Loan Approval Prediction

Mohammed Amine Frakso

**Executive Summary**

Banks and lenders take several variables into account when approving loans. Determining whether a given borrower will fully pay off the loan or cause it to default is difficult. If the lender is too strict, fewer loans get approved, which means there’s less interest to collect. But if they’re too laid-back, they end up approving loans that default which could jeopardize the financial stability of the institution or the entire economy if those institutions are too big to fail. A good example is what happened when the housing market bubble burst in 2008. In this Project I will be working on a loan prediction dataset, the question I’m trying to answer is whether a borrower would payback or not a loan and what are the variables that affects this outcome. In this project, the dataset was cleaned first, and the exploratory data analysis and feature engineering were performed. The strategies to deal with both missing values and imbalanced data sets were covered. Then we proposed five machine learning models to predict if the applicant could repay the loan. As we expected, borrowers with higher annual income and higher lower interest rates and smaller installments, are more likely to pay the loan.

**Background of the problem**

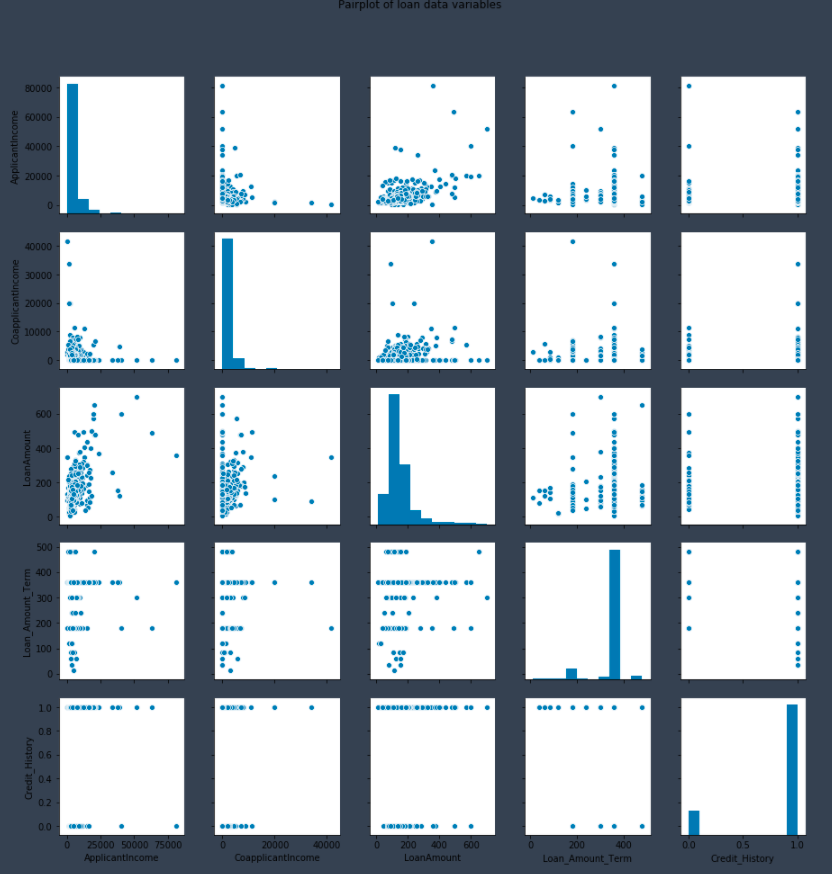
Loan repayment prediction is a very common real-life problem that each retail bank faces every time they are lending money to their clients, if done correctly, it can save a lot of man hours and money for the bank. Through the project, I will be starting by getting the system ready and loading the data, understanding the data and do some Exploratory Data Analysis (EDA) by plotting the data. The end goal is building a prediction model to solve this business problem, in a way that the bank can be assured it will be getting the money back. Hence the more accurate the model is in predicting the eligible customers the more beneficial it would be for the Bank.

