Factors That Affects The Success of A Loan Application

Investigation Overview

In this investigation which consists of a Loan Data from prosper company. i looked at finding some of the reasons why people apply for Loan and the factors that makes l their oan application Successful; with Major focus on the `Borrowers APR` and `Ratings`

Dataset Overview

The dataset contains 113,937 loans with 81 features which includes but not limited to the following; LoanOriginalAmount, BorrowerAPR, StatedMonthlyIncome, Term, ProsperRating (Alpha), EmploymentStatus and many others.it also has more than 50% of the columns containing numerical data. 871 data points were removed due to inconssitency and non relevance to the main focus of my investigations.The dataset can be found here, with feature documentation available here

In [2]:
```
# Here are the pakages that will be helpful in this data exploration
import numpy as np
import pandas as pd
import requests
import matplotlib.pyplot as plt
import seaborn as sns

%matplotlib inline
```

Steps taken to cleanup my Dataframe

In [3]:
```
#Loading the data and setting it up for Explorations
LoanData = pd.read_csv('prosperLoanData .csv')
# selecting the columns i need for my Analysis
cols = ['ListingCreationDate', 'ListingCategory (numeric)', 'BorrowerState','BorrowerF
          'ProsperRating (Alpha)','Term','DebtToIncomeRatio','LoanOriginalAmount
Loan = LoanData[cols]
```

In [4]:
```
#Made  a copy of the original data
Loan = Loan.copy()
# Dropping duplicates
Loan = Loan.drop_duplicates()
#Test
Loan.duplicated().sum()
```

Out[4]:
```
0
```

In [5]:
```
# Rename the columns
```

```
Loan = Loan.rename(columns = {'ListingCategory (numeric)': 'ListingCategory', 'Prosper
#Change datatypes
Loan.ListingCreationDate = pd.to_datetime(Loan.ListingCreationDate)
#Test
Loan['ListingCreationDate'].dtype

# Store all the numeric categories and their titles in a dictionary
category_titles = {0 : 'Not Available', 1 : 'Debt Consolidation', 2 : 'Home Improvemer
                   4 : 'Personal Loan', 5 : 'Student Use', 6 : 'Auto', 7 : 'Other', 8
                   9 : 'Boat', 10 : 'Cosmetic Procedure', 11 : 'Engagement Ring', 12 :
                  13 : 'Household Expenses', 14 : 'Large Purchases', 15 : 'Medical or
                  17 : 'RV', 18 : 'Taxes', 19 : 'Vacation', 20 : 'Wedding Loans'}

# Map the dictionary contents to the ListingCategory column
Loan.ListingCategory = Loan.ListingCategory.map(category_titles)

#Test
Loan.ListingCategory.unique()

Loan.IncomeRange = Loan.IncomeRange.str.replace('Not employed', '$0')

#Test
Loan.IncomeRange.unique()

# Store the correct variable orders in a dictionary
order_dict = {'ProsperRating': ['HR', 'E', 'D', 'C', 'B', 'A', 'AA'],
              'IncomeRange': ['$0', '$1-24,999', '$25,000-49,999',
                              '$50,000-74,999', '$75,000-99,999', '$100,000+'],
             'EmploymentStatus' :['Employed','Self-employed','Full-time','Part-time',
                                  'Retired','Other','Not employed', 'Not available']}

# Assign each column to the proper order
for key, value in order_dict.items():
    correct_order = pd.api.types.CategoricalDtype(categories=value, ordered=True)
    Loan[key] = Loan[key].astype(correct_order)
```

In [6]:
```
#Gathered Data from external sources to get a full names of the Borroers Sates Abbrevi
# Programmatically download the csv file
url = 'https://raw.githubusercontent.com/rashida048/Exploratory-data-Analysis-in-R/mai
response = requests.get(url)

with open('state_data.csv', 'wb') as file:
    file.write(response.content)

state_names = pd.read_csv('./state_data.csv')
```

