstanding on Prosper loans at the time the listing was created. This value will be null if the borrower had no prior loans.
58
Borrower's credit score change at the time the credit profile was pulled. This will be the change relative to the borrower's last Prosper loan. This value will be null if the borrower had no prior loans.
59
The number of days delinquent.
60
The cycle the loan was charged off. If the loan has not charged off the value will be null.
61
Number of months since the loan originated.
62
Unique numeric value associated with the loan.
63
The origination amount of the loan.
64
The date the loan was originated.
65
The quarter in which the loan was originated.
66
The unique key that is associated with the borrower. This is the same identifier that is used in the API member object.
67
The scheduled monthly loan payment.
68
Pre charge-off cumulative gross payments made by the borrower on the loan. If the loan has charged off, this value will exclude any recoveries.
69
Pre charge-off cumulative principal payments made by the borrower on the loan. If the loan has charged off, this value will exclude any recoveries.
70
Pre charge-off cumulative interest and fees paid by the borrower. If the loan has charged off, this value will exclude any recoveries.
71
Cumulative service fees paid by the investors who have invested in the loan.
72
Cumulative collection fees paid by the investors who have invested in the loan.
73
The gross charged off amount of the loan.
74
The principal that remains uncollected after any recoveries.
75
The interest and fee component of any recovery payments. The current payment policy applies payments in the following order: Fees, interest, principal.

```
76
Percent the listing was funded.
77
Number of recommendations the borrower had at the time the listing was created.
78
Number of friends that made an investment in the loan.
79
Dollar amount of investments that were made by friends.
80
The number of investors that funded the loan.
```

Programmatic Assessment: Loan_Data:

[10]:
```
# getting a general information about the dataset
Loan_Data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 113937 entries, 0 to 113936
Data columns (total 81 columns):
 #   Column                               Non-Null Count   Dtype
---  ------                               --------------   -----
 0   ListingKey                           113937 non-null  object
 1   ListingNumber                        113937 non-null  int64
 2   ListingCreationDate                  113937 non-null  object
 3   CreditGrade                          28953 non-null   object
 4   Term                                 113937 non-null  int64
 5   LoanStatus                           113937 non-null  object
 6   ClosedDate                           55089 non-null   object
 7   BorrowerAPR                          113912 non-null  float64
 8   BorrowerRate                         113937 non-null  float64
 9   LenderYield                          113937 non-null  float64
 10  EstimatedEffectiveYield              84853 non-null   float64
 11  EstimatedLoss                        84853 non-null   float64
 12  EstimatedReturn                      84853 non-null   float64
 13  ProsperRating (numeric)              84853 non-null   float64
 14  ProsperRating (Alpha)                84853 non-null   object
 15  ProsperScore                         84853 non-null   float64
 16  ListingCategory (numeric)            113937 non-null  int64
 17  BorrowerState                        108422 non-null  object
 18  Occupation                           110349 non-null  object
 19  EmploymentStatus                     111682 non-null  object
 20  EmploymentStatusDuration             106312 non-null  float64
 21  IsBorrowerHomeowner                  113937 non-null  bool
 22  CurrentlyInGroup                     113937 non-null  bool
 23  GroupKey                             13341 non-null   object
 24  DateCreditPulled                     113937 non-null  object
 25  CreditScoreRangeLower                113346 non-null  float64
 26  CreditScoreRangeUpper                113346 non-null  float64
```

```
27  FirstRecordedCreditLine              113240 non-null  object
28  CurrentCreditLines                   106333 non-null  float64
29  OpenCreditLines                      106333 non-null  float64
30  TotalCreditLinespast7years           113240 non-null  float64
31  OpenRevolvingAccounts                113937 non-null  int64
32  OpenRevolvingMonthlyPayment          113937 non-null  float64
33  InquiriesLast6Months                 113240 non-null  float64
34  TotalInquiries                       112778 non-null  float64
35  CurrentDelinquencies                 113240 non-null  float64
36  AmountDelinquent                     106315 non-null  float64
37  DelinquenciesLast7Years              112947 non-null  float64
38  PublicRecordsLast10Years             113240 non-null  float64
39  PublicRecordsLast12Months            106333 non-null  float64
40  RevolvingCreditBalance               106333 non-null  float64
41  BankcardUtilization                  106333 non-null  float64
42  AvailableBankcardCredit              106393 non-null  float64
43  TotalTrades                          106393 non-null  float64
44  TradesNeverDelinquent (percentage)   106393 non-null  float64
45  TradesOpenedLast6Months              106393 non-null  float64
46  DebtToIncomeRatio                    105383 non-null  float64
47  IncomeRange                          113937 non-null  object
48  IncomeVerifiable                     113937 non-null  bool
49  StatedMonthlyIncome                  113937 non-null  float64
50  LoanKey                              113937 non-null  object
51  TotalProsperLoans                    22085 non-null   float64
52  TotalProsperPaymentsBilled           22085 non-null   float64
53  OnTimeProsperPayments                22085 non-null   float64
54  ProsperPaymentsLessThanOneMonthLate  22085 non-null   float64
55  ProsperPaymentsOneMonthPlusLate      22085 non-null   float64
56  ProsperPrincipalBorrowed             22085 non-null   float64
57  ProsperPrincipalOutstanding          22085 non-null   float64
58  ScorexChangeAtTimeOfListing          18928 non-null   float64
59  LoanCurrentDaysDelinquent            113937 non-null  int64
60  LoanFirstDefaultedCycleNumber        16952 non-null   float64
61  LoanMonthsSinceOrigination           113937 non-null  int64
62  LoanNumber                           113937 non-null  int64
63  LoanOriginalAmount                   113937 non-null  int64
64  LoanOriginationDate                  113937 non-null  object
65  LoanOriginationQuarter               113937 non-null  object
66  MemberKey                            113937 non-null  object
67  MonthlyLoanPayment                   113937 non-null  float64
68  LP_CustomerPayments                  113937 non-null  float64
69  LP_CustomerPrincipalPayments         113937 non-null  float64
70  LP_InterestandFees                   113937 non-null  float64
71  LP_ServiceFees                       113937 non-null  float64
72  LP_CollectionFees                    113937 non-null  float64
73  LP_GrossPrincipalLoss                113937 non-null  float64
74  LP_NetPrincipalLoss                  113937 non-null  float64
```

```
 75  LP_NonPrincipalRecoverypayments      113937 non-null  float64
 76  PercentFunded                        113937 non-null  float64
 77  Recommendations                      113937 non-null  int64
 78  InvestmentFromFriendsCount           113937 non-null  int64
 79  InvestmentFromFriendsAmount          113937 non-null  float64
 80  Investors                            113937 non-null  int64
dtypes: bool(3), float64(50), int64(11), object(17)
memory usage: 68.1+ MB
```

[11]:
```
# getting a sample of the dataset
Loan_Data.sample(15)
```

[11]:

	ListingKey	ListingNumber	ListingCreationDate	\
33895	312D3583465619614BA7F17	844102	2013-07-18 08:40:10.107000000	
9604	0B573596637381710A9B089	1076913	2013-12-10 08:19:13.563000000	
110363	285C3488392003833C948D6	466789	2010-07-15 12:26:58.200000000	
109947	F8A836038880454107EBD91	1231593	2014-02-25 06:59:25.947000000	
52000	7968359013552470088D3D6	924235	2013-09-27 05:45:15.240000000	
109394	F7DA35907885096164F7FC1	945806	2013-10-09 11:54:09.730000000	
82174	A4B63521801690376EA8576	518246	2011-07-25 10:58:03.740000000	
89123	BB0C35793116643718CE118	783948	2013-05-20 07:46:28.757000000	
60915	C4E635499211965543BB7AA	599489	2012-06-13 13:11:52.183000000	
14063	68A7343183891465988076F	401164	2008-09-19 16:58:03.377000000	
75091	D5953379591117372E6DA24	88242	2007-01-24 03:44:21.333000000	
56298	BF193595936194077991ACC	1068585	2013-12-04 11:13:11.753000000	
71693	3CCA35994398036693F4ED0	1143069	2014-01-18 05:37:44.147000000	
109411	808E339675450706933FDC3	177841	2007-07-28 16:51:43.313000000	
47532	DEA43492968069728B1920F	473182	2010-08-25 17:12:03.237000000	

	CreditGrade	Term	LoanStatus	ClosedDate	BorrowerAPR	\
33895	NaN	60	Current	NaN	0.27637	
9604	NaN	60	Current	NaN	0.20347	
110363	NaN	36	Completed	2011-08-25 00:00:00	0.24822	
109947	NaN	60	Current	NaN	0.21345	
52000	NaN	60	Current	NaN	0.16294	
109394	NaN	36	Current	NaN	0.22875	
82174	NaN	36	Completed	2013-05-21 00:00:00	0.34887	
89123	NaN	36	Current	NaN	0.20053	
60915	NaN	36	Completed	2013-11-20 00:00:00	0.18316	
14063	A	36	Defaulted	2011-05-01 00:00:00	0.18958	
75091	C	36	Chargedoff	2008-08-08 00:00:00	0.14106	
56298	NaN	36	Current	NaN	0.25279	
71693	NaN	36	Current	NaN	0.14206	
109411	E	36	Completed	2010-08-07 00:00:00	0.30168	
47532	NaN	36	Completed	2011-12-09 00:00:00	0.39821	

	BorrowerRate	LenderYield	EstimatedEffectiveYield	EstimatedLoss	\

33895	0.2506	0.2406	0.22110	0.1025
9604	0.1795	0.1695	0.15767	0.0774
110363	0.2255	0.2155	0.13040	0.0800
109947	0.1890	0.1790	0.16554	0.0874
52000	0.1399	0.1299	0.12380	0.0449
109394	0.1915	0.1815	0.16862	0.0799
82174	0.3125	0.3025	0.28480	0.1700
89123	0.1639	0.1539	0.14520	0.0574
60915	0.1469	0.1369	0.13460	0.0420
14063	0.1750	0.1650	NaN	NaN
75091	0.1340	0.1240	NaN	NaN
56298	0.2150	0.2050	0.18828	0.1025
71693	0.1139	0.1039	0.09989	0.0349
109411	0.2900	0.2800	NaN	NaN
47532	0.3500	0.3400	0.13690	0.1830

	EstimatedReturn	ProsperRating (numeric)	ProsperRating (Alpha)	\
33895	0.11860	3.0	D	
9604	0.08027	4.0	C	
110363	0.13040	4.0	C	
109947	0.07814	4.0	C	
52000	0.07890	5.0	B	
109394	0.08872	4.0	C	
82174	0.11480	1.0	HR	
89123	0.08780	5.0	B	
60915	0.09260	5.0	B	
14063	NaN	NaN	NaN	
75091	NaN	NaN	NaN	
56298	0.08578	3.0	D	
71693	0.06499	6.0	A	
109411	NaN	NaN	NaN	
47532	0.13690	1.0	HR	

	ProsperScore	ListingCategory (numeric)	BorrowerState	\
33895	3.0	7	PA	
9604	4.0	1	MS	
110363	8.0	1	CO	
109947	8.0	1	GA	
52000	10.0	1	VA	
109394	7.0	1	NJ	
82174	1.0	1	IL	
89123	5.0	2	MI	
60915	6.0	1	CA	
14063	NaN	3	AL	
75091	NaN	0	NaN	
56298	4.0	1	TX	
71693	5.0	1	NC	

```
109411          NaN                         0           CA
47532           6.0                         7           NY

                    Occupation EmploymentStatus  EmploymentStatusDuration  \
33895              Professional         Employed                      39.0
9604                 Nurse (RN)         Employed                     199.0
110363  Administrative Assistant        Full-time                      35.0
109947                    Other         Employed                      27.0
52000        Medical Technician         Employed                      87.0
109394       Sales - Commission         Employed                      29.0
82174                 Executive        Full-time                      64.0
89123                     Other         Employed                     325.0
60915              Professional         Employed                      53.0
14063                     Other    Self-employed                      32.0
75091       Computer Programmer    Not available                       NaN
56298            Accountant/CPA         Employed                      64.0
71693                  Teacher         Employed                      72.0
109411             Professional        Full-time                      24.0
47532   Administrative Assistant         Employed                     128.0

       IsBorrowerHomeowner  CurrentlyInGroup                GroupKey  \
33895                 True             False                     NaN
9604                 False             False                     NaN
110363                True             False                     NaN
109947               False             False                     NaN
52000                False             False                     NaN
109394               False             False                     NaN
82174                False             False                     NaN
89123                False             False                     NaN
60915                 True             False                     NaN
14063                False              True  CCE4336558975150559A215
75091                False              True  C80D337515684920818AB06
56298                 True             False                     NaN
71693                False             False                     NaN
109411               False             False                     NaN
47532                False             False                     NaN

                  DateCreditPulled  CreditScoreRangeLower  \
33895          2013-07-18 08:40:02                  760.0
9604           2013-12-10 08:19:16                  680.0
110363         2010-07-15 12:26:53                  700.0
109947         2014-02-25 06:59:28                  640.0
52000          2013-09-27 05:45:17                  700.0
109394         2013-09-19 09:14:58                  660.0
82174          2011-07-25 10:58:01                  600.0
89123          2013-05-20 07:46:21                  720.0
60915          2012-05-17 19:43:56                  720.0
```

```
14063   2008-09-10 18:29:26.103000000                 720.0
75091   2007-01-24 03:37:06.213000000                 640.0
56298             2013-12-04 11:13:14                 660.0
71693             2014-01-18 05:37:44                 680.0
109411  2007-07-17 20:00:21.360000000                 580.0
47532             2010-08-21 09:50:32                 640.0

        CreditScoreRangeUpper FirstRecordedCreditLine  CurrentCreditLines  \
33895                   779.0     2000-07-26 00:00:00                10.0
9604                    699.0     1990-05-01 00:00:00                12.0
110363                  719.0     2003-06-03 00:00:00                 9.0
109947                  659.0     1996-09-19 00:00:00                 6.0
52000                   719.0     2000-03-17 00:00:00                 8.0
109394                  679.0     1988-09-30 00:00:00                11.0
82174                   619.0     1995-08-02 00:00:00                 7.0
89123                   739.0     1999-08-25 00:00:00                 6.0
60915                   739.0     2001-09-14 00:00:00                18.0
14063                   739.0     2005-07-21 00:00:00                 1.0
75091                   659.0     1998-08-01 00:00:00                 NaN
56298                   679.0     1996-06-24 00:00:00                14.0
71693                   699.0     1997-08-01 00:00:00                13.0
109411                  599.0     2002-12-04 00:00:00                12.0
47532                   659.0     1994-02-24 00:00:00                 4.0

        OpenCreditLines  TotalCreditLinespast7years  OpenRevolvingAccounts  \
33895              10.0                        20.0                      9
9604               11.0                        48.0                     13
110363              9.0                        21.0                      5
109947              6.0                        20.0                      4
52000               8.0                        25.0                      8
109394             10.0                        16.0                      9
82174               9.0                        36.0                      4
89123               6.0                        13.0                      3
60915              17.0                        38.0                     13
14063               1.0                         3.0                      1
75091               NaN                        11.0                      1
56298              11.0                        36.0                      8
71693              13.0                        41.0                     12
109411             11.0                        14.0                     10
47532               2.0                        12.0                      2

        OpenRevolvingMonthlyPayment  InquiriesLast6Months  TotalInquiries  \
33895                         322.0                   3.0             9.0
9604                          906.0                   1.0             3.0
110363                        331.0                   0.0             2.0
109947                         77.0                   0.0             4.0
52000                         372.0                   0.0             1.0
```

```
109394                       478.0                   0.0             4.0
82174                        135.0                   1.0             7.0
89123                        117.0                   1.0             3.0
60915                        500.0                   0.0            13.0
14063                         19.0                   1.0             3.0
75091                         15.0                   1.0             4.0
56298                        599.0                   0.0             2.0
71693                        420.0                   1.0             2.0
109411                       299.0                   2.0             2.0
47532                         79.0                   1.0             1.0

        CurrentDelinquencies  AmountDelinquent  DelinquenciesLast7Years  \
33895                    0.0               0.0                      0.0
9604                     0.0               0.0                      0.0
110363                   0.0               0.0                      0.0
109947                   0.0               0.0                      7.0
52000                    0.0               0.0                      0.0
109394                   0.0               0.0                      0.0
82174                    6.0            8134.0                      9.0
89123                    0.0               0.0                      4.0
60915                    0.0               0.0                      0.0
14063                    0.0               0.0                      0.0
75091                    5.0               NaN                      3.0
56298                    0.0               0.0                      0.0
71693                   12.0               0.0                     14.0
109411                   0.0               0.0                      0.0
47532                    0.0               0.0                      3.0

        PublicRecordsLast10Years  PublicRecordsLast12Months  \
33895                        0.0                        0.0
9604                         0.0                        0.0
110363                       0.0                        0.0
109947                       2.0                        0.0
52000                        1.0                        0.0
109394                       1.0                        0.0
82174                        0.0                        0.0
89123                        0.0                        0.0
60915                        0.0                        0.0
14063                        0.0                        0.0
75091                        2.0                        NaN
56298                        0.0                        0.0
71693                        1.0                        0.0
109411                       0.0                        0.0
47532                        0.0                        0.0

        RevolvingCreditBalance  BankcardUtilization  AvailableBankcardCredit  \
33895                   6413.0                 0.28                   6037.0
```

9604	29286.0	0.87	3533.0
110363	10639.0	0.78	2884.0
109947	1425.0	0.88	138.0
52000	8298.0	0.62	5002.0
109394	11342.0	0.30	22328.0
82174	4019.0	1.00	19.0
89123	1936.0	0.00	0.0
60915	22164.0	0.50	21024.0
14063	732.0	0.13	4768.0
75091	NaN	NaN	NaN
56298	23014.0	0.96	627.0
71693	9039.0	0.15	37495.0
109411	10589.0	0.91	852.0
47532	2545.0	0.98	21.0

	TotalTrades	TradesNeverDelinquent (percentage)	\
33895	18.0	0.88	
9604	36.0	1.00	
110363	18.0	1.00	
109947	16.0	0.81	
52000	24.0	1.00	
109394	16.0	1.00	
82174	30.0	0.69	
89123	13.0	0.76	
60915	36.0	1.00	
14063	3.0	1.00	
75091	NaN	NaN	
56298	30.0	0.90	
71693	27.0	0.81	
109411	14.0	0.78	
47532	8.0	0.62	