**Methods**

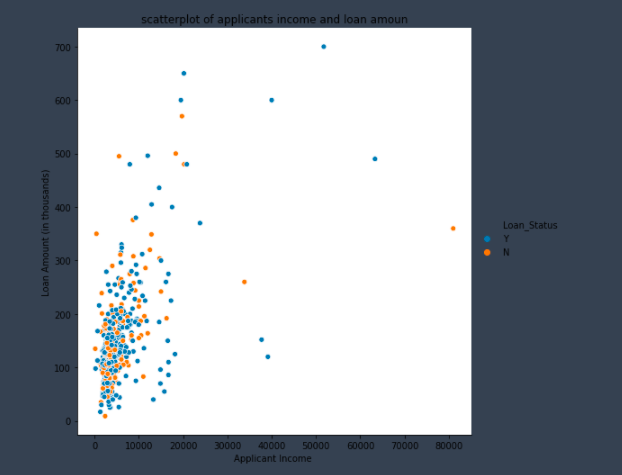
My datasets are divided into train and test and have 13 variables of details of applicants for loan and status whether the loan was approved or not. Using data analysis techniques, I will fit the data into best classification models and a binary classification model with maximum accuracy.

Data cleaning is an essential step in any project. We have some missing data that we will need to handle. We have some columns that are non numeric which will need some pre-processing.I will transform categorical variables Since the model only accepts numeric data, so I will use LabelEncoder module of sklearn library to encode the object datatype features. This will change their datatype to integer as well.

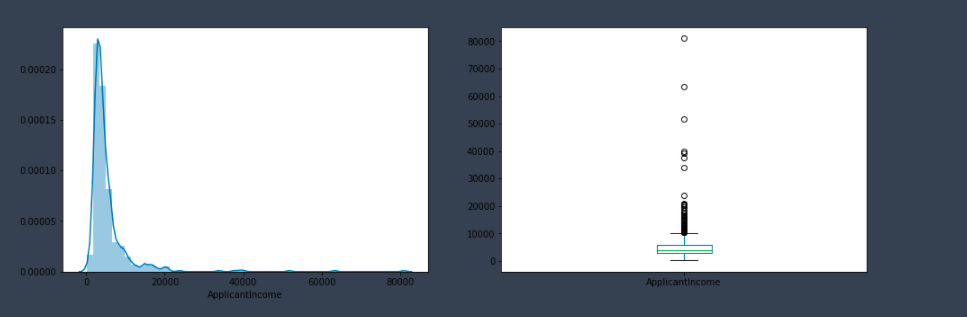
We will plot pairs plot which will allows us to see both distribution of single variables and relationships between two variables. This will make it easy to see the trends for follow-up analysis.



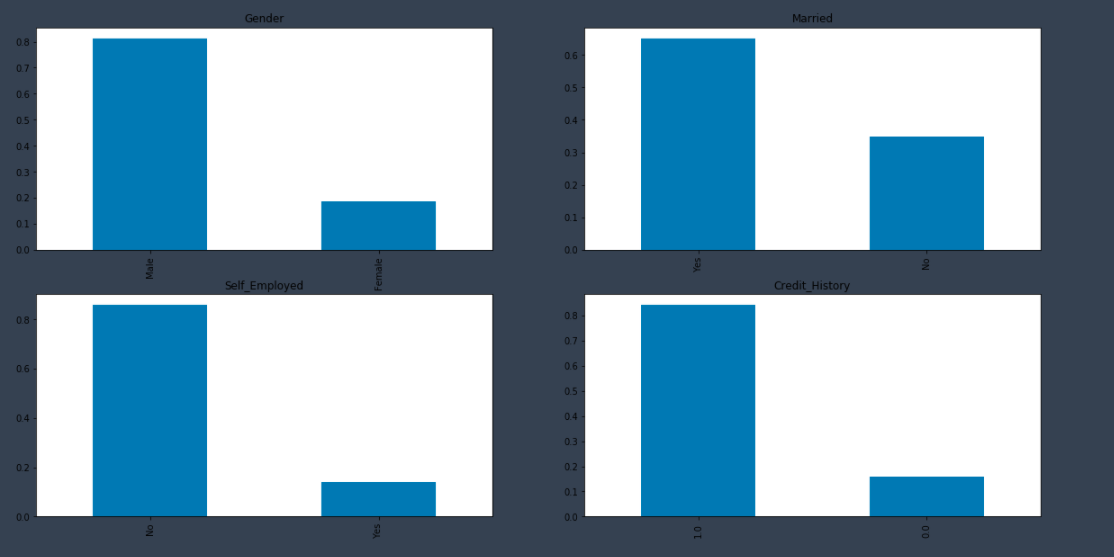
The scatter plot between loan amount and applicant income, then loan amount and term are trying to tell us something