In [7]:
```
# Merge Loan and state_names
loan_new = pd.merge(Loan, state_names, left_on='BorrowerState', right_on='State')
# Drop unwanted columns
loan_new.drop(columns = ['State','Latitude','Longitude','BorrowerState'], inplace=True

# Rename city column to BorrowerStates
loan_new= loan_new.rename(columns={'City': 'BorrowerStates'})

#lets Re-arrange the columns
cols = ['ListingCreationDate', 'ListingCategory', 'BorrowerStates','BorrowerRate','Bor
                'ProsperRating','Term','DebtToIncomeRatio','LoanOriginalAmount','LoanS

Loan_clean = loan_new[cols]
```

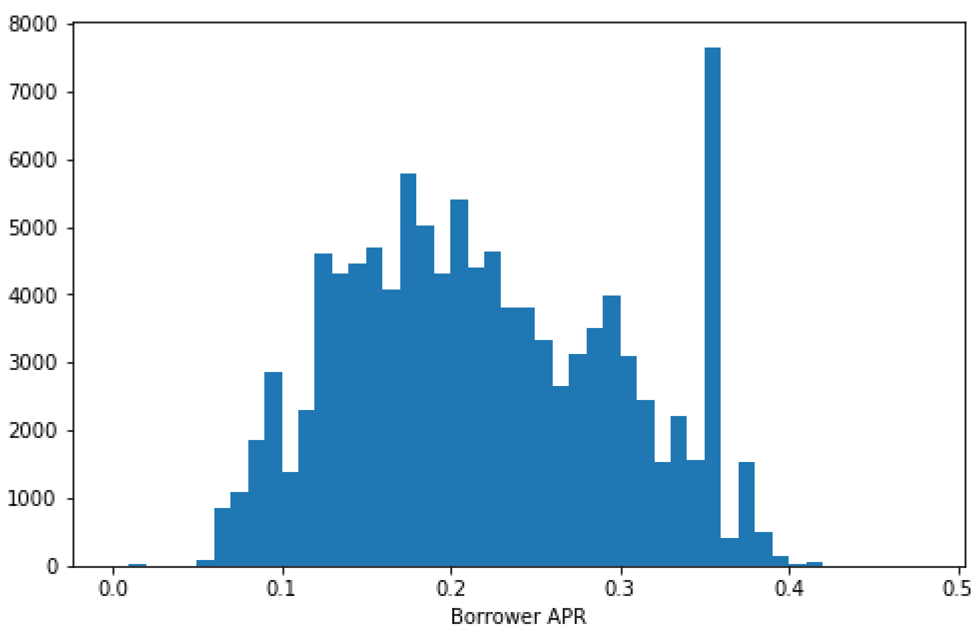
```
Loan_clean.shape
```

Out[7]:
```
(107551, 16)
```

The Dristrution of Borrower APR

> The distribution of Borrowers'APR seems multimodal. we have peaks at `0.09,0.8,0.22,0.3 and 0.35` areas, then a very sharp peac peak at `0.37 area`. The least of the APR comes after `0.43`.