	TradesOpenedLast6Months	DebtToIncomeRatio	IncomeRange	\
33895	0.0	0.31	$50,000-74,999	
9604	1.0	0.33	$50,000-74,999	
110363	1.0	0.26	$25,000-49,999	
109947	1.0	0.17	$50,000-74,999	
52000	0.0	0.29	$25,000-49,999	
109394	0.0	0.41	$25,000-49,999	
82174	0.0	0.24	$100,000+	
89123	1.0	0.09	$100,000+	
60915	1.0	0.37	$50,000-74,999	
14063	0.0	NaN	$50,000-74,999	
75091	NaN	0.03	Not displayed	
56298	0.0	0.53	$25,000-49,999	
71693	1.0	0.30	$50,000-74,999	
109411	1.0	0.50	$25,000-49,999	

```
47532                     0.0               0.07  $25,000-49,999

       IncomeVerifiable  StatedMonthlyIncome                 LoanKey  \
33895              True          4720.833333  EB2336886216248833 39C59
9604               True          4333.333333  424536997290205 93A97D58
110363             True          3583.333333  6A433594122982891CB6D03
109947             True          5583.333333  BE3A370775781352264ED7F
52000              True          3166.666667  BF63369464189509318DF06
109394             True          3583.333333  752A3694990002462B3520E
82174              True         10000.000000  8CED3625429832515F3951B
89123              True          8333.333333  80573683813836708805051
60915              True          6041.666667  CCE13654512759159C17EFB
14063             False          5000.000000  D72234333671678578 94E59
75091              True          4500.000000  74E13380450969724CA3302
56298              True          3250.000000  A1E03700179128837B51B88
71693              True          4166.666667  323A3703458726045B65E7A
109411             True          2083.333333  F2FF33954406971803FDF20
47532              True          3083.333333  871A3596748144375B9DD24

       TotalProsperLoans  TotalProsperPaymentsBilled  OnTimeProsperPayments  \
33895                NaN                         NaN                    NaN
9604                 NaN                         NaN                    NaN
110363               NaN                         NaN                    NaN
109947               NaN                         NaN                    NaN
52000                1.0                         2.0                    2.0
109394               1.0                         9.0                    9.0
82174                2.0                        29.0                   28.0
89123                1.0                         6.0                    6.0
60915                1.0                         9.0                    8.0
14063                NaN                         NaN                    NaN
75091                NaN                         NaN                    NaN
56298                NaN                         NaN                    NaN
71693                NaN                         NaN                    NaN
109411               NaN                         NaN                    NaN
47532                NaN                         NaN                    NaN

       ProsperPaymentsLessThanOneMonthLate  ProsperPaymentsOneMonthPlusLate  \
33895                                  NaN                              NaN
9604                                   NaN                              NaN
110363                                 NaN                              NaN
109947                                 NaN                              NaN
52000                                  0.0                              0.0
109394                                 0.0                              0.0
82174                                  1.0                              0.0
89123                                  0.0                              0.0
60915                                  1.0                              0.0
14063                                  NaN                              NaN
```

```
75091                                  NaN                                  NaN
56298                                  NaN                                  NaN
71693                                  NaN                                  NaN
109411                                 NaN                                  NaN
47532                                  NaN                                  NaN

       ProsperPrincipalBorrowed  ProsperPrincipalOutstanding  \
33895                       NaN                          NaN
9604                        NaN                          NaN
110363                      NaN                          NaN
109947                      NaN                          NaN
52000                    4000.0                         0.00
109394                  17000.0                     13469.99
82174                   11500.0                      4434.48
89123                    4000.0                      3490.76
60915                    2600.0                         0.00
14063                       NaN                          NaN
75091                       NaN                          NaN
56298                       NaN                          NaN
71693                       NaN                          NaN
109411                      NaN                          NaN
47532                       NaN                          NaN

       ScorexChangeAtTimeOfListing  LoanCurrentDaysDelinquent  \
33895                          NaN                          0
9604                           NaN                          0
110363                         NaN                          0
109947                         NaN                          0
52000                          NaN                          0
109394                         NaN                          0
82174                        -58.0                          0
89123                        -15.0                          0
60915                        181.0                          0
14063                          NaN                        679
75091                          NaN                       2161
56298                          NaN                          0
71693                          NaN                          0
109411                         NaN                          0
47532                          NaN                          0

       LoanFirstDefaultedCycleNumber  LoanMonthsSinceOrigination  LoanNumber  \
33895                            NaN                           8       95928
9604                             NaN                           3      122047
110363                           NaN                          44       43689
109947                           NaN                           1      133946
52000                            NaN                           6      103296
109394                           NaN                           5      105662
```

```
82174                            NaN                          32       51674
89123                            NaN                          10       91337
60915                            NaN                          21       68281
14063                           32.0                          66       37184
75091                           18.0                          85        7277
56298                            NaN                           3      120574
71693                            NaN                           2      128719
109411                           NaN                          79       17780
47532                            NaN                          42       44301

        LoanOriginalAmount  LoanOriginationDate LoanOriginationQuarter  \
33895                10000  2013-07-22 00:00:00                Q3 2013
9604                 15000  2013-12-16 00:00:00                Q4 2013
110363                5000  2010-07-27 00:00:00                Q3 2010
109947               14000  2014-02-28 00:00:00                Q1 2014
52000                 6000  2013-09-30 00:00:00                Q3 2013
109394               10000  2013-10-11 00:00:00                Q4 2013
82174                 4000  2011-07-27 00:00:00                Q3 2011
89123                 6000  2013-05-24 00:00:00                Q2 2013
60915                15000  2012-06-20 00:00:00                Q2 2012
14063                 8000  2008-09-30 00:00:00                Q3 2008
75091                 3000  2007-02-09 00:00:00                Q1 2007
56298                10000  2013-12-06 00:00:00                Q4 2013
71693                15000  2014-01-24 00:00:00                Q1 2014
109411                5000  2007-08-07 00:00:00                Q3 2007
47532                 1300  2010-09-07 00:00:00                Q3 2010

                      MemberKey  MonthlyLoanPayment  LP_CustomerPayments  \
33895   D28035837722547960E207C              293.87            2050.2242
9604    C31D3537811782590E61906              380.49             753.6033
110363  265234893528714 40CB216A             192.38            5962.9500
109947  952B3580410639421391919              362.40               0.0000
52000   EBEA3579314049024D578A0              139.58             695.6003
109394  BB6A3562222340616623 51D             367.32            1464.0334
82174   DF793406184956488AA1909              172.56            5733.7300
89123   44FC355667301770463263D              212.10            1908.9000
60915   CF263390333438710BBEB87              517.71           16740.3100
14063   BCBB34302950810386E62CA              287.22            7321.6200
75091   80063377865979767265722              101.66            1297.3600
56298   1AE93698951047469A3CFB0              379.32             752.7496
71693   D6963703017029893 0475D5             493.86             489.1792
109411  B5293395214738274EA7459              209.53            7551.4500
47532   511D34935127909397972FB               58.81            1783.8200

        LP_CustomerPrincipalPayments  LP_InterestandFees  LP_ServiceFees  \
33895                         619.01           1431.2142          -57.38
9604                          305.94            447.6633          -25.35
```

110363	5000.00	962.9500	-42.69
109947	0.00	0.0000	0.00
52000	358.75	336.8503	-24.25
109394	844.20	619.8334	-32.64
82174	4000.00	1733.7300	-54.99
89123	1231.66	677.2400	-41.32
60915	15000.00	1740.3100	-118.48
14063	5185.83	2135.7900	-125.12
75091	930.30	367.0600	-14.23
56298	397.03	355.7196	-16.82
71693	348.75	140.4292	-12.74
109411	5000.00	2551.4500	-87.99
47532	1300.00	483.8200	-13.84

	LP_CollectionFees	LP_GrossPrincipalLoss	LP_NetPrincipalLoss	\
33895	0.00	0.00	0.00	
9604	0.00	0.00	0.00	
110363	0.00	0.00	0.00	
109947	0.00	0.00	0.00	
52000	0.00	0.00	0.00	
109394	0.00	0.00	0.00	
82174	0.00	0.00	0.00	
89123	0.00	0.00	0.00	
60915	0.00	0.00	0.00	
14063	-360.21	2335.69	1439.02	
75091	0.00	2069.71	2069.70	
56298	0.00	0.00	0.00	
71693	0.00	0.00	0.00	
109411	0.00	0.00	0.00	
47532	0.00	0.00	0.00	

	LP_NonPrincipalRecoverypayments	PercentFunded	Recommendations	\
33895	0.00	1.0	0	
9604	0.00	1.0	0	
110363	0.00	1.0	0	
109947	0.00	1.0	0	
52000	0.00	1.0	0	
109394	0.00	1.0	0	
82174	0.00	1.0	0	
89123	0.00	1.0	0	
60915	0.00	1.0	0	
14063	203.41	1.0	0	
75091	0.00	1.0	0	
56298	0.00	1.0	0	
71693	0.00	1.0	0	
109411	0.00	1.0	0	
47532	0.00	1.0	0	

	InvestmentFromFriendsCount	InvestmentFromFriendsAmount	Investors
33895	0	0.0	1
9604	0	0.0	1
110363	1	26.0	167
109947	0	0.0	1
52000	0	0.0	1
109394	0	0.0	1
82174	0	0.0	5
89123	0	0.0	37
60915	0	0.0	126
14063	0	0.0	168
75091	0	0.0	110
56298	0	0.0	1
71693	0	0.0	1
109411	0	0.0	47
47532	0	0.0	41

[12]:
```
# checking for data duplication based on ListingNumber
Loan_Data['ListingNumber'].duplicated().any()
```

[12]:
```
True
```

[13]:
```
# dataset's shape
Loan_Data.shape
```

[13]:
```
(113937, 81)
```

[14]:
```
Loan_Data['EmploymentStatus'].value_counts()
```

[14]:
```
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
```

[15]:
```
Loan_Data['ProsperRating (numeric)'].isnull().sum()
```

[15]:
```
29084
```

[16]:
```
# getting descriptive statistics of the numeric variables
Loan_Data.describe()
```

[16]:

	ListingNumber	Term	BorrowerAPR	BorrowerRate
count	1.139370e+05	113937.000000	113912.000000	113937.000000
mean	6.278857e+05	40.830248	0.218828	0.192764
std	3.280762e+05	10.436212	0.080364	0.074818
min	4.000000e+00	12.000000	0.006530	0.000000
25%	4.009190e+05	36.000000	0.156290	0.134000
50%	6.005540e+05	36.000000	0.209760	0.184000
75%	8.926340e+05	36.000000	0.283810	0.250000
max	1.255725e+06	60.000000	0.512290	0.497500

	LenderYield	EstimatedEffectiveYield	EstimatedLoss	EstimatedReturn
count	113937.000000	84853.000000	84853.000000	84853.000000
mean	0.182701	0.168661	0.080306	0.096068
std	0.074516	0.068467	0.046764	0.030403
min	-0.010000	-0.182700	0.004900	-0.182700
25%	0.124200	0.115670	0.042400	0.074080
50%	0.173000	0.161500	0.072400	0.091700
75%	0.240000	0.224300	0.112000	0.116600
max	0.492500	0.319900	0.366000	0.283700

	ProsperRating (numeric)	ProsperScore	ListingCategory (numeric)
count	84853.000000	84853.000000	113937.000000
mean	4.072243	5.950067	2.774209
std	1.673227	2.376501	3.996797
min	1.000000	1.000000	0.000000
25%	3.000000	4.000000	1.000000
50%	4.000000	6.000000	1.000000
75%	5.000000	8.000000	3.000000
max	7.000000	11.000000	20.000000

	EmploymentStatusDuration	CreditScoreRangeLower	CreditScoreRangeUpper
count	106312.000000	113346.000000	113346.000000
mean	96.071582	685.567731	704.567731
std	94.480605	66.458275	66.458275
min	0.000000	0.000000	19.000000
25%	26.000000	660.000000	679.000000
50%	67.000000	680.000000	699.000000
75%	137.000000	720.000000	739.000000
max	755.000000	880.000000	899.000000

	CurrentCreditLines	OpenCreditLines	TotalCreditLinespast7years
count	106333.000000	106333.000000	113240.000000
mean	10.317192	9.260164	26.754539
std	5.457866	5.022644	13.637871
min	0.000000	0.000000	2.000000
25%	7.000000	6.000000	17.000000
50%	10.000000	9.000000	25.000000

```
75%             13.000000        12.000000                   35.000000
max             59.000000        54.000000                  136.000000

       OpenRevolvingAccounts  OpenRevolvingMonthlyPayment  \
count           113937.00000                113937.000000
mean                 6.96979                   398.292161
std                  4.63097                   447.159711
min                  0.00000                     0.000000
25%                  4.00000                   114.000000
50%                  6.00000                   271.000000
75%                  9.00000                   525.000000
max                 51.00000                 14985.000000

       InquiriesLast6Months  TotalInquiries  CurrentDelinquencies  \
count         113240.000000   112778.000000         113240.000000
mean               1.435085        5.584405              0.592052
std                2.437507        6.429946              1.978707
min                0.000000        0.000000              0.000000
25%                0.000000        2.000000              0.000000
50%                1.000000        4.000000              0.000000
75%                2.000000        7.000000              0.000000
max              105.000000      379.000000             83.000000

       AmountDelinquent  DelinquenciesLast7Years  PublicRecordsLast10Years  \
count     106315.000000            112947.000000             113240.000000
mean         984.507059                 4.154984                  0.312646
std         7158.270157                10.160216                  0.727868
min            0.000000                 0.000000                  0.000000
25%            0.000000                 0.000000                  0.000000
50%            0.000000                 0.000000                  0.000000
75%            0.000000                 3.000000                  0.000000
max       463881.000000                99.000000                 38.000000

       PublicRecordsLast12Months  RevolvingCreditBalance  BankcardUtilization  \
count              106333.000000            1.063330e+05        106333.000000
mean                    0.015094            1.759871e+04             0.561309
std                     0.154092            3.293640e+04             0.317918
min                     0.000000            0.000000e+00             0.000000
25%                     0.000000            3.121000e+03             0.310000
50%                     0.000000            8.549000e+03             0.600000
75%                     0.000000            1.952100e+04             0.840000
max                    20.000000            1.435667e+06             5.950000

       AvailableBankcardCredit    TotalTrades  \
count            106393.000000  106393.000000
mean              11210.225447      23.230034
std               19818.361309      11.871311
```

```
min                   0.000000       0.000000
25%                 880.000000      15.000000
50%                4100.000000      22.000000
75%               13180.000000      30.000000
max              646285.000000     126.000000

       TradesNeverDelinquent (percentage)  TradesOpenedLast6Months  \
count                       106393.000000            106393.000000
mean                             0.885897                 0.802327
std                              0.148179                 1.097637
min                              0.000000                 0.000000
25%                              0.820000                 0.000000
50%                              0.940000                 0.000000
75%                              1.000000                 1.000000
max                              1.000000                20.000000

       DebtToIncomeRatio  StatedMonthlyIncome  TotalProsperLoans  \
count      105383.000000         1.139370e+05       22085.000000
mean            0.275947         5.608026e+03           1.421100
std             0.551759         7.478497e+03           0.764042
min             0.000000         0.000000e+00           0.000000
25%             0.140000         3.200333e+03           1.000000
50%             0.220000         4.666667e+03           1.000000
75%             0.320000         6.825000e+03           2.000000
max            10.010000         1.750003e+06           8.000000

       TotalProsperPaymentsBilled  OnTimeProsperPayments  \
count                22085.000000           22085.000000
mean                    22.934345              22.271949
std                     19.249584              18.830425
min                      0.000000               0.000000
25%                      9.000000               9.000000
50%                     16.000000              15.000000
75%                     33.000000              32.000000
max                    141.000000             141.000000

       ProsperPaymentsLessThanOneMonthLate  ProsperPaymentsOneMonthPlusLate  \
count                         22085.000000                     22085.000000
mean                              0.613629                         0.048540
std                               2.446827                         0.556285
min                               0.000000                         0.000000
25%                               0.000000                         0.000000
50%                               0.000000                         0.000000
75%                               0.000000                         0.000000
max                              42.000000                        21.000000

       ProsperPrincipalBorrowed  ProsperPrincipalOutstanding  \
```

```
count              22085.000000                 22085.000000
mean                8472.311961                  2930.313906
std                 7395.507650                  3806.635075
min                    0.000000                     0.000000
25%                 3500.000000                     0.000000
50%                 6000.000000                  1626.550000
75%                11000.000000                  4126.720000
max                72499.000000                 23450.950000

       ScorexChangeAtTimeOfListing  LoanCurrentDaysDelinquent  \
count                 18928.000000              113937.000000
mean                     -3.223214                 152.816539
std                      50.063567                 466.320254
min                    -209.000000                   0.000000
25%                     -35.000000                   0.000000
50%                      -3.000000                   0.000000
75%                      25.000000                   0.000000
max                     286.000000                2704.000000

       LoanFirstDefaultedCycleNumber  LoanMonthsSinceOrigination  \
count                   16952.000000               113937.000000
mean                       16.268464                   31.896882
std                         9.005898                   29.974184
min                         0.000000                    0.000000
25%                         9.000000                    6.000000
50%                        14.000000                   21.000000
75%                        22.000000                   65.000000
max                        44.000000                  100.000000

          LoanNumber  LoanOriginalAmount  MonthlyLoanPayment  \
count  113937.000000        113937.00000       113937.000000
mean    69444.474271          8337.01385          272.475783
std     38930.479610          6245.80058          192.697812
min         1.000000          1000.00000            0.000000
25%     37332.000000          4000.00000          131.620000
50%     68599.000000          6500.00000          217.740000
75%    101901.000000         12000.00000          371.580000
max    136486.000000         35000.00000         2251.510000

       LP_CustomerPayments  LP_CustomerPrincipalPayments  LP_InterestandFees  \
count        113937.000000                 113937.000000       113937.000000
mean           4183.079489                   3105.536588         1077.542901
std            4790.907234                   4069.527670         1183.414168
min              -2.349900                      0.000000           -2.349900
25%            1005.760000                    500.890000          274.870000
50%            2583.830000                   1587.500000          700.840100
75%            5548.400000                   4000.000000         1458.540000
```

```
max           40702.390000                   35000.000000        15617.030000

       LP_ServiceFees  LP_CollectionFees  LP_GrossPrincipalLoss  \
count   113937.000000      113937.000000          113937.000000
mean       -54.725641         -14.242698             700.446342
std         60.675425         109.232758            2388.513831
min       -664.870000       -9274.750000             -94.200000
25%        -73.180000           0.000000               0.000000
50%        -34.440000           0.000000               0.000000
75%        -13.920000           0.000000               0.000000
max         32.060000           0.000000           25000.000000

       LP_NetPrincipalLoss  LP_NonPrincipalRecoverypayments  PercentFunded  \
count        113937.000000                    113937.000000  113937.000000
mean            681.420499                        25.142686       0.998584
std            2357.167068                       275.657937       0.017919
min            -954.550000                         0.000000       0.700000
25%               0.000000                         0.000000       1.000000
50%               0.000000                         0.000000       1.000000
75%               0.000000                         0.000000       1.000000
max           25000.000000                     21117.900000       1.012500

       Recommendations  InvestmentFromFriendsCount  \
count    113937.000000                113937.000000
mean          0.048027                     0.023460
std           0.332353                     0.232412
min           0.000000                     0.000000
25%           0.000000                     0.000000
50%           0.000000                     0.000000
75%           0.000000                     0.000000
max          39.000000                    33.000000

       InvestmentFromFriendsAmount      Investors
count                113937.000000  113937.000000
mean                     16.550751      80.475228
std                     294.545422     103.239020
min                       0.000000       1.000000
25%                       0.000000       2.000000
50%                       0.000000      44.000000
75%                       0.000000     115.000000
max                   25000.000000    1189.000000
```

