# Capstone Project 1: Milestone Report

URLease is a leasing company that has had trouble being profitable because of their clients' inability to pay their leases back in full. To remedy this problem, they have hired me, their Data Scientist to build a predictive analytics model that would provide a score to be used upon application to automatically approve or deny lease applications based on internal data and third party bureau data.  The goal of the model would be to predict the likelihood of an applicant's ability to pay back the lease in full.

**Data Acquisition, Wrangling and EDA.**

The data that will be used in building the model is comprised of three data sets:

1. A Lending Club data set from Kaggle that consists of accepted applications from 2007 through 2018 which I will be filtering down to the last two years: 2017-2018.

2. A Bureau of Labor Statistics – State Unemployment data set for 2017-2018 which I had to build from several files downloaded from their website.

3. A World Population Review State Abbreviation data set to be used to link the first data set with the second.

The Lending Club data set from Kaggle came somewhat clean and thus did not need much cleaning.  From the second file, I removed a few variables such as 'labor force' ,'employment' and ,'unemployment' which I will not use. I also removed the '(R)' from the unemployment variable values and I renamed variables to not include spaces in the names. From the third file I removed the  'Abbrev' variable.

After merging the files together, I removed merge key variables that the merging process generated. I also removed variables that have 0 non-nulls. Then I removed a variable that had a constant value. I then removed 20 variables that had 100% missing values or virtually 100% (99.9%) as these would have no value to the model.

I then looked at a large number of variables that had missing values where I first took a look at possible outliers to determine what to do before imputing missing values. I ended up finding the outliers and capping the values to 3 standard deviations around the mean for continuous variables. For imputations, I used the mean for continuous variables, -1 for discrete numeric variables, ‘Unknown’ for categorical variables, and ‘Jan-1900’ for a Date variable.

I then used the Pandas Profiling Report to simplify two variables with High Cardinality by keeping the five values with the highest frequencies and bunching all others into a new category I named ‘Other’. I dropped another two variables with High Cardinality because they were Date variables I will not be using in the model.

I then removed another variable that was flagged as unique on the Pandas Profiling Report.

Next, I worked through 11 variables that were flagged as Highly Skewed on the Pandas Profiling Report. For those, I identified outliers in the same way as before and I removed them.

**Initial Findings from EDA**

During the Exploratory Data Analysis (EDA) Phase of this Capstone Project 1, I generated the PandasProfilingReport.html file. I did this because of the large number of variables in the file (150 variables.) In looking at the results, I focused on finding if there were correlations between some of the variables.

At this point I looked at the list of variables available and I looked at their graphs within the PandasProfilingReport.html file.  I wanted to see if there was a correlation between fico score and Performance based on loan status. My assumption was that there would be a clear correlation between the Fico Score and Performance.

I first took a look at the distribution of unique values in loan status.

```
Current                213499

Fully Paid              23811

Charged Off              4416

Late (31-120 days)     3658

In Grace Period        1418

Late (16-30 days)       815

Default                   4
```

From the results above, I then created a new variable named Performance where Performance can take on three values: 1) Good, 2) Bad, or 3) Indeterminate.

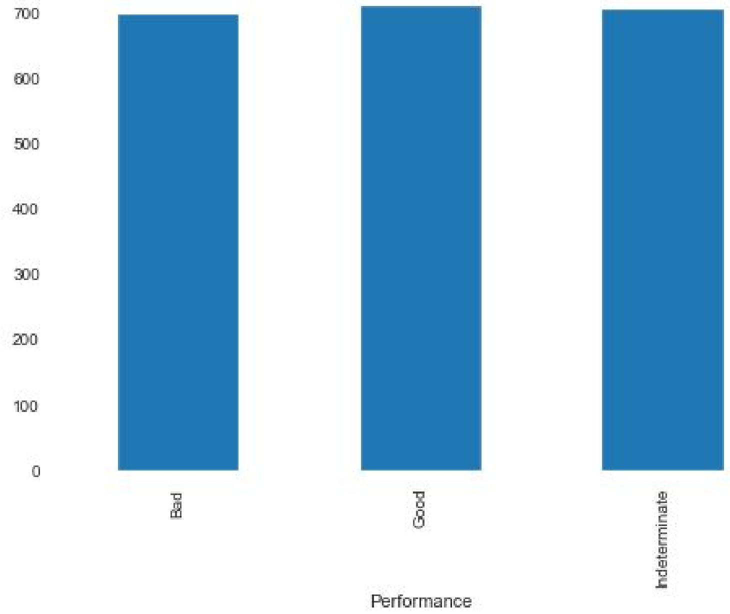
1) I defined as Good: records for which the loan_status was 'Fully Paid'.