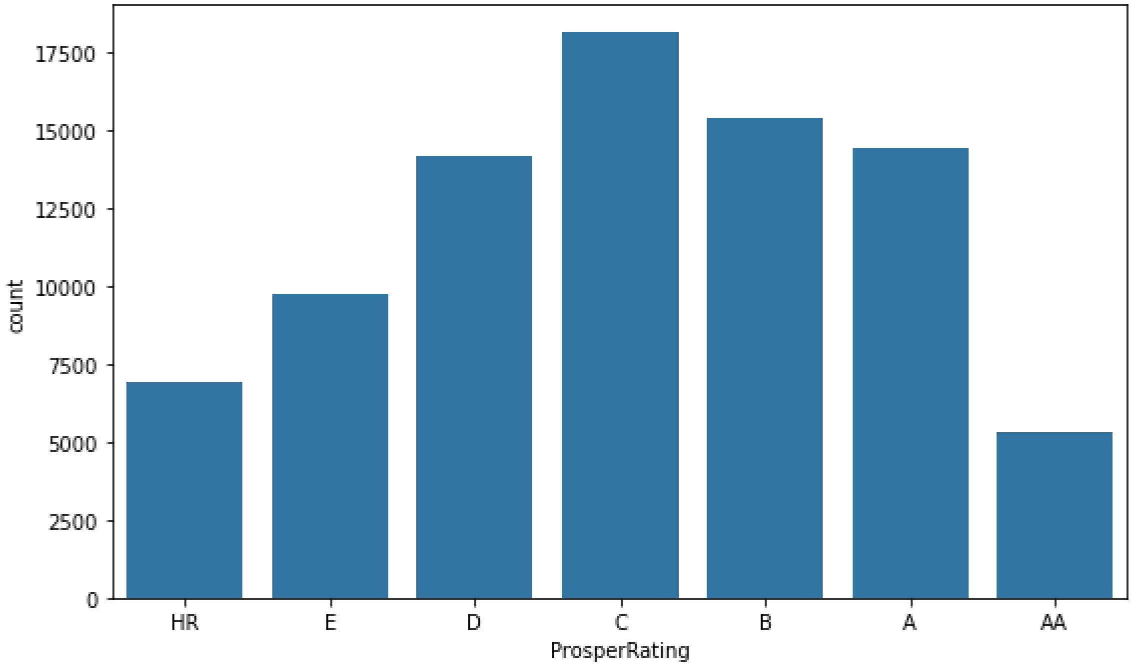
In [8]:
```
# Set color for all univariate plots
shades = sns.color_palette()[0]

# Create 40 evenly spaced bins for Borrower APR from zero to the maximum value
bins = np.arange(0, Loan_clean.BorrowerAPR.max()+0.06, 0.01)

plt.figure(figsize=(8, 5))
plt.hist(data=Loan_clean, x='BorrowerAPR', bins=bins, color = shades);
plt.xlabel('Borrower APR');
```



The Distribution of Prosper Rating

> The distribution of `Prosper Ratings` is a one mode distribution, with the most common rating belonging to the central grade represented (**C**), has the highest ratings, followed by (**B**) and then (**A**). (**AA**) are the least common.