[17]:
```
# Check for null values
Loan_Data.isnull().sum()
```

[17]:
```
ListingKey                          0
ListingNumber                       0
```

ListingCreationDate	0
CreditGrade	84984
Term	0
LoanStatus	0
ClosedDate	58848
BorrowerAPR	25
BorrowerRate	0
LenderYield	0
EstimatedEffectiveYield	29084
EstimatedLoss	29084
EstimatedReturn	29084
ProsperRating (numeric)	29084
ProsperRating (Alpha)	29084
ProsperScore	29084
ListingCategory (numeric)	0
BorrowerState	5515
Occupation	3588
EmploymentStatus	2255
EmploymentStatusDuration	7625
IsBorrowerHomeowner	0
CurrentlyInGroup	0
GroupKey	100596
DateCreditPulled	0
CreditScoreRangeLower	591
CreditScoreRangeUpper	591
FirstRecordedCreditLine	697
CurrentCreditLines	7604
OpenCreditLines	7604
TotalCreditLinespast7years	697
OpenRevolvingAccounts	0
OpenRevolvingMonthlyPayment	0
InquiriesLast6Months	697
TotalInquiries	1159
CurrentDelinquencies	697
AmountDelinquent	7622
DelinquenciesLast7Years	990
PublicRecordsLast10Years	697
PublicRecordsLast12Months	7604
RevolvingCreditBalance	7604
BankcardUtilization	7604
AvailableBankcardCredit	7544
TotalTrades	7544
TradesNeverDelinquent (percentage)	7544
TradesOpenedLast6Months	7544
DebtToIncomeRatio	8554
IncomeRange	0
IncomeVerifiable	0

```
StatedMonthlyIncome                     0
LoanKey                                 0
TotalProsperLoans                   91852
TotalProsperPaymentsBilled          91852
OnTimeProsperPayments               91852
ProsperPaymentsLessThanOneMonthLate 91852
ProsperPaymentsOneMonthPlusLate     91852
ProsperPrincipalBorrowed            91852
ProsperPrincipalOutstanding         91852
ScorexChangeAtTimeOfListing         95009
LoanCurrentDaysDelinquent               0
LoanFirstDefaultedCycleNumber       96985
LoanMonthsSinceOrigination              0
LoanNumber                              0
LoanOriginalAmount                      0
LoanOriginationDate                     0
LoanOriginationQuarter                  0
MemberKey                               0
MonthlyLoanPayment                      0
LP_CustomerPayments                     0
LP_CustomerPrincipalPayments            0
LP_InterestandFees                      0
LP_ServiceFees                          0
LP_CollectionFees                       0
LP_GrossPrincipalLoss                   0
LP_NetPrincipalLoss                     0
LP_NonPrincipalRecoverypayments         0
PercentFunded                           0
Recommendations                         0
InvestmentFromFriendsCount              0
InvestmentFromFriendsAmount             0
Investors                               0
dtype: int64
```

1.1.1 Data Assessment Summary:

1. The dataset contans a lot of null values
2. Too many variables for the Purpose of the investigation
3. There are ListingNumber duplication
4. Erroneous datatypes for `TotalTrades, TotalInquiries, and ListingCategory (numeric)`
5. There are null values in ProsperRating (numeric) column
6. The time, day, month and year are combined in `ListingCreationDate` column.

Data Cleaning 1. Fill in the null values of occupation and DebtToIncomeRatio 2. Pick out features of intrest 3. Drop the rows with ListingNumber duplication 4. Change `TotalTrades and TotalInquiries` datatypes to Integer, and `ListingCategory (numeric)` to Category 5. Drop rows with no ProsperRating data 6. Separate the time, day, month and year into four columns

from `ListingCreationDate` column and give names to the months

[18]:
```
# Pick out features of intrest
#Code:
features_intrest=␣
 ↪['ListingNumber','ListingCreationDate','LoanOriginalAmount','LoanStatus','ListingCategory␣
 ↪(numeric)','BorrowerState','BorrowerAPR','BorrowerRate','StatedMonthlyIncome',␣
 ↪'ProsperRating (Alpha)',
       ␣
 ↪'Occupation','Term','EmploymentStatus','TotalInquiries','DebtToIncomeRatio','MonthlyLoanPay
Loan_Data_subset= Loan_Data[features_intrest]
```

[19]:
```
#Test: data information
Loan_Data_subset.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 113937 entries, 0 to 113936
Data columns (total 18 columns):
 #   Column                     Non-Null Count   Dtype
---  ------                     --------------   -----
 0   ListingNumber              113937 non-null  int64
 1   ListingCreationDate        113937 non-null  object
 2   LoanOriginalAmount         113937 non-null  int64
 3   LoanStatus                 113937 non-null  object
 4   ListingCategory (numeric)  113937 non-null  int64
 5   BorrowerState              108422 non-null  object
 6   BorrowerAPR                113912 non-null  float64
 7   BorrowerRate               113937 non-null  float64
 8   StatedMonthlyIncome        113937 non-null  float64
 9   ProsperRating (Alpha)      84853 non-null   object
 10  Occupation                 110349 non-null  object
 11  Term                       113937 non-null  int64
 12  EmploymentStatus           111682 non-null  object
 13  TotalInquiries             112778 non-null  float64
 14  DebtToIncomeRatio          105383 non-null  float64
 15  MonthlyLoanPayment         113937 non-null  float64
 16  TotalTrades                106393 non-null  float64
 17  Investors                  113937 non-null  int64
dtypes: float64(7), int64(5), object(6)
memory usage: 15.6+ MB
```

[20]:
```
#Test: sample of the data
Loan_Data_subset.sample(15)
```

[20]:

	ListingNumber	ListingCreationDate	LoanOriginalAmount	\
54238	862294	2013-08-06 12:40:15.030000000	10000	
65890	508787	2011-05-26 07:59:41.607000000	5000	

```
45546         439550  2009-12-22 10:20:14.427000000               7000
45651         533279  2011-10-17 12:17:09.480000000               6000
27531         633000  2012-09-04 10:27:43.737000000               3000
69489          90533  2007-01-29 19:02:15.443000000               2500
46700          90486  2007-01-29 17:11:45.843000000              20000
54965         609069  2012-07-09 11:13:51.767000000               5000
45876         990877  2013-11-12 17:54:16.297000000              15000
30840         151309  2007-06-13 05:24:14.813000000               5000
83662         812868  2013-06-18 05:25:34.633000000               4000
58793         553439  2012-01-22 12:48:32.887000000              11000
44305         585007  2012-05-02 09:53:35.307000000               2000
67214         486481  2010-12-01 21:45:42.483000000               4000
30733         907255  2013-09-18 09:30:18.180000000               7000

       LoanStatus  ListingCategory (numeric) BorrowerState  BorrowerAPR  \
54238     Current                          1           MI      0.21025
65890     Current                          2           IL      0.30532
45546   Completed                          5           MO      0.29334
45651  Chargedoff                          1           FL      0.29510
27531     Current                          1           CA      0.35797
69489  Chargedoff                          0           AZ      0.22744
46700   Defaulted                          0          NaN      0.16717
54965   Completed                         18           NJ      0.08829
45876     Current                          1           MI      0.13189
30840  Chargedoff                          0           OR      0.25757
83662     Current                          1           MA      0.32538
58793     Current                          1           TX      0.17359
44305     Current                          1           TX      0.35797
67214   Completed                          7           IL      0.33097
30733     Current                          1           MA      0.26333

       BorrowerRate  StatedMonthlyIncome ProsperRating (Alpha)  \
54238        0.1734          2577.250000                     C
65890        0.2699          3166.666667                     D
45546        0.2700          5166.666667                     D
45651        0.2599          3333.333333                     D
27531        0.3177          3000.000000                    HR
69489        0.2200          6666.666667                   NaN
46700        0.1600          2775.000000                   NaN
54965        0.0749         14583.333333                    AA
45876        0.1039          6694.583333                     A
30840        0.2500          1968.583333                   NaN
83662        0.2859          2875.000000                     E
58793        0.1449          5000.000000                     B
44305        0.3177          4875.000000                    HR
67214        0.2950          7083.333333                     D
30733        0.2379          5166.666667                     D
```

	Occupation	Term	EmploymentStatus	\
54238	Other	36	Other	
65890	Truck Driver	36	Employed	
45546	Laborer	36	Full-time	
45651	Professional	36	Employed	
27531	Sales - Retail	36	Full-time	
69489	Professional	36	Not available	
46700	Other	36	Not available	
54965	Sales - Commission	36	Employed	
45876	Computer Programmer	36	Employed	
30840	Clerical	36	Full-time	
83662	Laborer	36	Employed	
58793	Nurse (RN)	36	Employed	
44305	Police Officer/Correction Officer	36	Employed	
67214	Professional	36	Employed	
30733	Sales - Retail	60	Employed	

	TotalInquiries	DebtToIncomeRatio	MonthlyLoanPayment	TotalTrades	\
54238	2.0	0.22	358.22	12.0	
65890	2.0	0.41	204.10	34.0	
45546	4.0	0.33	285.78	23.0	
45651	8.0	0.08	241.71	13.0	
27531	3.0	0.24	130.28	4.0	
69489	46.0	0.54	95.48	NaN	
46700	20.0	0.56	703.14	NaN	
54965	1.0	0.08	155.51	14.0	
45876	12.0	0.20	486.76	18.0	
30840	18.0	0.50	198.80	23.0	
83662	1.0	0.39	166.73	17.0	
58793	1.0	0.32	378.58	39.0	
44305	5.0	0.49	86.85	38.0	
67214	0.0	0.18	168.71	18.0	
30733	1.0	0.46	200.52	39.0	

	Investors
54238	1
65890	97
45546	351
45651	42
27531	37
69489	42
46700	189
54965	112
45876	1
30840	60
83662	1

```
58793         19
44305         13
67214         46
30733          1
```

[21]:
```
# Drop the rows with ListingNumber duplication
#Code:
Loan_Data_subset.drop_duplicates(inplace=True)
```

```
<ipython-input-21-a90e2ed42f0a>:3: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  Loan_Data_subset.drop_duplicates(inplace=True)
```

[22]:
```
#Test: data information
Loan_Data_subset.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 113066 entries, 0 to 113936
Data columns (total 18 columns):
 #   Column                      Non-Null Count   Dtype
---  ------                      --------------   -----
 0   ListingNumber               113066 non-null  int64
 1   ListingCreationDate         113066 non-null  object
 2   LoanOriginalAmount          113066 non-null  int64
 3   LoanStatus                  113066 non-null  object
 4   ListingCategory (numeric)   113066 non-null  int64
 5   BorrowerState               107551 non-null  object
 6   BorrowerAPR                 113041 non-null  float64
 7   BorrowerRate                113066 non-null  float64
 8   StatedMonthlyIncome         113066 non-null  float64
 9   ProsperRating (Alpha)       83982 non-null   object
 10  Occupation                  109537 non-null  object
 11  Term                        113066 non-null  int64
 12  EmploymentStatus            110811 non-null  object
 13  TotalInquiries              111907 non-null  float64
 14  DebtToIncomeRatio           104594 non-null  float64
 15  MonthlyLoanPayment          113066 non-null  float64
 16  TotalTrades                 105522 non-null  float64
 17  Investors                   113066 non-null  int64
dtypes: float64(7), int64(5), object(6)
memory usage: 16.4+ MB
```

[23]:
```
#Test: Check for duplication
Loan_Data_subset.duplicated().any()
```

[23]: False

[24]:
```
#Drop rows with no ProsperRating data
#Code:
Loan_Data_subset= Loan_Data_subset[Loan_Data_subset['ProsperRating (Alpha)'].
 ↪notnull()]
```

[25]:
```
#Test: data information
Loan_Data_subset.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 83982 entries, 1 to 113936
Data columns (total 18 columns):
 #   Column                      Non-Null Count  Dtype
---  ------                      --------------  -----
 0   ListingNumber               83982 non-null  int64
 1   ListingCreationDate         83982 non-null  object
 2   LoanOriginalAmount          83982 non-null  int64
 3   LoanStatus                  83982 non-null  object
 4   ListingCategory (numeric)   83982 non-null  int64
 5   BorrowerState               83982 non-null  object
 6   BorrowerAPR                 83982 non-null  float64
 7   BorrowerRate                83982 non-null  float64
 8   StatedMonthlyIncome         83982 non-null  float64
 9   ProsperRating (Alpha)       83982 non-null  object
 10  Occupation                  82708 non-null  object
 11  Term                        83982 non-null  int64
 12  EmploymentStatus            83982 non-null  object
 13  TotalInquiries              83982 non-null  float64
 14  DebtToIncomeRatio           76768 non-null  float64
 15  MonthlyLoanPayment          83982 non-null  float64
 16  TotalTrades                 83982 non-null  float64
 17  Investors                   83982 non-null  int64
dtypes: float64(7), int64(5), object(6)
memory usage: 12.2+ MB
```

[26]:
```
#Test: Check for null values
Loan_Data_subset['ProsperRating (Alpha)'].isnull().any()
```

[26]: False

[27]:
```
#Fill in the null values of occupation and DebtToIncomeRatio
#Code 1:
Loan_Data_subset['Occupation'] = Loan_Data_subset['Occupation'].
 ↪fillna('Unknown')
```

[28]:
```
#Code 2:
Loan_Data_subset['DebtToIncomeRatio'] = Loan_Data_subset['DebtToIncomeRatio'].
 ↪fillna(Loan_Data_subset['DebtToIncomeRatio'].mean())
```

[29]:
```
# Test: data information
Loan_Data_subset.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 83982 entries, 1 to 113936
Data columns (total 18 columns):
 #   Column                      Non-Null Count  Dtype
---  ------                      --------------  -----
 0   ListingNumber               83982 non-null  int64
 1   ListingCreationDate         83982 non-null  object
 2   LoanOriginalAmount          83982 non-null  int64
 3   LoanStatus                  83982 non-null  object
 4   ListingCategory (numeric)   83982 non-null  int64
 5   BorrowerState               83982 non-null  object
 6   BorrowerAPR                 83982 non-null  float64
 7   BorrowerRate                83982 non-null  float64
 8   StatedMonthlyIncome         83982 non-null  float64
 9   ProsperRating (Alpha)       83982 non-null  object
 10  Occupation                  83982 non-null  object
 11  Term                        83982 non-null  int64
 12  EmploymentStatus            83982 non-null  object
 13  TotalInquiries              83982 non-null  float64
 14  DebtToIncomeRatio           83982 non-null  float64
 15  MonthlyLoanPayment          83982 non-null  float64
 16  TotalTrades                 83982 non-null  float64
 17  Investors                   83982 non-null  int64
dtypes: float64(7), int64(5), object(6)
memory usage: 12.2+ MB
```

[30]:
```
# Test 1: Check for null values in Occupation
Loan_Data_subset['Occupation'].isnull().any()
```

[30]:
```
False
```

[31]:
```
# Test 2: Check for null values in DebtToIncomeRatio
Loan_Data_subset['DebtToIncomeRatio'].isnull().any()
```

[31]:
```
False
```

[32]:
```
#Change TotalTrades and TotalInquiries datatypes to Integer, and␣
 ↪ListingCategory (numeric) to Category
#Code 1:
```

```
Loan_Data_subset['TotalInquiries'] = Loan_Data_subset['TotalInquiries'].
 ↪astype(int)
Loan_Data_subset['TotalTrades'] = Loan_Data_subset['TotalTrades'].astype(int)
```

[33]:
```
# Code 2:
Loan_Data_subset['ListingCategory (numeric)'] =␣
 ↪Loan_Data_subset['ListingCategory (numeric)'].astype("category")
```

[34]:
```
# Test: data information
Loan_Data_subset.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 83982 entries, 1 to 113936
Data columns (total 18 columns):
 #   Column                      Non-Null Count  Dtype
---  ------                      --------------  -----
 0   ListingNumber               83982 non-null  int64
 1   ListingCreationDate         83982 non-null  object
 2   LoanOriginalAmount          83982 non-null  int64
 3   LoanStatus                  83982 non-null  object
 4   ListingCategory (numeric)   83982 non-null  category
 5   BorrowerState               83982 non-null  object
 6   BorrowerAPR                 83982 non-null  float64
 7   BorrowerRate                83982 non-null  float64
 8   StatedMonthlyIncome         83982 non-null  float64
 9   ProsperRating (Alpha)       83982 non-null  object
 10  Occupation                  83982 non-null  object
 11  Term                        83982 non-null  int64
 12  EmploymentStatus            83982 non-null  object
 13  TotalInquiries              83982 non-null  int64
 14  DebtToIncomeRatio           83982 non-null  float64
 15  MonthlyLoanPayment          83982 non-null  float64
 16  TotalTrades                 83982 non-null  int64
 17  Investors                   83982 non-null  int64
dtypes: category(1), float64(5), int64(6), object(6)
memory usage: 11.6+ MB
```

[35]:
```
# Test 1 and 2: Check for TotalTrades, TotalInquiries and ListingCategory␣
 ↪(numeric) datatypes
print(Loan_Data_subset['TotalInquiries'].dtypes)
print(Loan_Data_subset['TotalTrades'].dtypes)
print(Loan_Data_subset['ListingCategory (numeric)'].dtypes)
```

```
int64
int64
category
```