2) I then defined as Bad: records for which the loan_status was either 'Late (16-30 days)', 'Late (31-120 days)' or 'Charged Off'.

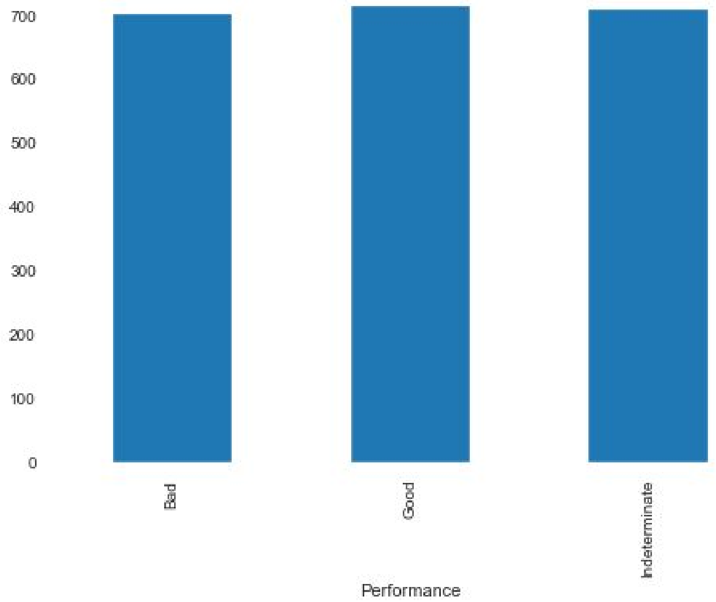
3) I then defined as Indeterminate: records for which the loan_status was ‘Current’, 'In Grace Period', or 'Default'.

```
Indeterminate    214921

Good             23811

Bad              8889
```

I then generated the mean value for a few variables by Performance value and plotted the results to see if I could find an obvious difference in the means. I first ran this test with the variable fico_range_low.
But as the results below show, there was no significant difference in the mean value across the Performance.