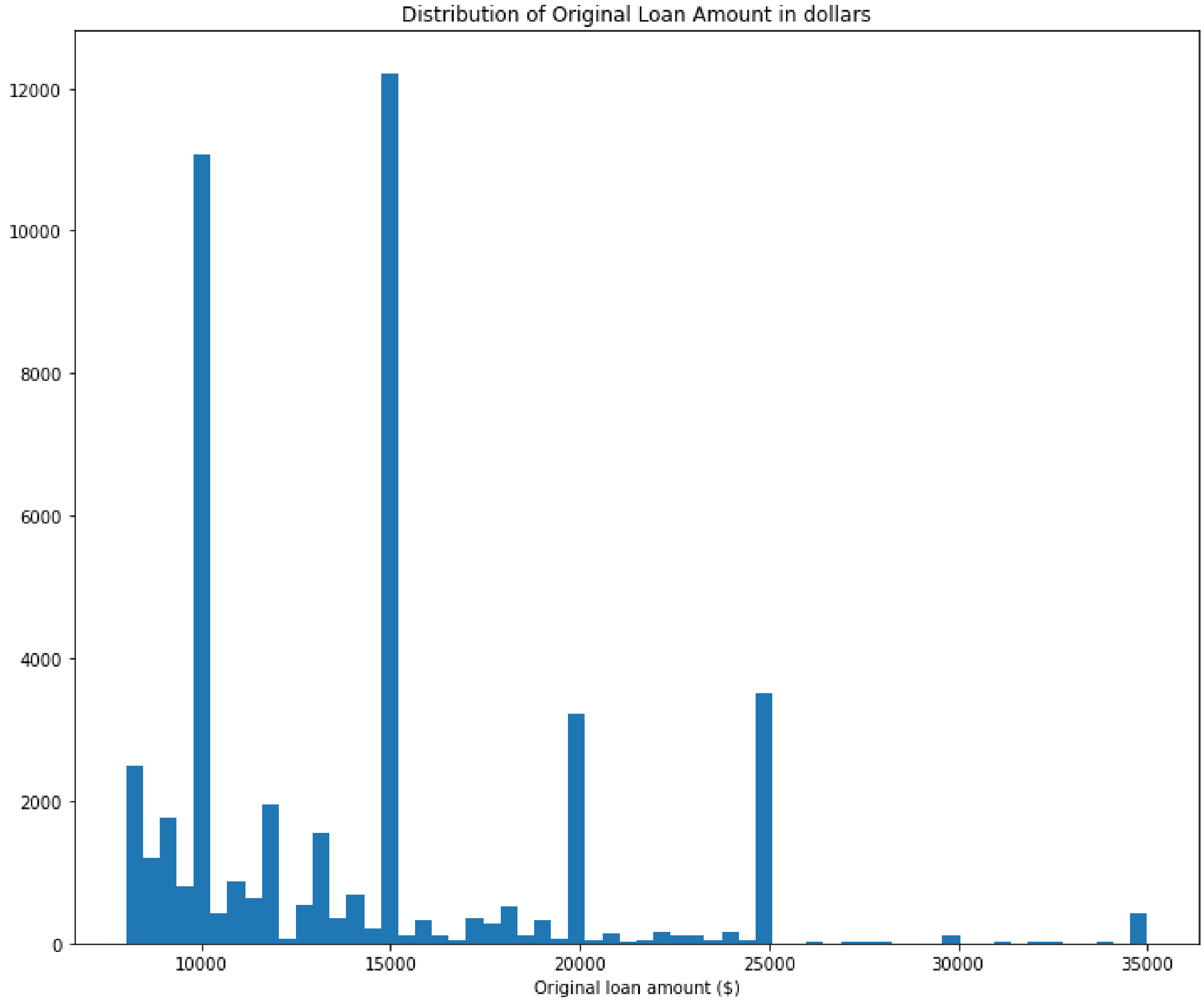
In [9]:
```
fig = plt.figure(figsize=(20, 12))
# Prosper Ratings
plt.subplot(2,2,1)
sns.countplot(data=Loan_clean, x='ProsperRating', color = shades);
```



The Distribution of Original Loan Amount

> The higheest amount of Loan being taken from this sample frequently are 15,000 followed by 10,000,25,000 and then 20.000(dollars). The visualization also shows that people take smaller amounts more often that big amount of money from the compnay.

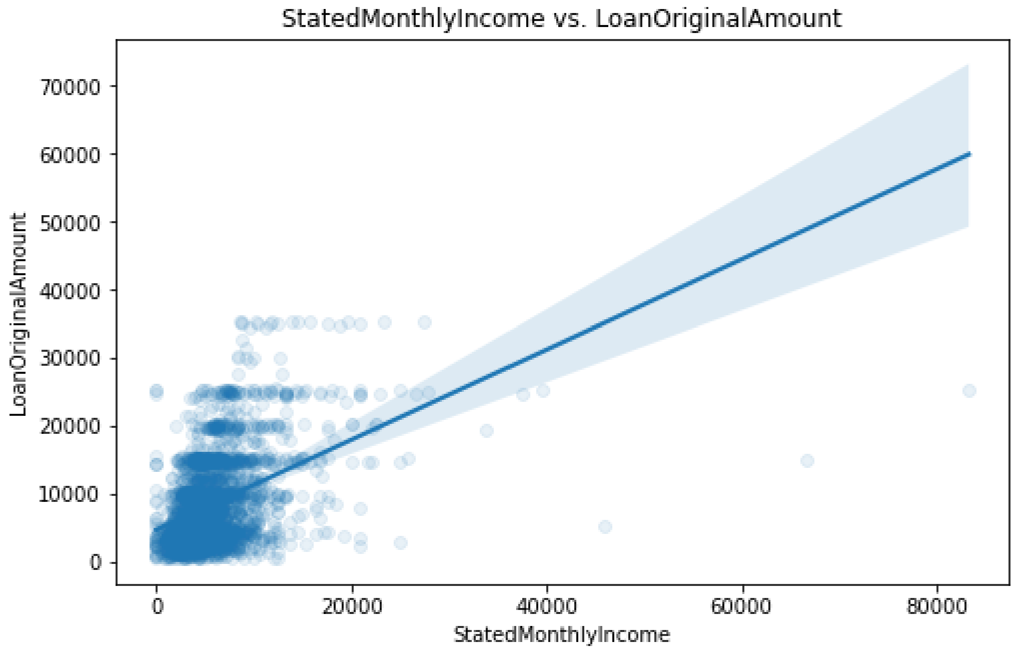
In [11]:
```
# Distribution of LoanOriginalAmount
bins = np.arange(8000, Loan_clean.LoanOriginalAmount.max()+400, 450)
plt.figure(figsize=[12, 10])
plt.hist(data = Loan_clean, x = 'LoanOriginalAmount', bins = bins);
plt.xlabel('Original loan amount ($)');
plt.title('Distribution of Original Loan Amount in dollars');
```