[36]:
```
# Separate the time, day, month and year into four columns from␣
↪`ListingCreationDate` column and give names to the months
#Code 1: Splitting into year
Loan_Data_subset['Year']=Loan_Data_subset['ListingCreationDate'].apply(lambda x:
↪ x.split("-")[0]).astype(str)
```

[37]:
```
#Test 1: sample of the Year column
Loan_Data_subset['Year'].sample(10)
```

[37]:
```
76776     2012
64584     2011
109563    2013
15379     2013
2657      2013
59218     2013
89244     2013
23962     2013
56550     2013
36201     2013
Name: Year, dtype: object
```

[38]:
```
#Code 2.1: Splitting into month
Loan_Data_subset['Month'] = Loan_Data_subset['ListingCreationDate'].
↪apply(lambda x: x.split("-")[1]).astype(str)
```

[39]:
```
#Test 2.1: sample of the Month column
Loan_Data_subset['Month'].sample(10)
```

[39]:
```
20296     12
51035     10
106823    06
19009     07
61804     09
57360     05
10435     07
28216     06
109028    04
105312    12
Name: Month, dtype: object
```

[40]:
```
#Code 2.2: give names to the months instead of the numbers
Loan_Data_subset['Month'].
↪replace(['01','02','03','04','05','06','07','08','09','10','11','12'],['Jan','Feb','Mar','A
↪= True)
```

[41]:
```
#Test 2.2: sample of the Month column
Loan_Data_subset['Month'].sample(10)
```

[41]:
```
31645      May
39783      Oct
71127      Oct
25023      Apr
19046      Jul
81181     Sept
37326     Sept
109576     Nov
64702      Oct
107145     Dec
Name: Month, dtype: object
```

[42]:
```
#Code:
Loan_Data_subset['ListingCreationDate'] =␣
↪Loan_Data_subset['ListingCreationDate'].apply(lambda x: x.split("-")[2]).
↪astype(str)
```

[43]:
```
#Code 3:
Loan_Data_subset['Day'] = Loan_Data_subset['ListingCreationDate'].apply(lambda␣
↪x: x.split(" ")[0]).astype(str)
```

[44]:
```
#Test 3: sample of Day column
Loan_Data_subset['Day'].sample(5)
```

[44]:
```
1308      10
32073     17
92650     04
73556     21
109888    08
Name: Day, dtype: object
```

[45]:
```
#Code 4:
Loan_Data_subset['Time'] = Loan_Data_subset['ListingCreationDate'].apply(lambda␣
↪x: x.split(" ")[1]).astype(str)
```

[46]:
```
#Test 4: sample of Time column
Loan_Data_subset['Time'].sample(5)
```

[46]:
```
101109    15:23:11.650000000
100579    11:06:28.300000000
61329     17:10:04.737000000
113813    03:38:02.983000000
109531    10:18:11.703000000
Name: Time, dtype: object
```

[47]:
```
# Code: Dropping the ListingCreationDate column
Loan_Data_subset.drop(columns = ['ListingCreationDate'],inplace = True)
```

[48]:
```
#Test: data information
Loan_Data_subset.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 83982 entries, 1 to 113936
Data columns (total 21 columns):
 #   Column                      Non-Null Count  Dtype
---  ------                      --------------  -----
 0   ListingNumber               83982 non-null  int64
 1   LoanOriginalAmount          83982 non-null  int64
 2   LoanStatus                  83982 non-null  object
 3   ListingCategory (numeric)   83982 non-null  category
 4   BorrowerState               83982 non-null  object
 5   BorrowerAPR                 83982 non-null  float64
 6   BorrowerRate                83982 non-null  float64
 7   StatedMonthlyIncome         83982 non-null  float64
 8   ProsperRating (Alpha)       83982 non-null  object
 9   Occupation                  83982 non-null  object
 10  Term                        83982 non-null  int64
 11  EmploymentStatus            83982 non-null  object
 12  TotalInquiries              83982 non-null  int64
 13  DebtToIncomeRatio           83982 non-null  float64
 14  MonthlyLoanPayment          83982 non-null  float64
 15  TotalTrades                 83982 non-null  int64
 16  Investors                   83982 non-null  int64
 17  Year                        83982 non-null  object
 18  Month                       83982 non-null  object
 19  Day                         83982 non-null  object
 20  Time                        83982 non-null  object
dtypes: category(1), float64(5), int64(6), object(9)
memory usage: 13.5+ MB
```

Exploratory Data Analysis (EDA)

[49]:
```
#Descriptive Statistics
Loan_Data_subset.describe()
```

[49]:

	ListingNumber	LoanOriginalAmount	BorrowerAPR	BorrowerRate	\
count	8.398200e+04	83982.000000	83982.000000	83982.00000	
mean	7.712395e+05	9061.224381	0.226945	0.19630	
std	2.359100e+05	6279.649648	0.080047	0.07475	
min	4.162750e+05	1000.000000	0.045830	0.04000	
25%	5.570608e+05	4000.000000	0.163610	0.13590	
50%	7.341785e+05	7500.000000	0.219450	0.18750	
75%	9.756778e+05	13500.000000	0.292540	0.25740	
max	1.255725e+06	35000.000000	0.423950	0.36000	

	StatedMonthlyIncome	Term	TotalInquiries	DebtToIncomeRatio	\

```
count         8.398200e+04  83982.000000    83982.000000       83982.000000
mean          5.930614e+03     42.462813        4.285514           0.258692
std           8.268432e+03     11.639032        3.828780           0.305687
min           0.000000e+00     12.000000        0.000000           0.000000
25%           3.426938e+03     36.000000        2.000000           0.160000
50%           5.000000e+03     36.000000        3.000000           0.240000
75%           7.083333e+03     60.000000        6.000000           0.310000
max           1.750003e+06     60.000000       78.000000          10.010000

       MonthlyLoanPayment   TotalTrades     Investors
count         83982.00000  83982.000000  83982.000000
mean            291.40139     23.925115     68.677788
std             186.47539     11.610432     95.318077
min               0.00000      1.000000      1.000000
25%             157.11250     15.000000      1.000000
50%             251.28000     23.000000     32.000000
75%             387.62000     31.000000     98.000000
max            2251.51000    122.000000   1189.000000
```

[50]:
```
# a general look at the cleaned dataset
Loan_Data_subset.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 83982 entries, 1 to 113936
Data columns (total 21 columns):
 #   Column                      Non-Null Count  Dtype
---  ------                      --------------  -----
 0   ListingNumber               83982 non-null  int64
 1   LoanOriginalAmount          83982 non-null  int64
 2   LoanStatus                  83982 non-null  object
 3   ListingCategory (numeric)   83982 non-null  category
 4   BorrowerState               83982 non-null  object
 5   BorrowerAPR                 83982 non-null  float64
 6   BorrowerRate                83982 non-null  float64
 7   StatedMonthlyIncome         83982 non-null  float64
 8   ProsperRating (Alpha)       83982 non-null  object
 9   Occupation                  83982 non-null  object
 10  Term                        83982 non-null  int64
 11  EmploymentStatus            83982 non-null  object
 12  TotalInquiries              83982 non-null  int64
 13  DebtToIncomeRatio           83982 non-null  float64
 14  MonthlyLoanPayment          83982 non-null  float64
 15  TotalTrades                 83982 non-null  int64
 16  Investors                   83982 non-null  int64
 17  Year                        83982 non-null  object
 18  Month                       83982 non-null  object
 19  Day                         83982 non-null  object
```

```
 20  Time                        83982 non-null  object
dtypes: category(1), float64(5), int64(6), object(9)
memory usage: 13.5+ MB
```

1.1.2 The Structure of the datset:

The dataset has 83982 loans and 21 features, such as BorrowerAPR, StatedMonthlyIncome, ProsperRating (Alpha), etc. ### The main features of interest in the dataset: The main features of interest are the ones that would help in predicting the borrowers' Annual Percentage Rate (APR) for the loans and in which way the employment status and debtToIncome ratio are connected with different metrics in the dataset. ### The features in the dataset that will help support the investigation into the features of interest: It is anticipated that the total loan amount would affect the loan's APR negatively, as well as the borrowers stated monthly income, loan term, Prosper rating and employment status. Moreover, Employment status and occupation would have an impact on the ProsperRating and loan Amount.

Univariate Exploration

[51]:
```
# Figuring out the ListingNumber for each year
years = ['2009','2010','2011','2012','2013','2014']
years_catg = pd.api.types.CategoricalDtype(ordered = True, categories = years)
Loan_Data_subset['Year'] = Loan_Data_subset['Year'].astype(years_catg)
Loan_Data_subset.groupby('Year')['ListingNumber'].count()
```

[51]:
```
Year
2009     2178
2010     5530
2011    11442
2012    19556
2013    34791
2014    10485
Name: ListingNumber, dtype: int64
```

[52]:
```
# Figuring out the ListingNumber for each month
months =␣
 ↪['Jan','Feb','Mar','Apr','May','Jun','Jul','Aug','Sept','Oct','Nov','Dec']
months_catg = pd.api.types.CategoricalDtype(ordered = True, categories = months)
Loan_Data_subset['Month'] = Loan_Data_subset['Month'].astype(months_catg)
Loan_Data_subset.groupby('Month')['ListingNumber'].count()
```

[52]:
```
Month
Jan     9036
Feb     7887
Mar     5376
Apr     4906
May     5469
Jun     5805
Jul     6630
```

```
Aug     6401
Sept    7480
Oct     8586
Nov     8122
Dec     8284
Name: ListingNumber, dtype: int64
```

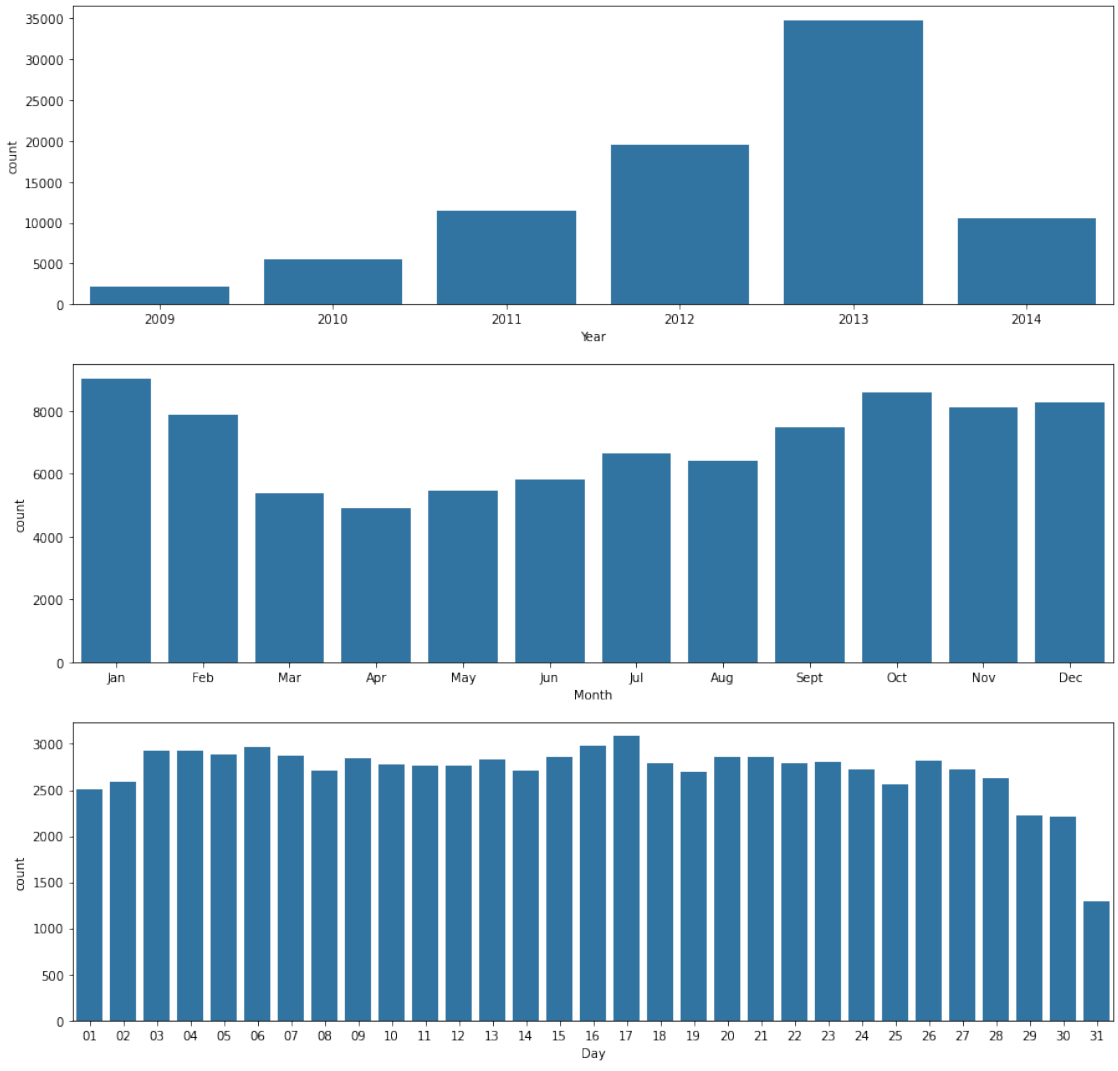
[53]:
```
# Figuring out the ListingNumber for each day
days = ['01', '02', '03', '04', '05', '06', '07', '08', '09', '10', '11', '12',␣
 ↪'13','14', '15', '16', '17', '18', '19', '20', '21', '22', '23', '24', '25',␣
 ↪'26', '27', '28', '29', '30', '31']
days_catg = pd.api.types.CategoricalDtype(ordered = True, categories = days)
Loan_Data_subset['Day'] = Loan_Data_subset['Day'].astype(days_catg)
Loan_Data_subset.groupby('Day')['ListingNumber'].count()
```

[53]:
```
Day
01    2513
02    2587
03    2919
04    2932
05    2890
06    2961
07    2876
08    2704
09    2845
10    2782
11    2757
12    2769
13    2834
14    2709
15    2856
16    2985
17    3082
18    2789
19    2694
20    2852
21    2864
22    2789
23    2809
24    2717
25    2566
26    2812
27    2724
28    2625
29    2230
30    2214
31    1296
```

```
Name: ListingNumber, dtype: int64
```

[54]:
```
#loans listed year, month and day
fig, ax = plt.subplots(nrows=3, figsize = [15,15])
color = sns.color_palette()[0]
sns.countplot(data = Loan_Data_subset, x = 'Year', color = color, ax = ax[0])
sns.countplot(data = Loan_Data_subset, x = 'Month', color = color, ax = ax[1])
sns.countplot(data = Loan_Data_subset, x = 'Day', color = color, ax = ax[2])
```

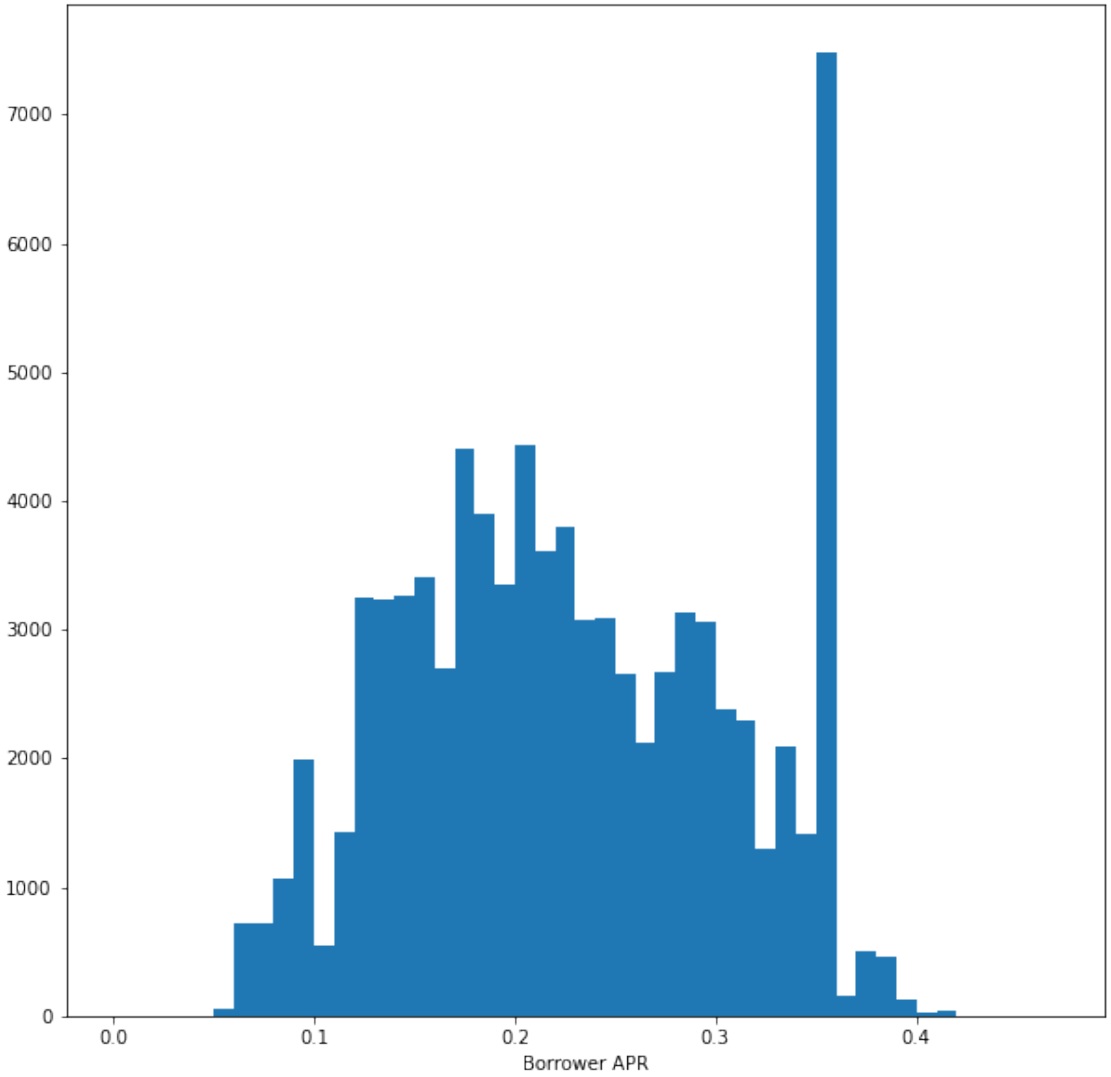
[54]:
```
<AxesSubplot:xlabel='Day', ylabel='count'>
```