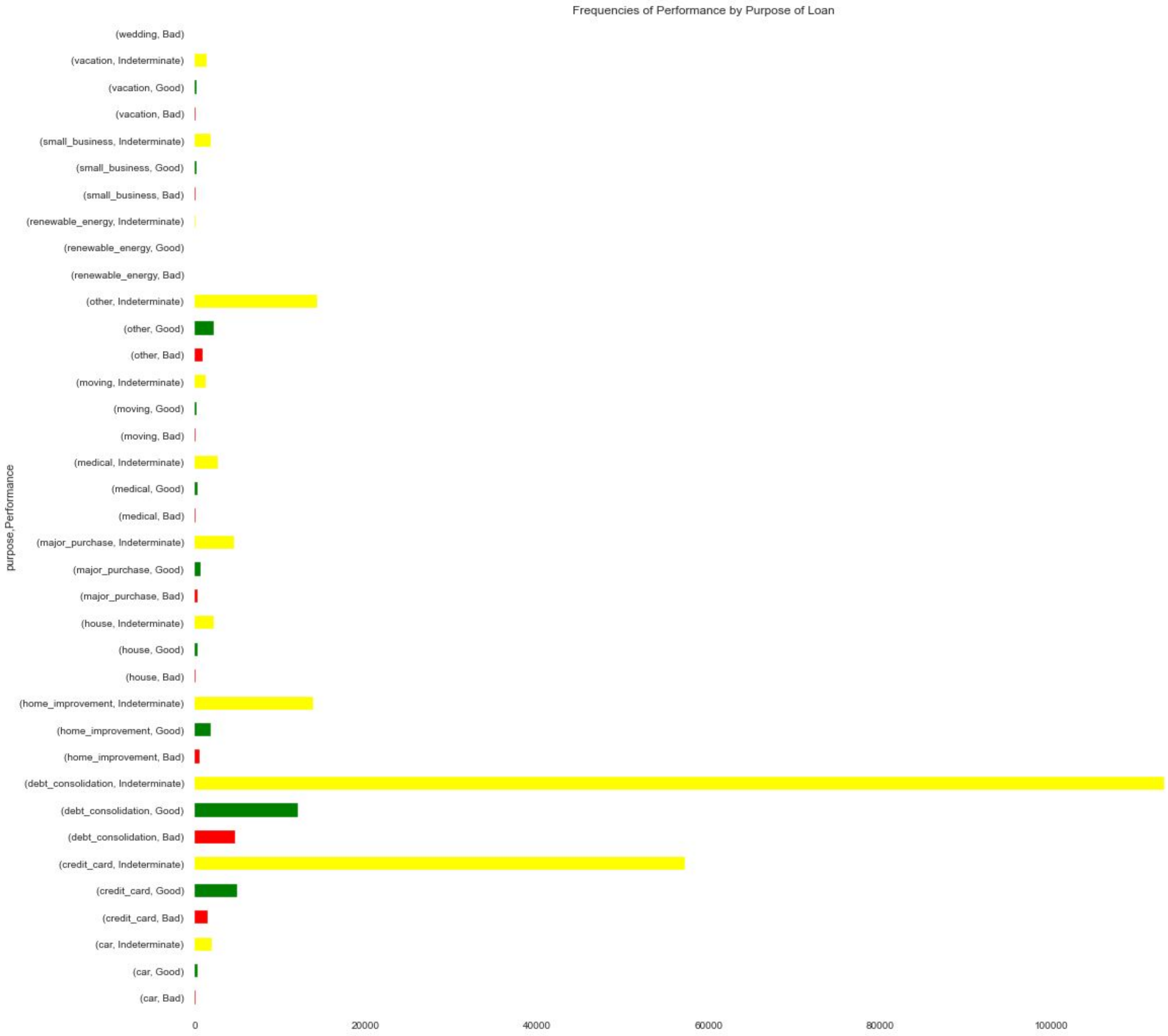


I next ran this test with the variable fico_range_high. Again, as the results below show, there was no significant difference in the mean value across the Performance.



My assumption was that there would be a high variance between at least Good and Bad Performance based on Fico Score. That was proved to not be the case.

Then I got the Performance distribution by Purpose as the assumption was that the Purpose of taking the loan might explain the performance on the loan. I first ran counts to look at the data by purpose and Purformance, but that wasn’t very helpful to clearly see, so I then created a bar graph for it.



In looking at the bar graph above, it wasn't very clear if there was a correlation between the Purpose for taking a loan and the Performance. One takeaway seemed to be that the number of Good for debt_consolidation seemed to be twice as the number of Good for credit_card, while the number of Bad for debt_consolidation seemed to be three times that of credit_card, which would indicate that it would be riskier to approve loans when the purpose was debt_consolidation.

My assumption that there would be a clear correlation between the Fico Score and Performance turned out not to be true. My assumption that there would be a clear correlation between the Purpose and Performance turned out to also not be true.

However, with Purpose, we saw that at least when comparing the debt_consolidation purpose with the credit_card purpose, we were able to determine that approving loans whose Purpose was debt_consolidation turned out to be riskier than approving loans whose Purpose was credit_card.