A closer look on the relationship between `StatedMonthlyIncome and BorrowerAPR`

> The perfect Analogy to this visualization is " the Higher you earn the higher loan you take". We also see here that low income earners took more loan that the higher income earners

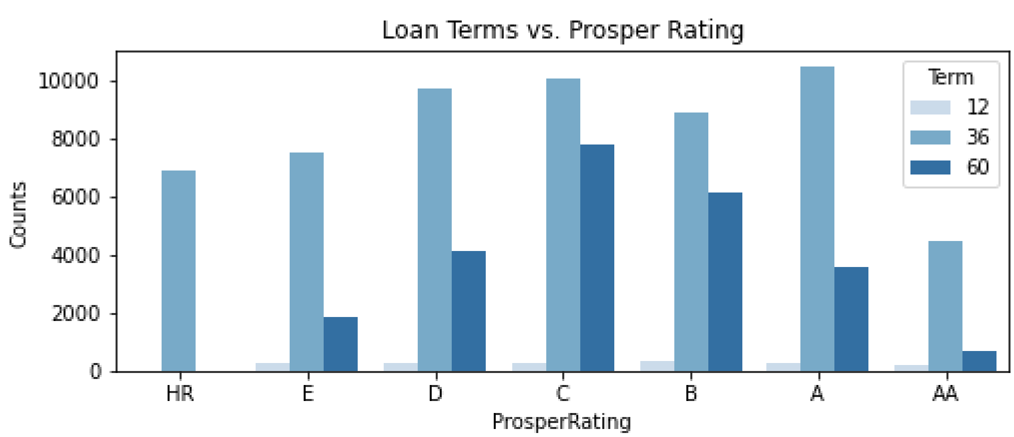
In [12]:
```
# we will also use the same saample size (300) for consistency
plt.figure(figsize=(8, 5))
sns.regplot(data=Loan_clean.sample(3000, random_state=1), x='StatedMonthlyIncome', y='
            x_jitter=4, color= shades, y_jitter=500,scatter_kws={'alpha': 0.1});
plt.xlabel('StatedMonthlyIncome')
plt.ylabel('LoanOriginalAmount')
plt.title('StatedMonthlyIncome vs. LoanOriginalAmount');
```



The Relationship between `ProsperRating` and `Term`

> From the above visualization, Most people took the 36 months loan term, followed by the 6oth month, few people took the 12 month. i will really like to know the income and employment statues of these distribution to clearly reason why people choose 36 months over 60 months. May be people just want to be moderate in their loan terms." payoff your loan and take another one"

In [13]:
```
plt.figure(figsize = [8, 10])

# subplot 1: Prosper rating vs term
plt.subplot(3, 1, 1)
sns.countplot(data = Loan_clean, x = 'ProsperRating', hue = 'Term', palette = 'Blues')
plt.xlabel('ProsperRating')
plt.ylabel('Counts')
plt.title('Loan Terms vs. Prosper Rating');
```