[55]:
```
#the distribution of the borrower APR; since it's the main variable of interest
Bins = np.arange(0, Loan_Data_subset['BorrowerAPR'].max()+0.05, 0.01)
plt.figure(figsize=[10, 10])
```

```
plt.hist(data = Loan_Data_subset, x = 'BorrowerAPR', bins = Bins);
plt.xlabel('Borrower APR')
```

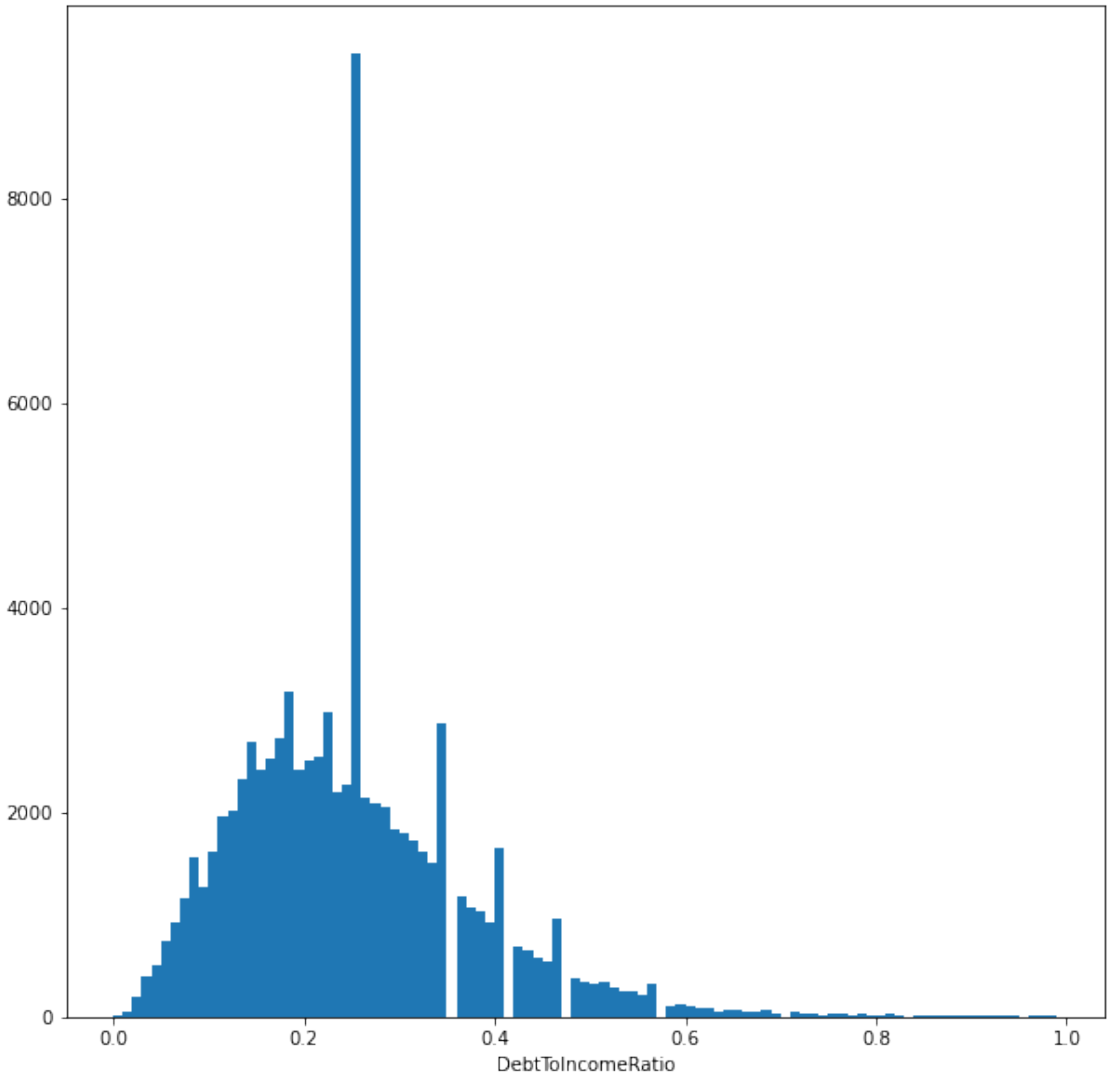
[55]: Text(0.5, 0, 'Borrower APR')



The distribution above appears to be a multimodal distribution; which means it is a distribution with more than one peak or "mode." An insignificant peak centered at 0.1 and 0.3, and a larger one at 0.2. Furthermore, it is obvious that only few loans have an APR greater than 0.4.

[56]:
```
#the distribution of the DebtToIncomeRatio
Bins = np.arange(0,1, 0.01)
plt.figure(figsize=[10, 10])
plt.hist(data = Loan_Data_subset, x = 'DebtToIncomeRatio', bins = Bins)
plt.xlabel('DebtToIncomeRatio')
```

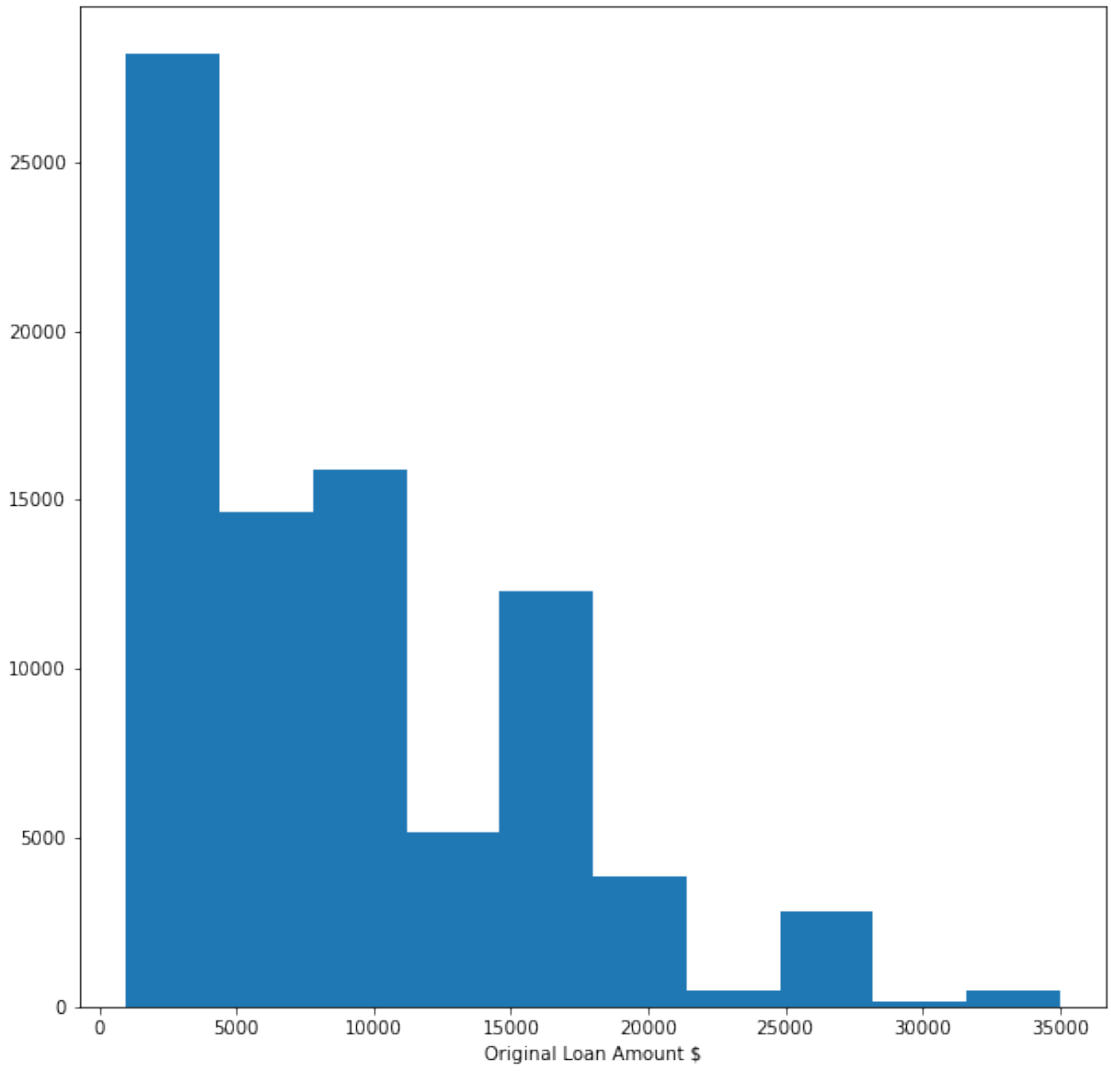
[56]: Text(0.5, 0, 'DebtToIncomeRatio')



The distribution above appears to be a unimodal distribution; which means it is a distribution with one clear peak or most frequent value. It has a peak at 0.2 and unusual one at 0.25; which implies that the majority of the people would rather a 1:4 ratio of debt to income, and this is great.

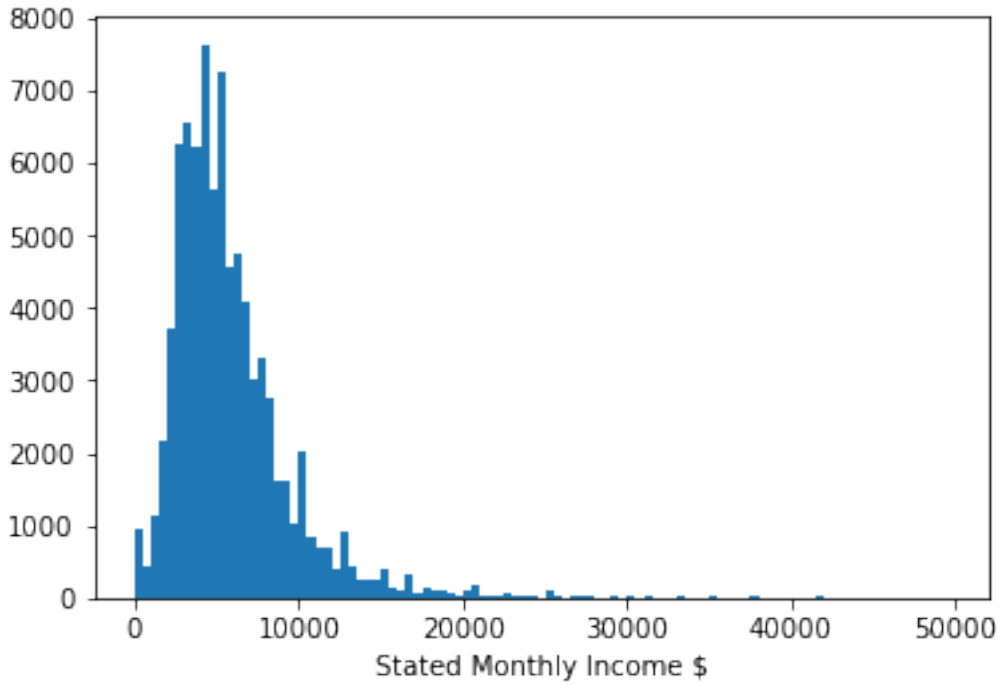
[57]:
```
# histogram of the original loan amount; which is the first predictor variable␣
 ↪of interest
plt.figure(figsize=[10, 10])
plt.hist(Loan_Data_subset.LoanOriginalAmount)
plt.xlabel('Original Loan Amount $')
```

[57]: Text(0.5, 0, 'Original Loan Amount $')



The distribution above shows that most of the loans are multiples of 5K; since the larger frequencies are at 10K, 15K, 20K, and 35K.

[58]:
```
# A distribution of the monthly income; which is another variable of interest
Bins = np.arange(0, 50000, 500)
plt.hist(data=Loan_Data_subset, x='StatedMonthlyIncome', bins=Bins);
plt.xlabel('Stated Monthly Income $');
```



The distribution above is obviously right screwed, and shows that the Stated Monthly Income is less than 30K with some outliers that better be removed around 50K-100K.

[59]:
```
# removing the outliers stated above:
Loan_Data_subset =␣
 ↪Loan_Data_subset[Loan_Data_subset['StatedMonthlyIncome']<30000]
```

[60]:
```
# Test: check the data
Loan_Data_subset.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 83719 entries, 1 to 113936
Data columns (total 21 columns):
 #   Column                      Non-Null Count  Dtype
---  ------                      --------------  -----
 0   ListingNumber               83719 non-null  int64
 1   LoanOriginalAmount          83719 non-null  int64
 2   LoanStatus                  83719 non-null  object
 3   ListingCategory (numeric)   83719 non-null  category
 4   BorrowerState               83719 non-null  object
 5   BorrowerAPR                 83719 non-null  float64
 6   BorrowerRate                83719 non-null  float64
 7   StatedMonthlyIncome         83719 non-null  float64
 8   ProsperRating (Alpha)       83719 non-null  object
 9   Occupation                  83719 non-null  object
```

```
 10  Term                       83719 non-null  int64
 11  EmploymentStatus           83719 non-null  object
 12  TotalInquiries             83719 non-null  int64
 13  DebtToIncomeRatio          83719 non-null  float64
 14  MonthlyLoanPayment         83719 non-null  float64
 15  TotalTrades                83719 non-null  int64
 16  Investors                  83719 non-null  int64
 17  Year                       83719 non-null  category
 18  Month                      83719 non-null  category
 19  Day                        83719 non-null  category
 20  Time                       83719 non-null  object
dtypes: category(4), float64(5), int64(6), object(6)
memory usage: 11.8+ MB
```

[61]:
```
Loan_Data_subset['ProsperRating (Alpha)'].unique()
```

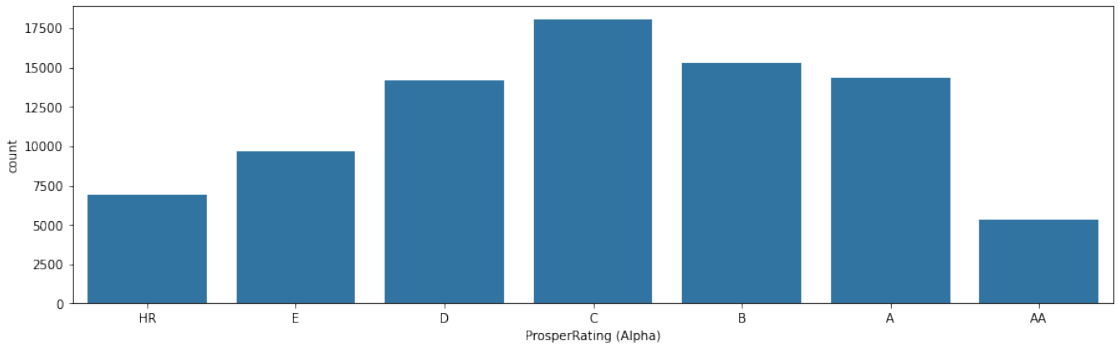
[61]:
```
array(['A', 'D', 'B', 'E', 'C', 'AA', 'HR'], dtype=object)
```

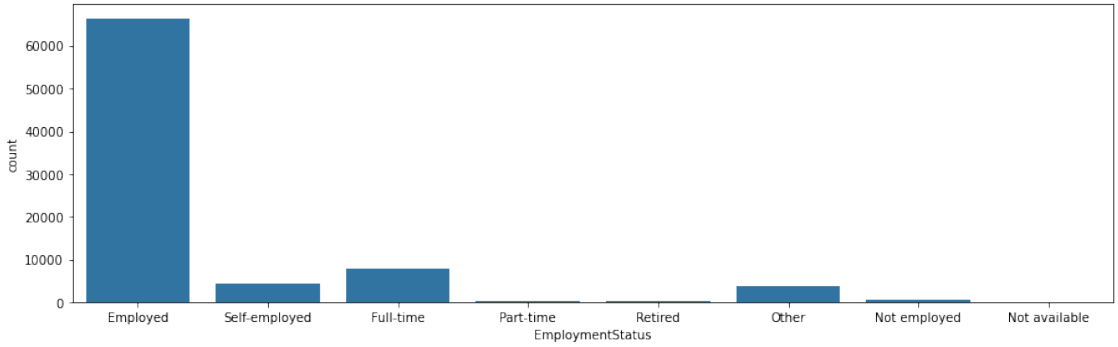
The distributions of ProsperRating (Alpha), Occupation and EmploymentStatus:

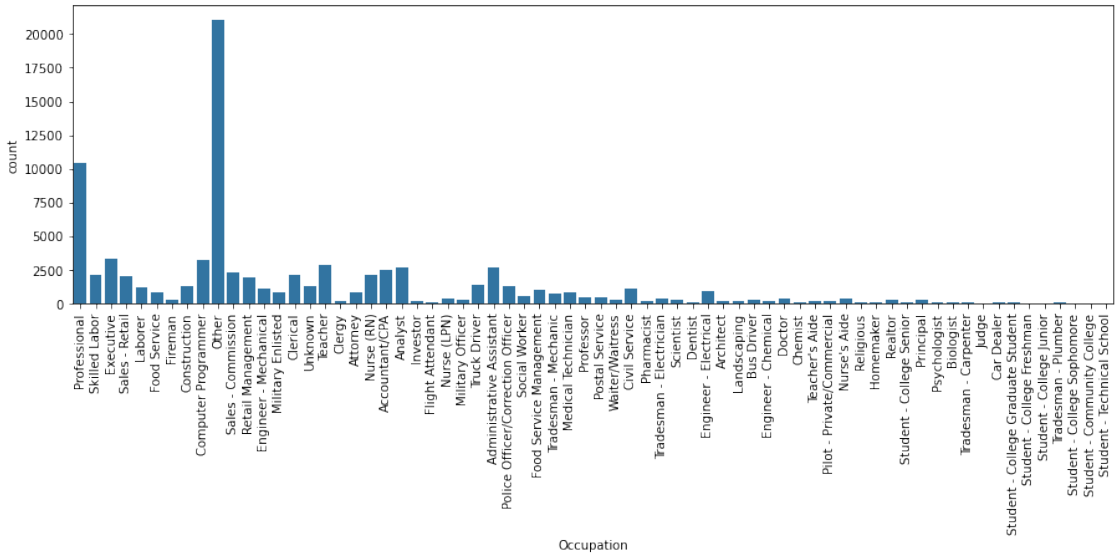
[62]:
```
#change the ProsperRating (Alpha) to ordered-categoricaltypes
prosper_rating= ['HR','E','D','C','B','A','AA']
prosper_rating_catg = pd.api.types.CategoricalDtype(ordered = True, categories␣
 ↪= prosper_rating)
Loan_Data_subset['ProsperRating (Alpha)'] = Loan_Data_subset['ProsperRating␣
 ↪(Alpha)'].astype(prosper_rating_catg)
```

[63]:
```
#change the EmploymentStatus to ordered-categoricaltypes
status =␣
 ↪['Employed','Self-employed','Full-time','Part-time','Retired','Other','Not␣
 ↪employed', 'Not available']
status_catg = pd.api.types.CategoricalDtype(ordered = True, categories = status)
Loan_Data_subset['EmploymentStatus'] = Loan_Data_subset['EmploymentStatus'].
 ↪astype(status_catg)
```

[64]:
```
#The distributions of ProsperRating (Alpha), Occupation and EmploymentStatus:
fig, ax= plt.subplots(nrows=3, figsize = [15,15])
color= sns.color_palette()[0]
sns.countplot(data= Loan_Data_subset, x = 'ProsperRating (Alpha)', color =␣
 ↪color, ax = ax[0])
sns.countplot(data= Loan_Data_subset, x = 'EmploymentStatus', color = color, ax␣
 ↪= ax[1]);
sns.countplot(data= Loan_Data_subset, x = 'Occupation', color = color, ax =␣
 ↪ax[2]);
plt.xticks(rotation= 90);
```







Form above, we can see that the ratings of most of the borrowers are from D to A. Also, it shows that the majority of the borrowers are employed or have a full-time job. Finally, it shows that students are the least of the borrowers.