We can see that most points distribution are at left bottom of the figure which indicates that most applicants wanted to have a lesser loan amount.



# The distribution of income is left skewed and have heavier tails than a normal distribution indicating some outlier values towards higher income side.

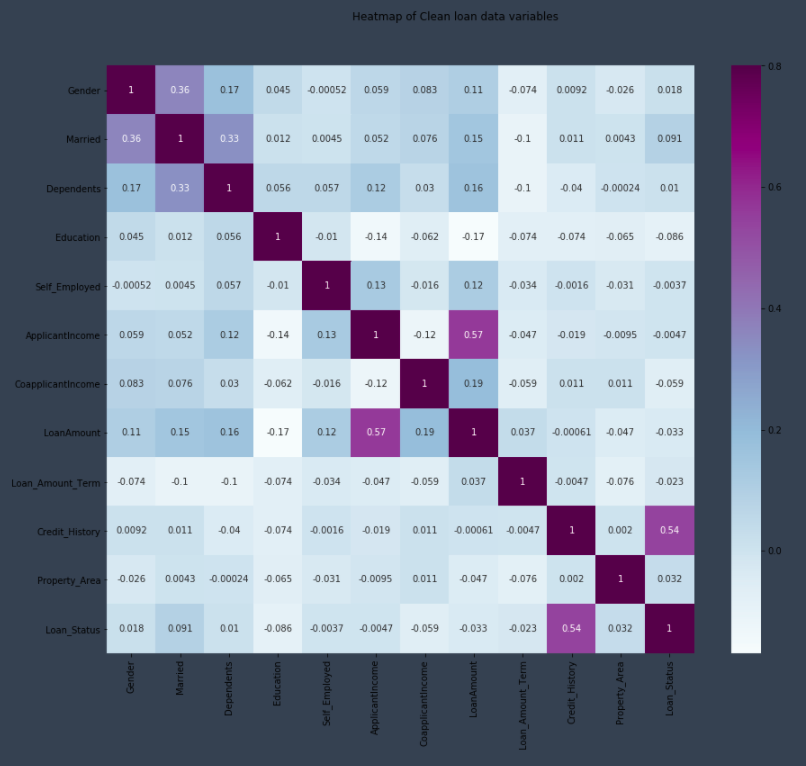


In our Data we have more male applicants than females,

most of the applicants are married

Only few are self employed

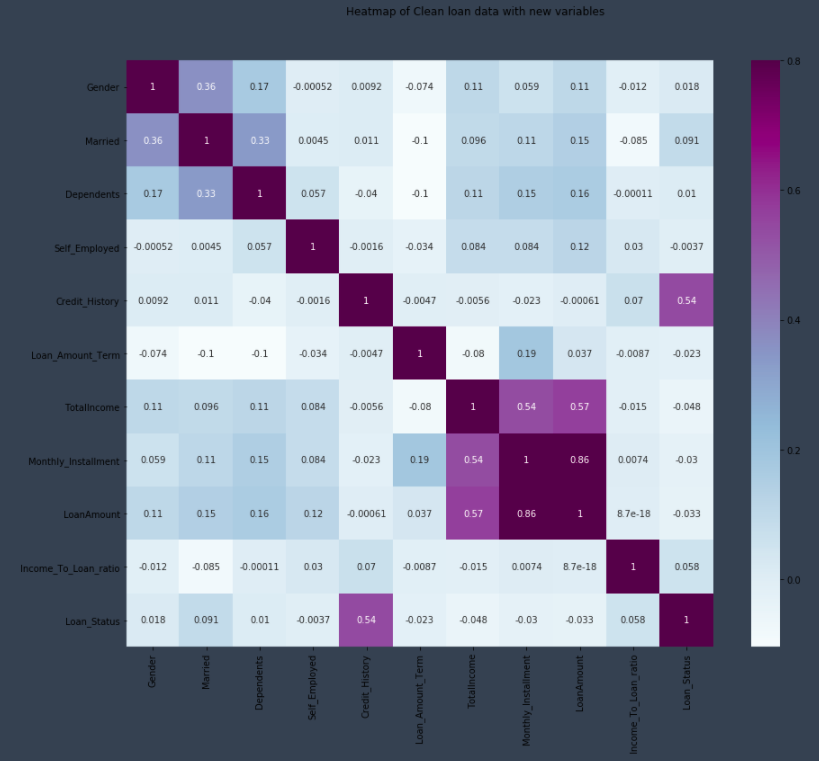
Most of applicants have a good credit history