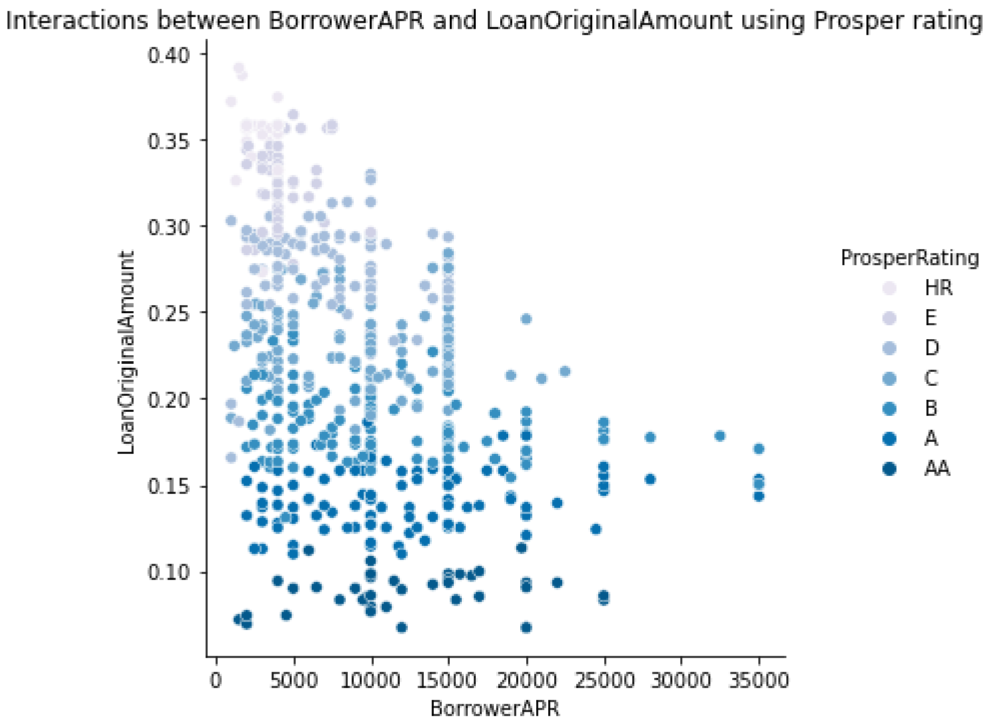


Interactions between `BorrowerAPR` and `LoanOriginalAmount`

using `Prosper rating`

> The relationship between the three variables is clear t. People with low prosper ratings take smaller loans at higher percentage rates, while those with higher prosper ratings enjoy higher loan amounts at lower rates.