1.1.3 Were there any unusual points or was there a need to perform any transformations?

The distribution of the APR is multimodal; but despite that, there are no unusual points nor the need of any transformations. ### Of the features investigated, were there any unusual distribu-

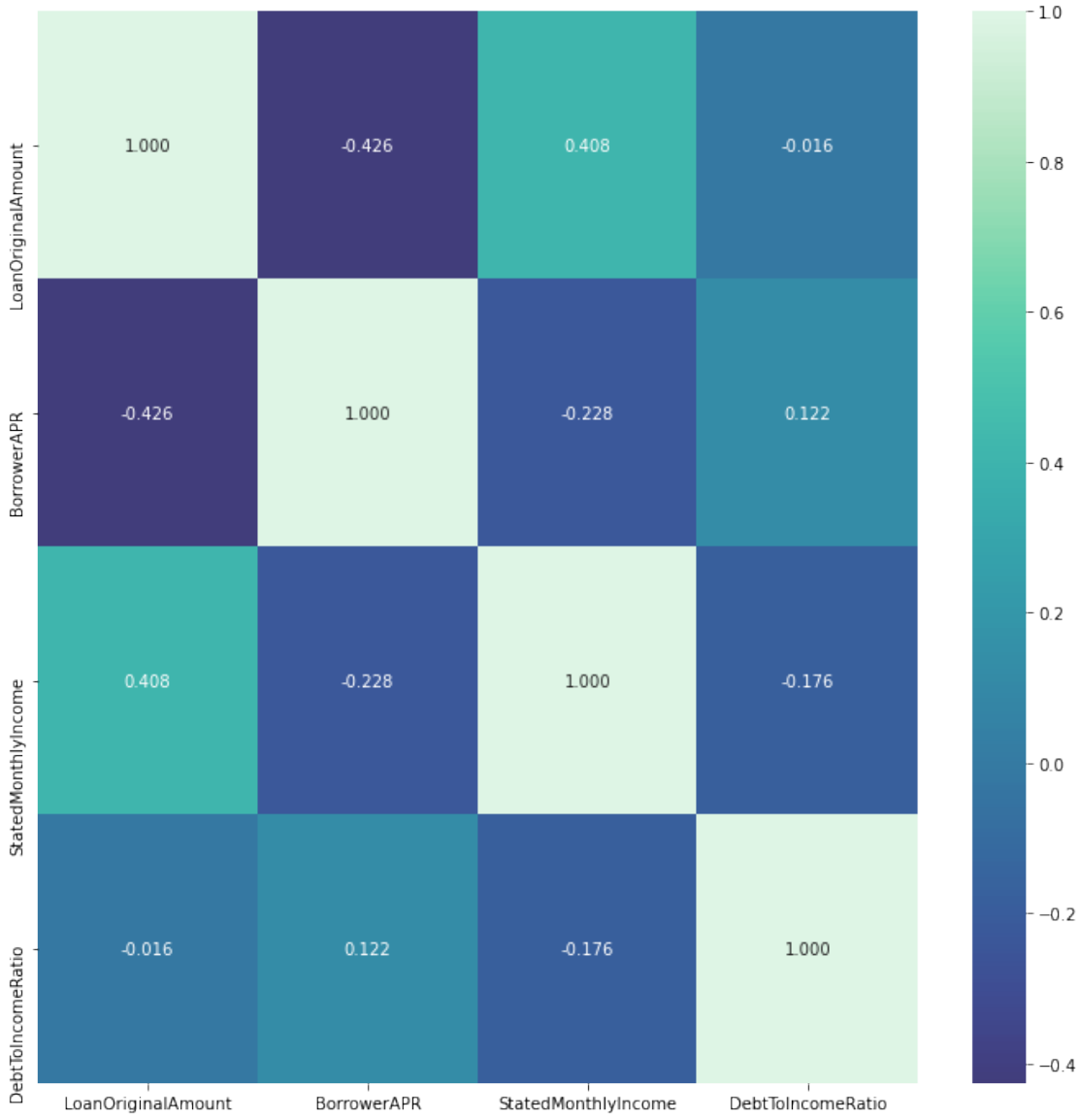
tions? The distributions of StatedMonthlyIncome and DebtToIncomeRatio are highly right skewed. ### The operations performed on the data to tidy, adjust, or change the form of it: Removing the outliers in the StatedMonthlyIncome as to not affect the outcomes of the analyses.

Bivariate Exploration

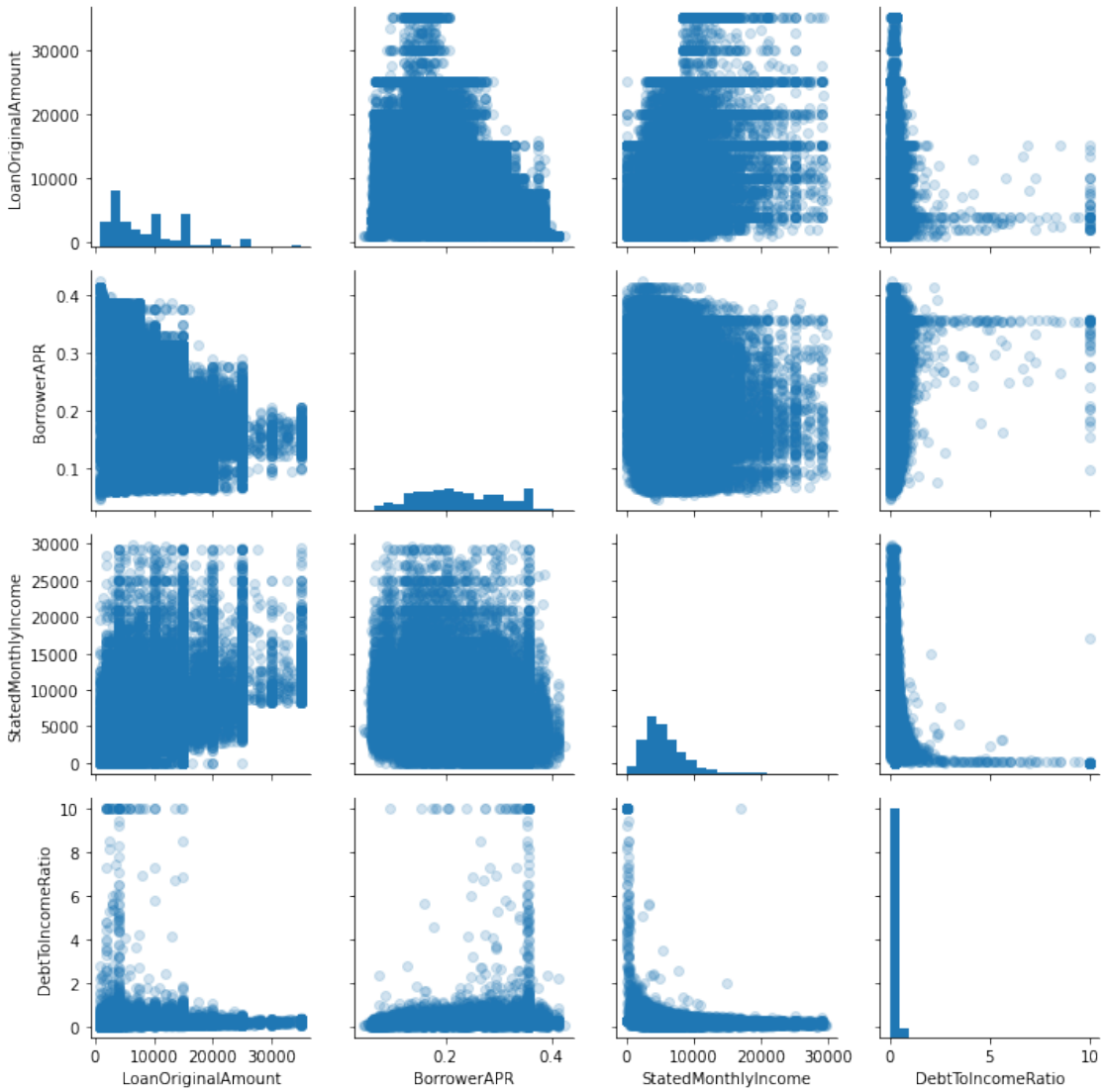
Without further ado, we are going to take a look at the correlations between the features in the dataset.

[65]:
```
#the features to be correlated
features= ['LoanOriginalAmount', 'BorrowerAPR',␣
 ↪'StatedMonthlyIncome','DebtToIncomeRatio']
```

[66]:
```
#heatmap correlation plot
plt.figure(figsize = [12, 12])
sns.heatmap(Loan_Data_subset[features].corr(), annot = True, fmt = '.3f', cmap␣
 ↪= 'mako', center = 0);
```



[67]:
```
#Subplot grid for plotting pairwise relationships in the dataset
corr = sns.PairGrid(data = Loan_Data_subset, vars = features)
corr = corr.map_diag(plt.hist, bins=20)
corr.map_offdiag(plt.scatter, alpha=0.2);
```



The correlation and the scatter plot above proves the hypothesis to be true, that is the more the loan amount to be, the less the borrower APR; the correlation coefficient of the Loan Original Amount and the Borrower APR is -0.426, and the scatter plot shows a negatively correlated relationship between these two features, too. However, the Loan Original Amoun is positively correlated with the Stated Monthly Income with a correlation coefficient of 0.408; and that actually makes sense since borrowers could loan more money if they have more monthly income.

Next, we will look at how the categorical variables correlate with the Stated Monthly Income, Borrower APR, and Loan Original Amount:

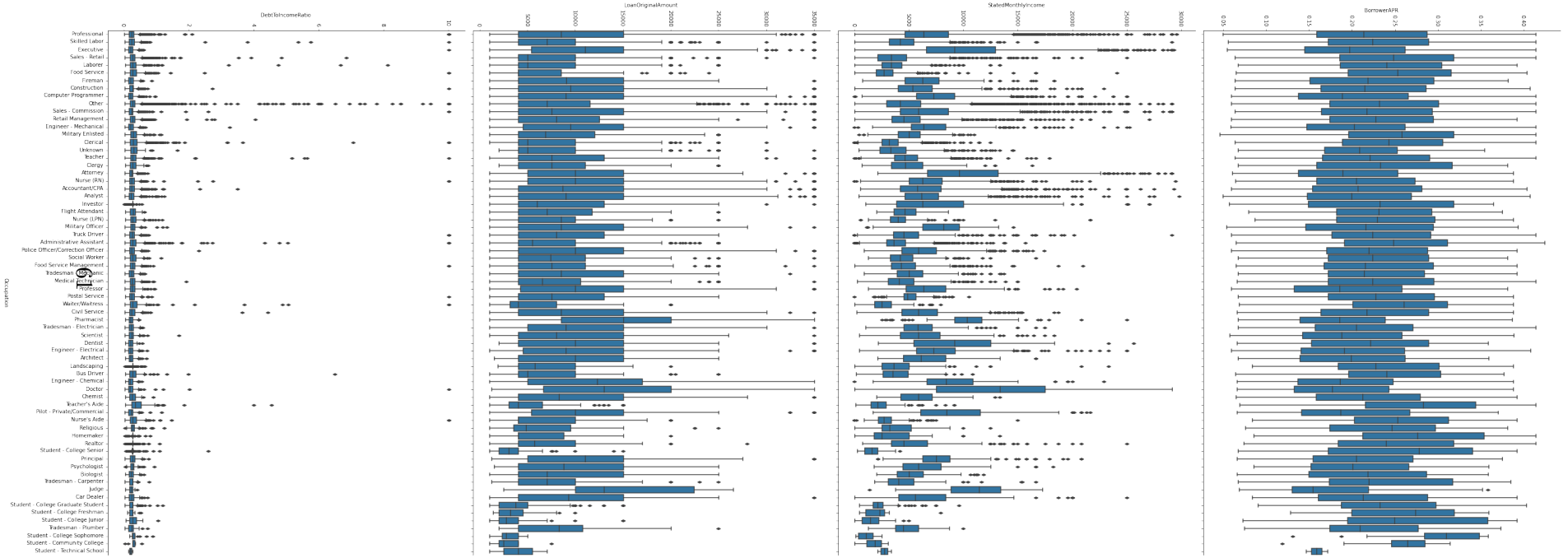
[68]:
```
#method to help creating box plots with seaborn PairGrid
def Box_PairGrid(x,y,**kwargs):
    color = sns.color_palette()[0]
    sns.boxplot(x,y,color=color)
```

[69]:
```
#the values of the x and y axis
y_axis1 = ['BorrowerAPR', 'StatedMonthlyIncome',␣
↪'LoanOriginalAmount','DebtToIncomeRatio']
x_axis1 = ['Occupation']
```

[70]:
```
#plotting the relationships
plt.figure(figsize = [15, 15])
corr = sns.PairGrid(data = Loan_Data_subset, y_vars=y_axis1, x_vars=x_axis1,␣
↪height = 10, aspect = 1.5)
corr.map(Box_PairGrid);
plt.xticks(rotation=90);
```

```
/Users/yara/opt/anaconda3/envs/last_project/lib/python3.8/site-
packages/seaborn/_decorators.py:36: FutureWarning: Pass the following variables
as keyword args: x, y. From version 0.12, the only valid positional argument
will be `data`, and passing other arguments without an explicit keyword will
result in an error or misinterpretation.
  warnings.warn(
/Users/yara/opt/anaconda3/envs/last_project/lib/python3.8/site-
packages/seaborn/_decorators.py:36: FutureWarning: Pass the following variables
as keyword args: x, y. From version 0.12, the only valid positional argument
will be `data`, and passing other arguments without an explicit keyword will
result in an error or misinterpretation.
  warnings.warn(
/Users/yara/opt/anaconda3/envs/last_project/lib/python3.8/site-
packages/seaborn/_decorators.py:36: FutureWarning: Pass the following variables
as keyword args: x, y. From version 0.12, the only valid positional argument
will be `data`, and passing other arguments without an explicit keyword will
result in an error or misinterpretation.
  warnings.warn(
/Users/yara/opt/anaconda3/envs/last_project/lib/python3.8/site-
packages/seaborn/_decorators.py:36: FutureWarning: Pass the following variables
as keyword args: x, y. From version 0.12, the only valid positional argument
will be `data`, and passing other arguments without an explicit keyword will
result in an error or misinterpretation.
  warnings.warn(
```

```
<Figure size 1080x1080 with 0 Axes>
```


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

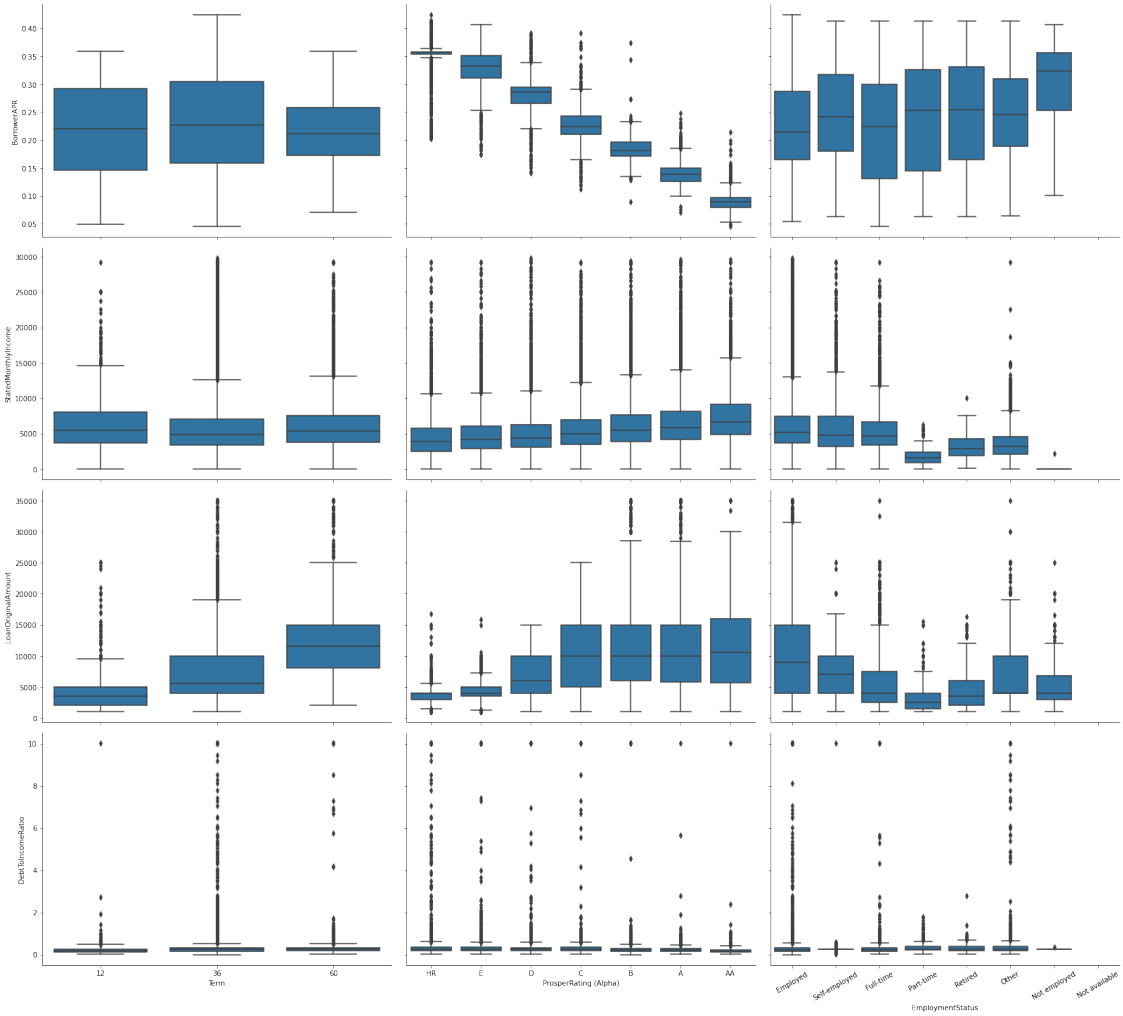
[71]:
```
#the values of the x
x_axis2= ['Term','ProsperRating (Alpha)', 'EmploymentStatus']
```