From my clean data, I am going to remove the Property Area feature because the heatmap suggests that this feature doesn't have any strong effect on any other variable including the target variable.

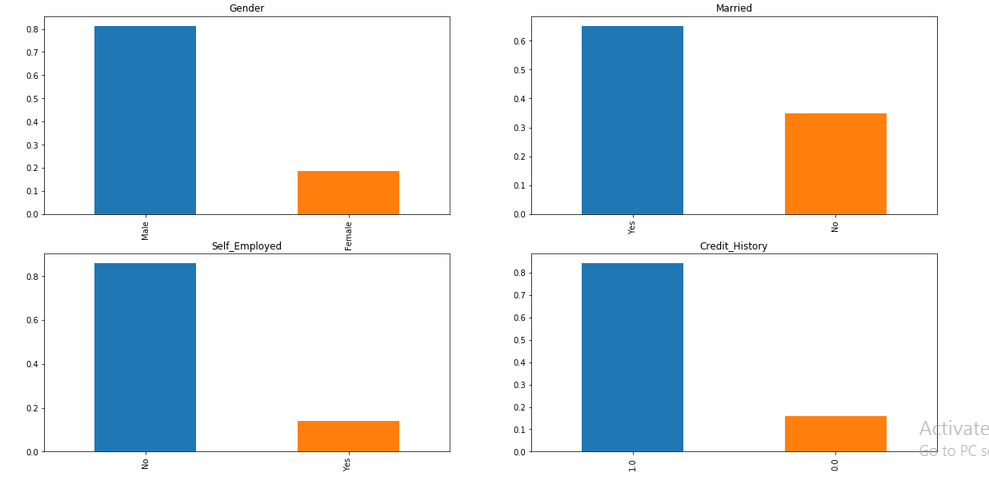
**Feature engineering**

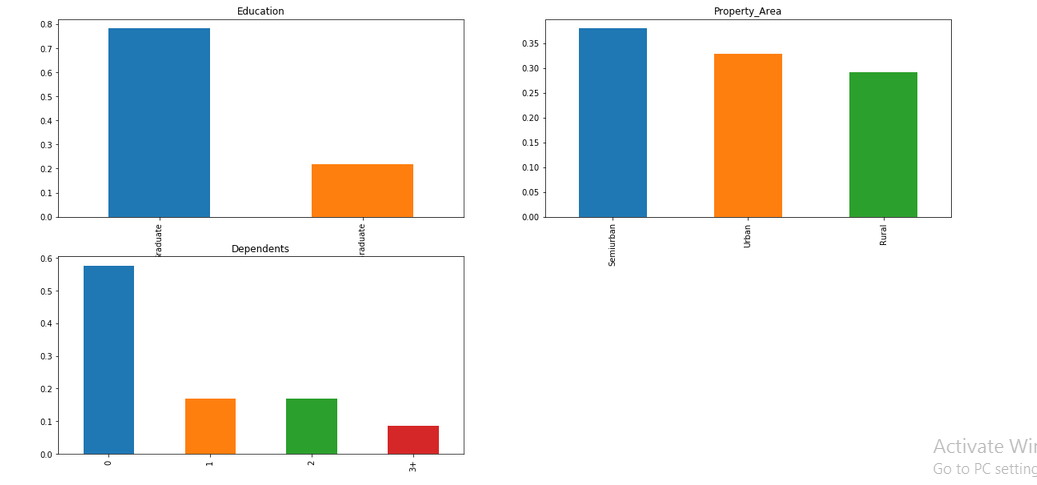
Based on the domain knowledge, we can come up with new features that might affect the target variable. We will create a new feature which is the monthly amount to be paid by the applicant to repay the loan. The idea behind making this variable is that people who have high Monthly amounts to be paid might find it difficult to pay back the loan. We can calculate the monthly amount by taking the ratio of the loan amount with respect to the loan amount term.