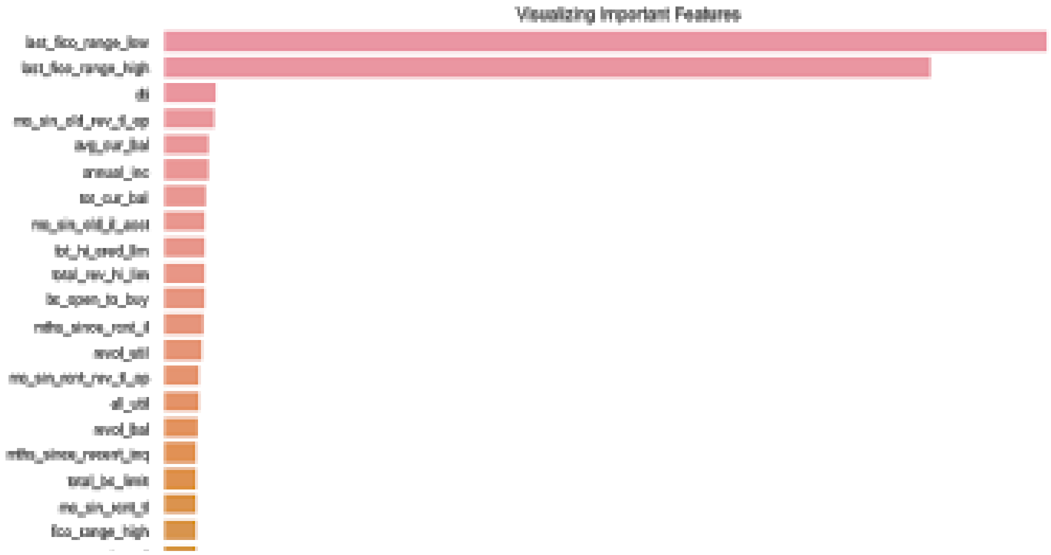
To further analyze the data with the goal of finding correlations between variables I generated a random subset of the data, small enough for quick analysis, but large enough to derive proper statistics. With this new data set, I generated a quick model to focus my attention on variables that would eventually be relevant to the model I would later build.  I did this because I have over 150 variables in my data set.

- To eliminate some of the variables up front, I looked at a Pandas Profiling file I had previously generated to find variables with categorical columns that are unique, have entirely missing values, are rejected, are unsupported, have high cardinality, or are date fields.
- I then had to isolate a list of categorical variables to be able to generate dummy variables for modeling, which consists of changing categorical variables into binary flag variables.
- With this list I generated the dummy variables
- I then found all the variables that had NaN values and imputed the NaN values

With the data in place to start building a model, I decided to build a Random Forest Classifier model. At this point I used the feature_importances_ variable and found that most of the top 45 variables in the model were Performance variables that I would have not had access to at time of lease application, so I had to remove those.

After removing them and rerunning a Random Forest Classifier model, I used the feature_importances_ variable again to see that the top 45 features were all ok to leave in the model.

I then built a bar plot in Seaborn to look at the features in order of importance to focus my research on feature correlations. I looked at the top 3 variables and tried to see if they were correlated with the Perf response variable.  Below is a snippet of the full graph produced via Seaborn.



I then generated a Logistic Regression model.

I used the Pearson correlation coefficient to see how correlated the variables were to the Performance response variable. The coefficient takes on values between -1 and 1, with 0 meaning no correlation while 1 is positively correlated and -1 is negatively correlated.

- The top variable last_fico_range_low had a Pearson correlation coefficient of 0.5791423235578518, which means it is somewhat highly correlated.
- The second variable last_fico_range_high had a Pearson correlation coefficient of 0.7102556649422719, which means it is even more highly correlated
- The third variable dti (Debt to Income ratio) had  a Pearson correlation coefficient of -0.04553847373302067, which means it is not correlated as the coefficient is extremely close to 0.

From this analysis, I would have thought that the most relevant feature for the model would have also been the most correlated to the response variable. As we can see from the results on the comparison of the three most relevant variables in the model, the second most relevant is the most correlated to the response variable, while the first is still strongly correlated, and the third is not correlated.