[72]:
```
#plot matrix of categorical features vs. numeric features
plt.figure(figsize = [15, 15])
corr = sns.PairGrid(data= Loan_Data_subset, y_vars=y_axis1 , x_vars= x_axis2 ,␣
 ↪height=5, aspect = 1.5)
corr.map(Box_PairGrid);
plt.xticks(rotation=30);
```

```
/Users/yara/opt/anaconda3/envs/last_project/lib/python3.8/site-
packages/seaborn/_decorators.py:36: FutureWarning: Pass the following variables
as keyword args: x, y. From version 0.12, the only valid positional argument
will be `data`, and passing other arguments without an explicit keyword will
result in an error or misinterpretation.
  warnings.warn(
/Users/yara/opt/anaconda3/envs/last_project/lib/python3.8/site-
packages/seaborn/_decorators.py:36: FutureWarning: Pass the following variables
as keyword args: x, y. From version 0.12, the only valid positional argument
will be `data`, and passing other arguments without an explicit keyword will
result in an error or misinterpretation.
  warnings.warn(
/Users/yara/opt/anaconda3/envs/last_project/lib/python3.8/site-
packages/seaborn/_decorators.py:36: FutureWarning: Pass the following variables
as keyword args: x, y. From version 0.12, the only valid positional argument
will be `data`, and passing other arguments without an explicit keyword will
result in an error or misinterpretation.
  warnings.warn(
/Users/yara/opt/anaconda3/envs/last_project/lib/python3.8/site-
packages/seaborn/_decorators.py:36: FutureWarning: Pass the following variables
as keyword args: x, y. From version 0.12, the only valid positional argument
will be `data`, and passing other arguments without an explicit keyword will
result in an error or misinterpretation.
  warnings.warn(
/Users/yara/opt/anaconda3/envs/last_project/lib/python3.8/site-
packages/seaborn/_decorators.py:36: FutureWarning: Pass the following variables
as keyword args: x, y. From version 0.12, the only valid positional argument
will be `data`, and passing other arguments without an explicit keyword will
result in an error or misinterpretation.
  warnings.warn(
/Users/yara/opt/anaconda3/envs/last_project/lib/python3.8/site-
packages/seaborn/_decorators.py:36: FutureWarning: Pass the following variables
as keyword args: x, y. From version 0.12, the only valid positional argument
will be `data`, and passing other arguments without an explicit keyword will
result in an error or misinterpretation.
```

```
  warnings.warn(
/Users/yara/opt/anaconda3/envs/last_project/lib/python3.8/site-
packages/seaborn/_decorators.py:36: FutureWarning: Pass the following variables
as keyword args: x, y. From version 0.12, the only valid positional argument
will be `data`, and passing other arguments without an explicit keyword will
result in an error or misinterpretation.
  warnings.warn(
/Users/yara/opt/anaconda3/envs/last_project/lib/python3.8/site-
packages/seaborn/_decorators.py:36: FutureWarning: Pass the following variables
as keyword args: x, y. From version 0.12, the only valid positional argument
will be `data`, and passing other arguments without an explicit keyword will
result in an error or misinterpretation.
  warnings.warn(
/Users/yara/opt/anaconda3/envs/last_project/lib/python3.8/site-
packages/seaborn/_decorators.py:36: FutureWarning: Pass the following variables
as keyword args: x, y. From version 0.12, the only valid positional argument
will be `data`, and passing other arguments without an explicit keyword will
result in an error or misinterpretation.
  warnings.warn(
/Users/yara/opt/anaconda3/envs/last_project/lib/python3.8/site-
packages/seaborn/_decorators.py:36: FutureWarning: Pass the following variables
as keyword args: x, y. From version 0.12, the only valid positional argument
will be `data`, and passing other arguments without an explicit keyword will
result in an error or misinterpretation.
  warnings.warn(
/Users/yara/opt/anaconda3/envs/last_project/lib/python3.8/site-
packages/seaborn/_decorators.py:36: FutureWarning: Pass the following variables
as keyword args: x, y. From version 0.12, the only valid positional argument
will be `data`, and passing other arguments without an explicit keyword will
result in an error or misinterpretation.
  warnings.warn(
/Users/yara/opt/anaconda3/envs/last_project/lib/python3.8/site-
packages/seaborn/_decorators.py:36: FutureWarning: Pass the following variables
as keyword args: x, y. From version 0.12, the only valid positional argument
will be `data`, and passing other arguments without an explicit keyword will
result in an error or misinterpretation.
  warnings.warn(
```

```
<Figure size 1080x1080 with 0 Axes>
```

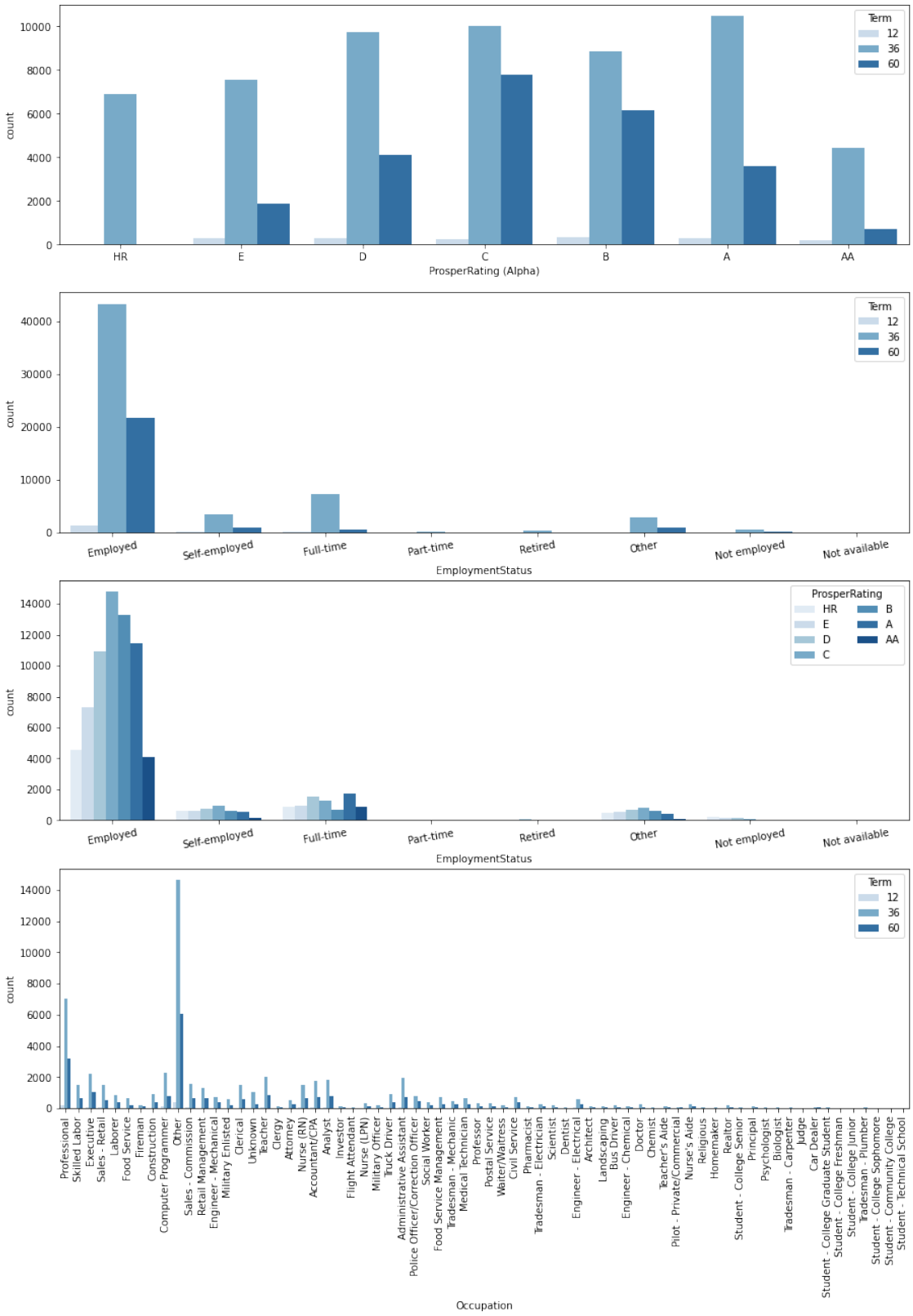


Form the figures above, it is obvious that the LoanOriginalAmount is positively correlated with the loan Term; i.e it increases with the increase of the Term. On the other hand, the borrower APR is negatively correlated with the Prosper Rating; i.e it decreases with the increase of the rating. Hence, the borrowers with the worst Prosper ratings have the highest APR, which means that the ProsperRating strongly affects the borrower APR. Also, ProsperRating is positively correlated with both StatedMonthlyIncome and LoanOriginalAmount; i.e borrowers with higher rating have better monthly income and loan amount. Finally, borrowers who happens to be either employed, self-employed or full-time have more monthly income and loan amont than the rest.

Now, we will take a look at the four categorical features (which are the ProsperRating, Term, EmploymentStatus and Occupation) and the relationships between them:

[73]:
```
plt.figure(figsize = [15, 20])
#Subplot 1:ProsperRating vs. Term
plt.subplot(4, 1, 1)
sns.countplot(data= Loan_Data_subset, x= 'ProsperRating (Alpha)', hue= 'Term',␣
 ↪palette= 'Blues')
```

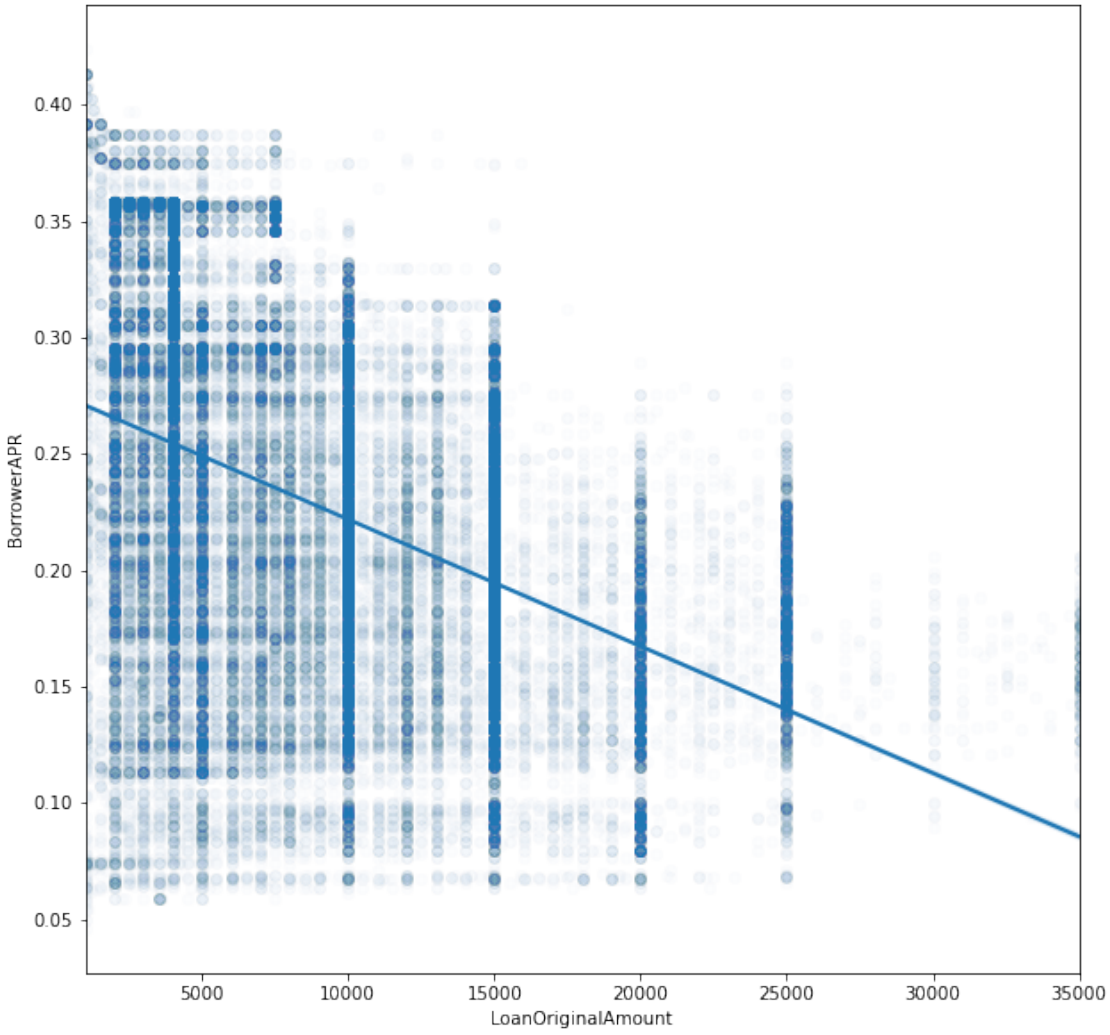
```
#Subplot 2:EmploymentStatus vs. Term
ax = plt.subplot(4, 1, 2)
sns.countplot(data = Loan_Data_subset, x = 'EmploymentStatus', hue = 'Term',␣
↪palette = 'Blues')
ax.legend(loc = 'upper right', ncol = 1, title='Term');
plt.xticks(rotation = 10);
#Subplot 3:ProsperRating vs. EmploymentStatus
ax = plt.subplot(4, 1, 3)
sns.countplot(data = Loan_Data_subset, x = 'EmploymentStatus', hue =␣
↪'ProsperRating (Alpha)', palette = 'Blues')
ax.legend(loc = 1, ncol = 2, title='ProsperRating');
plt.xticks(rotation = 10);
#Subplot 4:Occupation vs. Term
ax = plt.subplot(4, 1, 4)
sns.countplot(data = Loan_Data_subset, x = 'Occupation', hue = 'Term', palette␣
↪= 'Blues');
ax.legend(loc = 'upper right', ncol = 1, title='Term');
plt.xticks(rotation=90);
```


Term
12
36
60
count
ProsperRating (Alpha)
HR
E
D
C
B
A
AA
EmploymentStatus
Employed
Self-employed
Full-time
Part-time
Retired
Other
Not employed
Not available
ProsperRating
HR
E
D
C
B
A
AA
Occupation
Professional
Skilled Labor
Executive
Sales - Retail
Laborer
Food Service
Fireman
Construction
Computer Programmer
Other
Sales - Commission
Retail Management
Engineer - Mechanical
Military Enlisted
Clerical
Unknown
Teacher
Clergy
Attorney
Nurse (RN)
Accountant/CPA
Analyst
Investor
Flight Attendant
Nurse (LPN)
Military Officer
Truck Driver
Administrative Assistant
Police Officer/Correction Officer
Social Worker
Food Service Management
Tradesman - Mechanic
Medical Technician
Professor
Postal Service
Waiter/Waitress
Civil Service
Pharmacist
Tradesman - Electrician
Scientist
Dentist
Engineer - Electrical
Architect
Landscaping
Bus Driver
Engineer - Chemical
Doctor
Chemist
Teacher's Aide
Pilot - Private/Commercial
Nurse's Aide
Religious
Homemaker
Realtor
Student - College Senior
Principal
Psychologist
Biologist
Tradesman - Carpenter
Judge
Car Dealer
Student - College Graduate Student
Student - College Freshman
Student - College Junior
Tradesman - Plumber
Student - College Sophomore
Student - Community College
Student - Technical School

The figures above shows that despite the lack of data in EmploymentStatus when it came to Part-time, Retired and Not employed borrowers in order to have a look at the interactions between it and the Term and LoanOriginalAmount features, we still could see that there is an interaction between the Term and the ProsperRating features. As it shown, there is only 36-months loans for HR ProsperRating borrowers, and more 60-months loans for B and C Ratings borrowers; and apparently, the 36-months is the most preferable duration.

Finally, we will have a better look at how the BorrowerAPR and the LoanOriginalAmount are related to each other for the entire data:

[74]:
```
# a regplot to represent the relationship between  borrowers APR and the␣
↪LoanOriginalAmount
plt.figure(figsize = [10, 10])
sns.regplot(data = Loan_Data_subset, x='LoanOriginalAmount', y='BorrowerAPR',␣
↪scatter_kws={'alpha':0.01});
```



The plot above shows that the BorrowerAPR is negatively correlated with the LoanOriginalAmount, since the BorrowerAPR has a large range that decrease with the increse of the LoanOriginalAmount.