**Results**

After cleaning the data doing some feature selection/engineering and building different models I come to the following conclusions:



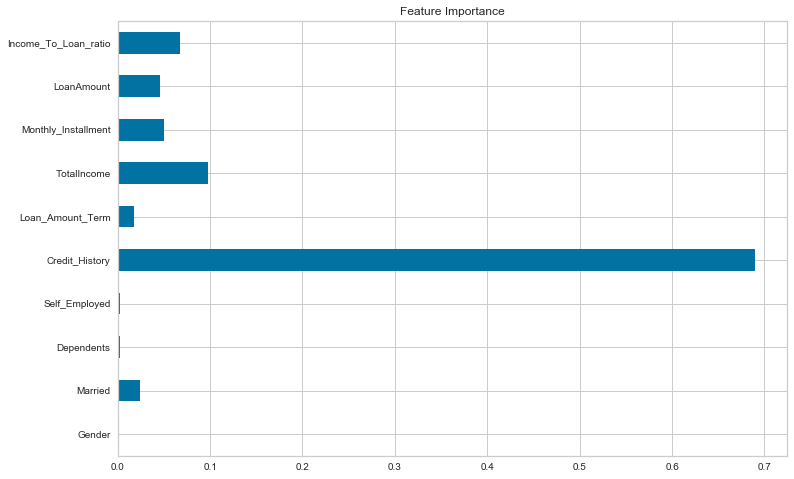


By looking at the column’s description in the above table and the graphs, we can make many assumptions and generate some hypothesis like:

* Clients with higher salary might have a greater chance of loan approval.
* Clients who are graduate have a better chance of loan approval.
* Married people would have an upper hand than unmarried people for loan approval.
* The applicant who has a smaller number of dependents have a high probability for loan approval.
* The lesser the loan amount the higher the chance for getting loan.
* We can see that approximately 81% are Male and 19% are female.
* Percentage of applicants with no dependents is higher.
* There are a greater number of graduates than non graduates.
* Semi Urban people is slightly higher than Urban people among the applicants.
* Larger Percentage of people have a good credit history.
* The percentage of people that the loan has been approved has been higher rather than the percentage of applicant for which the loan has been declined.
* **Classification of our models**

1. Keras Deep learning model gives: 82% prediction accuracy
2. Random Forest model gives: 81% prediction accuracy
3. Logistic Regression model gives: 81% prediction accuracy
4. Naïve Bayes model gives: 80% prediction accuracy
5. KNN with Grid Search model gives: 69% prediction accuracy
6. Decision Tree model gives: 68% prediction accuracy

* **Feature importance**



**Discussion**

Most classification problems in the real world are imbalanced. Also, almost always data sets have missing values. In our analyse we used different strategies to deal with both missing values and imbalanced data sets. We also explored different ways of building ensembles in sklearn. Below are some takeaway points:

* There is no definitive guide of which algorithms to use given any situation. What may work on some data sets may not necessarily work on others.
* Sometimes we may be willing to give up some improvement to the model if that would increase the complexity much more than the percentage change in the improvement to the evaluation metrics.
* In some classification problems, False Negatives are a lot more expensive than False Positives. Therefore, we can reduce cut-off points to reduce the False Negatives.
* When building multiple models, it is a good strategy to use good models that are as different as possible to reduce correlation.
* Missing values sometimes add more information to the model than we might expect. One way of capturing it is to add binary features for each feature that has missing values to check if each example is missing or not

**Conclusion**

In our Analysis, we demonstrated the use of machine learning algorithms on a well engineered dataset to predict loan repayment ability. To achieve the best performance, we displayed that data pre-processing, a careful selection of techniques of balancing dataset and classification algorithms are all very important.