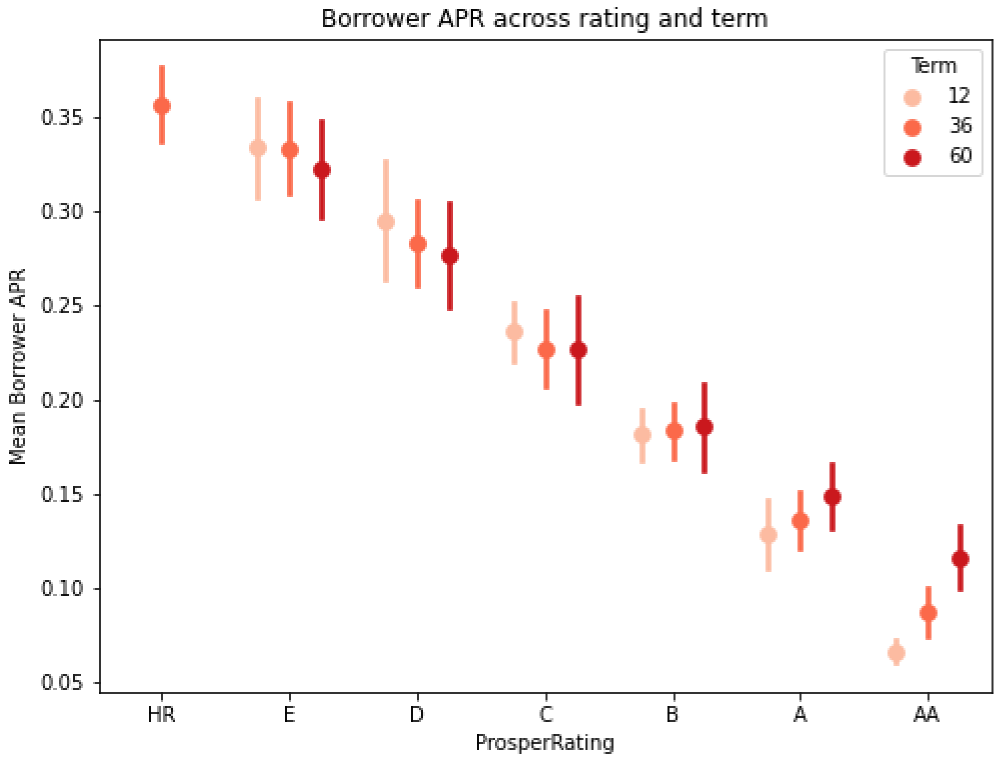
In [14]:
```
# Visualize interactions with a seaborn relplot
sns.relplot(data=Loan_clean.sample(800, random_state=1), y='BorrowerAPR', x='LoanOrigi
            hue='ProsperRating', palette='PuBu', height=5, aspect=1);
plt.xlabel('BorrowerAPR')
plt.ylabel('LoanOriginalAmount')
plt.title('Interactions between BorrowerAPR and LoanOriginalAmount using Prosper ratin
```



Interactions between Ratings, APR and Loan Term

Interestingly, the borrower APR decrease with the increase of borrow term for people with HR-C raings. But for people with B-AA ratings, the APR increase with the increase of borrow term.

In [15]:
```
# for Prosper Rating, Borrower APR and  LoanTerm
fig = plt.figure(figsize = [8,6])
ax = sns.pointplot(data = Loan_clean, x = 'ProsperRating', y = 'BorrowerAPR', hue = 'T
           palette = 'Reds', linestyles = '', dodge = 0.5, ci='sd')
plt.title('Borrower APR across rating and term')
plt.ylabel('Mean Borrower APR')
ax.set_yticklabels([],minor = True);
```



Summary

> This Prosper loan data explorations was aimed at the motivations accross different borrowers when applying for loan, and also the different factors that maay affect the Success of a Loan application.
>
> In terms of borrower motivations, we found surprising results. Rather than take loans to start businesses or purchase assets, the largest population seems to take large amount of Loans t finance weddings, child adoptions, boat acquisitions, and the purchase of engagement rings; THis are things that oridinarily doesnt have any return on investment, hence money spent on them can not be recovered. This explains the reason, we have Many people taking Loans just to finance their debts. Debt consolidation also accounts for the highest loan amounts collected from the platform on average. While all these may point to possibly self indulgent reasons, it can also be the liefstyle of the people living in that Region
>
> We also measured loan favorability using the annual percentage rate attached to a loan (the Borrower APR). we found out that the sucess of most Loan application is influenced positively by high and verifiable incomes, homeownership, low debt to income ratio, and the presence of a current means of employmenWe ,Also Borrower APR is negatively correlated with the loan original amount, loan term, and prosper rating(these are only detailed informations needed while filling out an application form not an asset that might be considered as collateral).
>
> On further exploration, another surprising interaction was discovered. There seemed to be a dichotomy in the interaction between borrower APR and prosper

> ratings. Between the lower ratings of HR to B, borrower APR and prosper ratings were negatively correlated. This interaction turns positive within the high prosper rating group (B to AA). We attributed this to the possible influence of lurking variables, such as the loan term, and borrowing power of high income earners who are usually rated higher on the prosper scale. High income earners seemed to borrow more when long-term loans are involved, increasing their debt to income ratio. Hence, an increase in APR might be a great way to disincentivize 'overborrowing'. On the other hand, decreasing APR by term might be a great way to encourage low-rated borrowers to take long-term loans; giving them enough time to repay their loan.