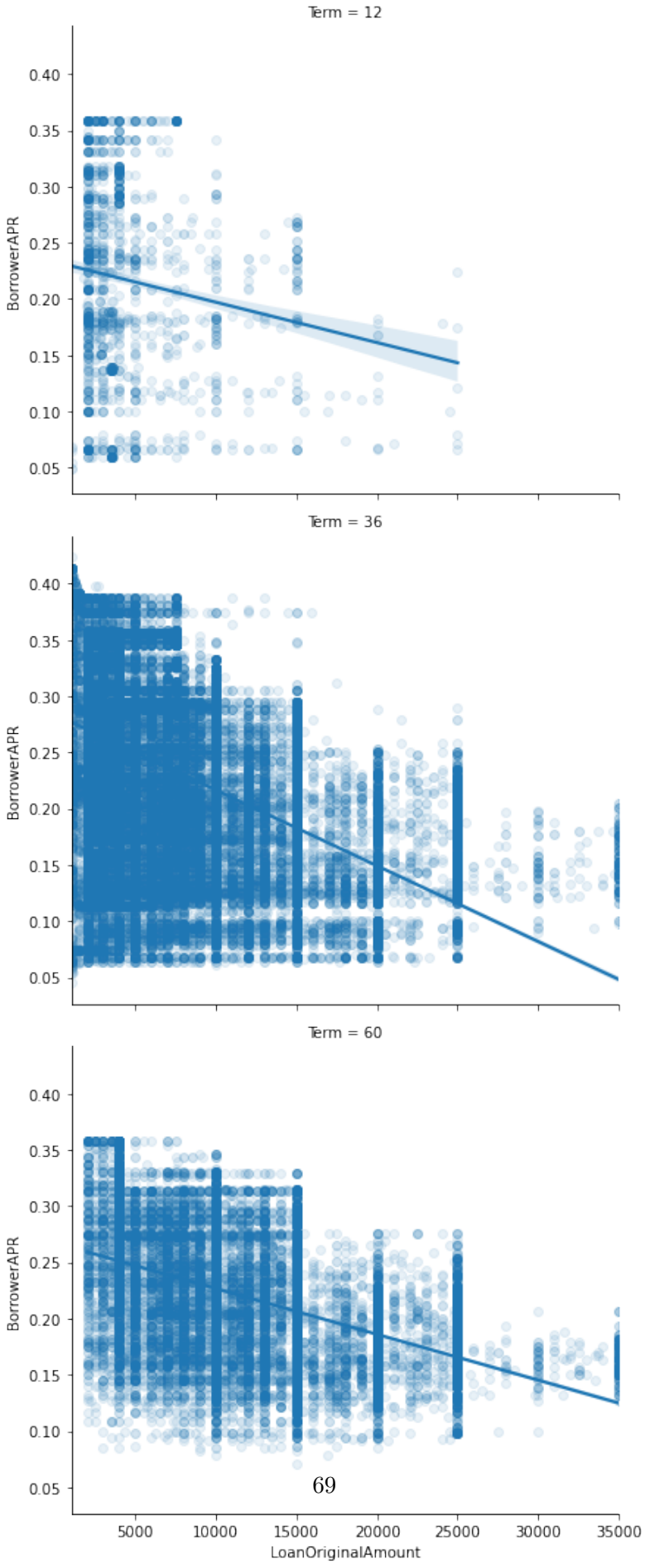
The way the features of interest varied with other features in the dataset: The BorrowerAPR is negatively correlated with the LoanOriginalAmount, which means the less the loan amount the higher the BorrowerAPR was. Also, at different amounts of the loan, the BorrowerAPR has a large range, but it is negatively correlated. Moreover, the ProsperRating has a strong negative impact on the BorrowerAPR, where it decreases with the better ProsperRating.

1.2 Observation of any interesting relationships between the other features that are not the main features of interest:

After the intense exploration conducted, it has been noticed that the LoanOriginalAmount is positively correlated with the StatedMonthlyIncome, and that made more snese since borrowers with higher monthly income are able to loan more money. It also has been noticed that borrowers with better ProsperRating have higher monthly income and greater loan amount. Finally, the interaction between the ProsperRating and the Term is obvious, as in there are more 60-months loans in B and C ProsperRatings, and only 36-months for HR.

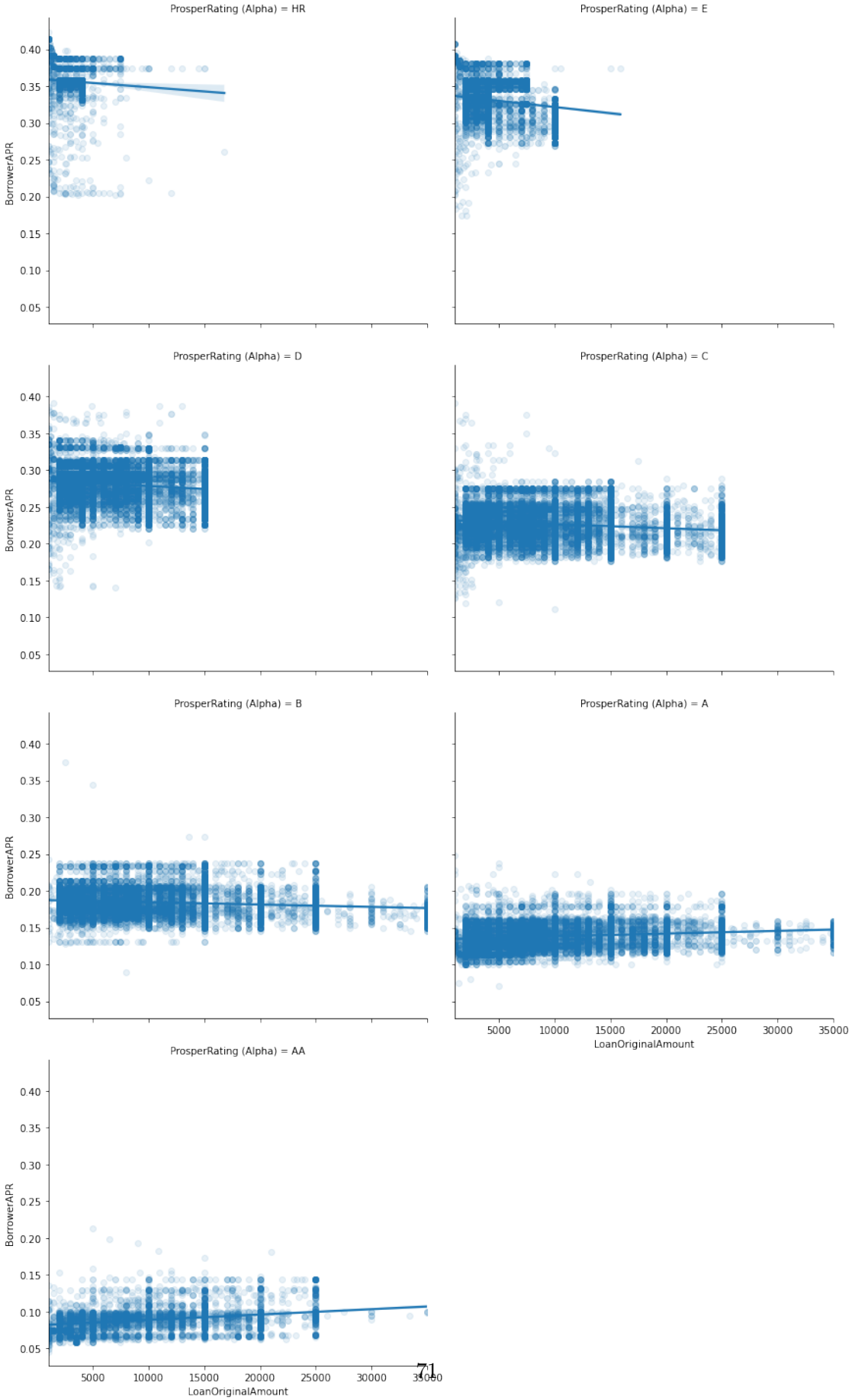
Multivariate Exploration In this final part, we will focus on exploring the categorical variables ProsperRating and Term, and how they affect the relationship between BorrowerAPR and LoanOriginalAmount:

[98]:
```
# The effect of the Term variable on both the BorrowerAPR and the␣
↪LoanOriginalAmount
corr=sns.FacetGrid(data= Loan_Data_subset, aspect=1.2, height=5, col='Term',␣
↪col_wrap=1)
corr.map(sns.regplot,'LoanOriginalAmount','BorrowerAPR', x_jitter=0.04,␣
↪scatter_kws={'alpha':0.1});
corr.add_legend();
```


Term = 12
Term = 36
Term = 60
BorrowerAPR
0.40
0.35
0.30
0.25
0.20
0.15
0.10
0.05
5000
10000
15000
20000
25000
30000
35000
LoanOriginalAmount

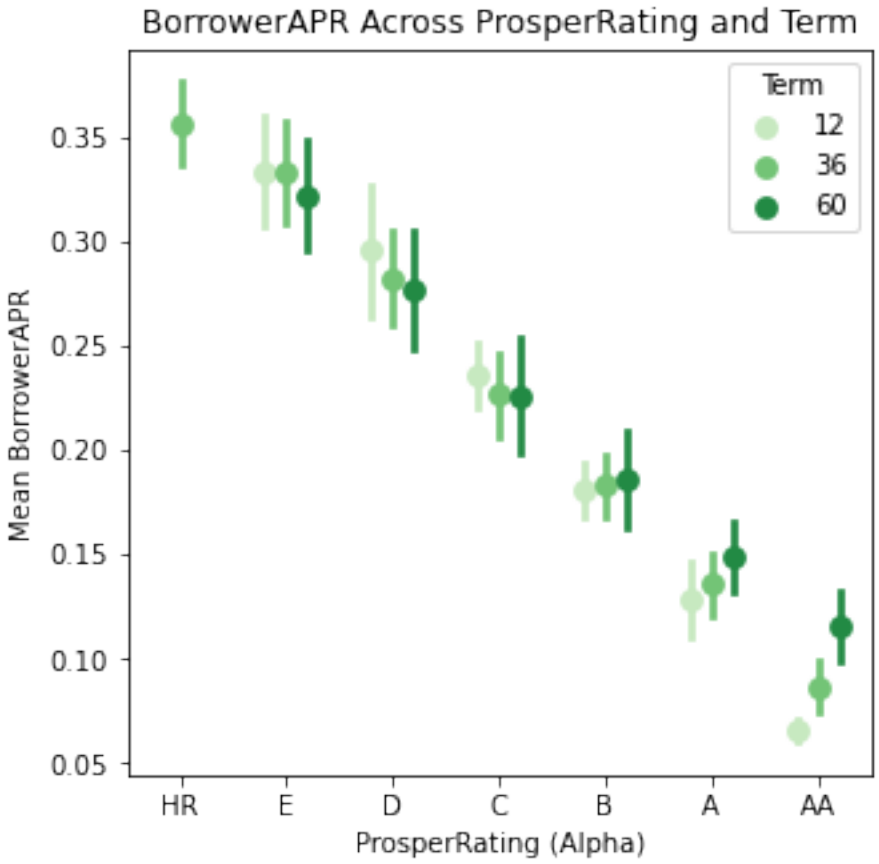
Form what we can see above, it seems that the Term does not affect the relationship between the BorrowerAPR and LoanOriginalAmount.

[84]:
```
# The effect of the ProsperRating variable on both the BorrowerAPR and the␣
 ↪LoanOriginalAmount
corr=sns.FacetGrid(data=Loan_Data_subset, aspect=1.2, height=5,␣
 ↪col='ProsperRating (Alpha)', col_wrap=2)
corr.map(sns.regplot, 'LoanOriginalAmount', 'BorrowerAPR', x_jitter=0.04,␣
 ↪scatter_kws={'alpha':0.1});
corr.add_legend();
```


ProsperRating (Alpha) = HR
ProsperRating (Alpha) = E
ProsperRating (Alpha) = D
ProsperRating (Alpha) = C
ProsperRating (Alpha) = B
ProsperRating (Alpha) = A
ProsperRating (Alpha) = AA
BorrowerAPR
LoanOriginalAmount
0.40
0.35
0.30
0.25
0.20
0.15
0.10
0.05
5000
10000
15000
20000
25000
30000
35000

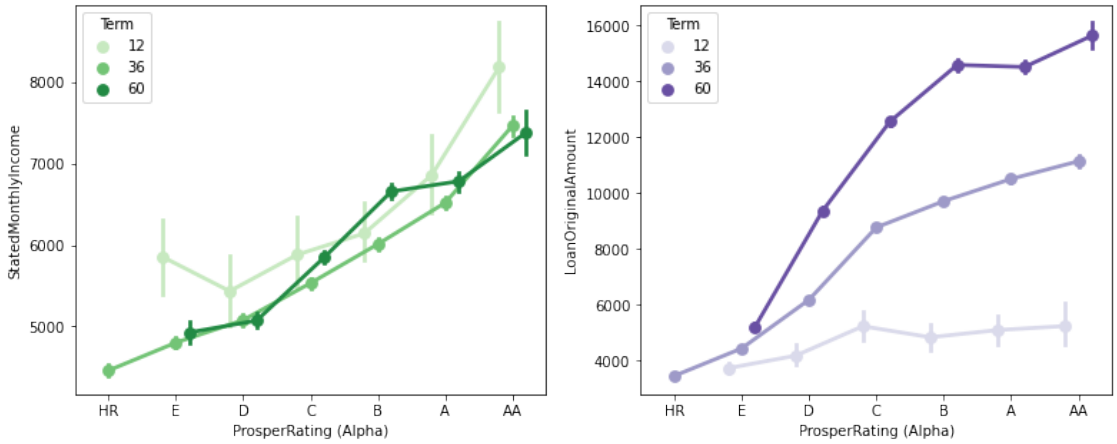
Form the figures above, we can see that the LoanOriginalAmount increases and the BorrowerAPR decreases with higher ProsperRating. Curiously, the relationship shifted slightly from negative to positive between the BorrowerAPR and LoanOriginalAmount when the ProsperRating joined the equation and increased from HR to A or better. This might happen due to the fact that people with ratings of A or AA tend borrow more money, and by increasting the BorrowerAPR, it would be possible to prevent them from borrowing more and maximize the profit. Yet, people with low ProsperRating tend to borrow less money, hence, decreasing the BorrowerAPR would only encourage them to borrow more.

[109]:
```
# The effect of the BorrowerAPR variable on both the ProsperRating and the Term
fig = plt.figure(figsize = [5,5])
ax = sns.pointplot(data = Loan_Data_subset, x = 'ProsperRating (Alpha)', y =␣
↪'BorrowerAPR', hue = 'Term',palette = 'Greens', linestyles = '', dodge = 0.
↪4, ci='sd')
plt.title('BorrowerAPR Across ProsperRating and Term')
plt.ylabel('Mean BorrowerAPR')
ax.set_yticklabels([],minor = True);
```



As the figure above shows, the BorrowerAPR decreases with the increase of Term for people with rainges of HR to C. As for the people with rainges of B to AA, the BorrowerAPR increases with the increase of Term.

[107]:
```
# the effects of the ProsperRating and the Term on StatedMonthlyIncome and␣
 ↪LoanOriginalAmount
fig, ax = plt.subplots(ncols=2, figsize=[13,5])
sns.pointplot(data = Loan_Data_subset, x = 'ProsperRating (Alpha)', y =␣
 ↪'StatedMonthlyIncome', hue = 'Term',palette = 'Greens', dodge = 0.4,␣
 ↪ax=ax[0])
sns.pointplot(data = Loan_Data_subset, x = 'ProsperRating (Alpha)', y =␣
 ↪'LoanOriginalAmount', hue = 'Term',palette = 'Purples', dodge = 0.4,␣
 ↪ax=ax[1]);
```



From the figures above, it does not look like there is an interaction with the StatedMonthlyIncome that affects the relationship between the Term and the ProsperRating; and the Term's pattern among different ratings is the same. However, there is an interaction with the LoanOriginalAmount that affects the ProsperRating and the Term. It is obvious that with a better ProsperRating, the LoanOriginalAmount of all the Terms increases, and the increase amplitude of LoanOriginalAmount between the Terms increases, as well.

The features that strengthened each other in terms of looking at the features of interest: After the extended investigation and exploration of the BorrowerAPR against the LoanOriginalAmount via looking at the impact of the ProsperRating on the relationship, the relationship shifted slightly from negative to positive between the BorrowerAPR and the LoanOriginalAmount when the the ProsperRating increased from HR to AA. Also, the ProsperRating and the Term affect the LoanOriginalAmount in such a way that with the better ProsperRating, the LoanOriginalAmount would increase for all the Terms, and the increase amplitude of LoanOriginalAmount between the Terms would increase, as well.

1.3 The interesting or surprising interactions between features:

Interestingly, the BorrowerAPR and the LoanOriginalAmount are negatively correlated when the ProsperRating is from HR to B. However, the relationship shifted positively when the ProsperRating is from A and AA. Furthermore, the BorrowerAPR decreases with the increase of the Term when the ProsperRating is from HR to C, and increases with the increase of the Term when the ProsperRating is from B to AA.

[110]:
```
#Downloading the cleaned dataset
Loan_Data_subset.to_csv('prosperLoanData_cleaned.csv')
```

Resources: * https://towardsdatascience.com/how-to-show-all-columns-rows-of-a-pandas-dataframe-c49d4507fcf * https://www.geeksforgeeks.org/python-pandas-categoricaldtype/ * https://pandas.pydata.org/pandas-docs/version/0.23.4/generated/pandas.api.types.CategoricalDtype.html * https://stackoverflow.com/questions/58010602/how-to-type-pd-api-types-categoricaldtype * https://pandas.pydata.org/pandas-docs/stable/reference/api/pandas.DataFrame.drop_duplicates.html * https://seaborn.pydata.org/generated/seaborn.PairGrid.html * https://seaborn.pydata.org/generated/seaborn.heatmap.html * https://seaborn.pydata.org/tutorial/color_palette * https://python-graph-gallery.com/92-control-color-in-seaborn-heatmaps/ * https://www.datacamp.com/community/tutorials/histograms-matplotlib * https://www.bbc.co.uk/bitesize/guides/zspfcwx/revision/1 * https://seaborn.pydata.org/generated/seaborn.Pair * https://heartbeat.fritz.ai/seaborn-heatmaps-13-ways-to-customize-correlation-matrix-visualizations-f1c49c816f07 * https://seaborn.pydata.org/generated/seaborn.PairGrid.html * https://matplotlib.org/3.1.0/gallery/subplots_axes_and_figures/subplots_demo.html * https://jakevdp.github.io/PythonDataScienceHandbook/04.06-customizing-legends.html * https://stackoverflow.com/questions/6541123/improve-subplot-size-spacing-with-many-subplots-in-matplotlib * https://seaborn.pydata.org/generated/seaborn.regplot.html * https://seaborn.pydata.org/generated/seaborn.pointplot.html