Deep learning Neural networks work and Random forest models work quite well on our dataset, and the use Logistic regression is also effective.

Out of all the models we have tried, most of them were able to have test accuracy of almost 80%. Therefore, if anyone has a good credit history a good household income and a good income to loan ratio, then he will have a better chance of a loan application being accepted.

It is often very difficult to get the insights of loan approval from a Bank. This analysis provides interesting information about the likelihood of getting a loan approved from a bank.

We though that the decision could be made just with the variables we started with, but while developing our analysis we found out that by engineering new features we can help build better models. like Total income and Debt to income ratio.

We put some emphasis on the importance of Feature Engineering and how doing so can improve the overall accuracy of the model. However, performing proper and insightful feature engineering is a skill a data since can develop only by having some substantial subject matter expertise. They also need to prioritize planning ahead with a sustainable and logical business strategy, followed by the implementation.

High-level Predictive Analysis libraries like Scikit-Learn do most of the heavy lifting in the backend, yet, the libraries are robust enough to still yield surprisingly good results even with default parameters. With just a couple of lines of code, literally, anyone is capable of training a model on the dataset & submitting it.

The probability of default, it expresses the likelihood the borrower will not maintain the financial capability to make scheduled debt payments. For individual borrowers, default probability is most represented as a combination of two factors: debt-to-income ratio and credit score.

Credit rating agencies estimate the probability of default for businesses and entities that issue debt instruments, such as corporate bonds. Generally speaking, higher PODs correspond with higher interest rates and higher required down payments on a loan. Borrowers can help share default risk by pledging collateral against a loan.

In summary this dataset is old and small by today's standards. However, it did allow to create some models and then use various machine learning tools and techniques to look inside. At the beginning I thought that factors such as gender and education be major factors in the model. This dataset didn't show that. Instead, the Credit History of an applicant his current Total Income and income to loan ratio are the most critical factors in approving a loan. I actually feel like I've learnt a thing or two about how banks approve loans to their customers!

In the future, we want to continue exploring more sophisticated learning algorithms and dimension reduction and feature engineering techniques to further improve model performance on this significant prediction task. I suspect this sort of approach will become increasingly important as machine learning has a greater and greater role in shaping the banking industry.

**Acknowledgements**

I want to acknowledge and say thank you to my fellow classmates that have given peer reviews to help go through all the milestones leading up to this final paper.  I also want to Thank Professor Alsaleem for the feedback on previous millstones.

**References:**

Home Credit Default Risk. Kaggle, <https://www.kaggle.com/c/homecredit-default-risk/data>

Gorton, Gary, and James Kahn. ”The design of bank loan contracts.” The Review of Financial Studies 13, no. 2 (2000): 331-364..

Laurens van der Maaten, Geoffrey Hinton. ”Visualizing Data using t-SNE.” Journal of Machine Learning Research, 2008. [19] Visualising High-dimensional Datasets Using PCA and t-SNE.

Jason Brownlee. “A gentle introduction to k-fold cross-validation.”Accessed Oc-tober,7:2018, 2018.

Ofir Chakon. “PRACTICAL MACHINE LEARNING: RIDGE REGRESSIONVS.LASSO.” August 2017

<https://towardsdatascience.com/visualising-high-dimensional-datasetsusing-pca-and-t-sne-in-python->

<https://towardsdatascience.com/the-second-most-important-skill-to-have-as-a-data-scientist-2311f9efa143>

<https://www.kaggle.com/sazid28/home-loan-prediction>

<https://www.kaggle.com/muhammadwaqargul/credit-risk-modeling-with-different-ml-dl-models>