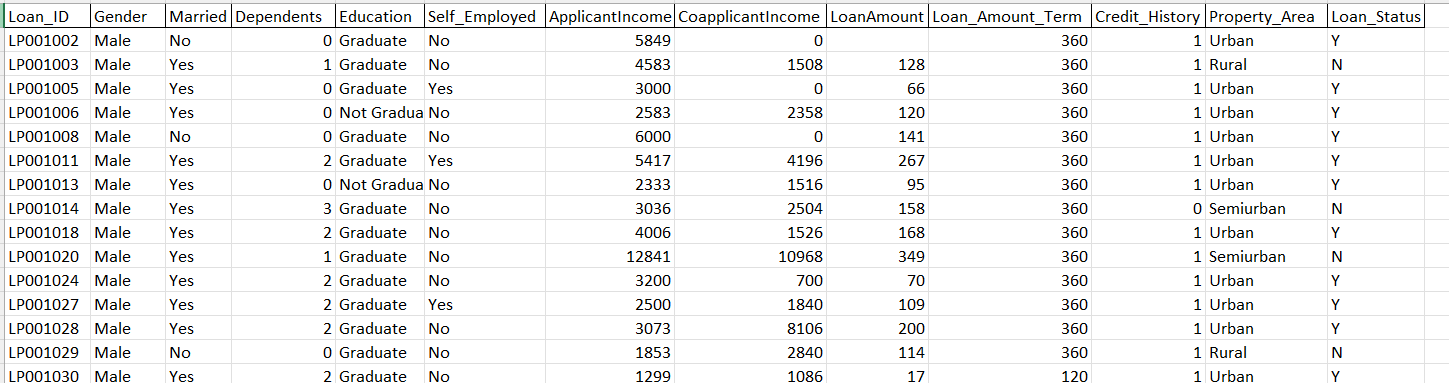
**Loan Approval Prediction using Machine Learning**

Loans are the major requirement of the modern world. By approving loans, banks get a major part of the total profit. But when it comes to deciding whether the applicant’s profile is relevant to be granted with a loan or not, banks have to look after many aspects.

**Dataset:**

* The raw data consists of 598 rows and 13 columns (features).
* It is a collection of financial records and associated information used to determine the eligibility of individuals or organizations for obtaining loans from a lending institution.
* Loan\_Status is the dependent feature (target or dependent variable) and the rest 12 are independent variables.

Glimpse of the dataset:



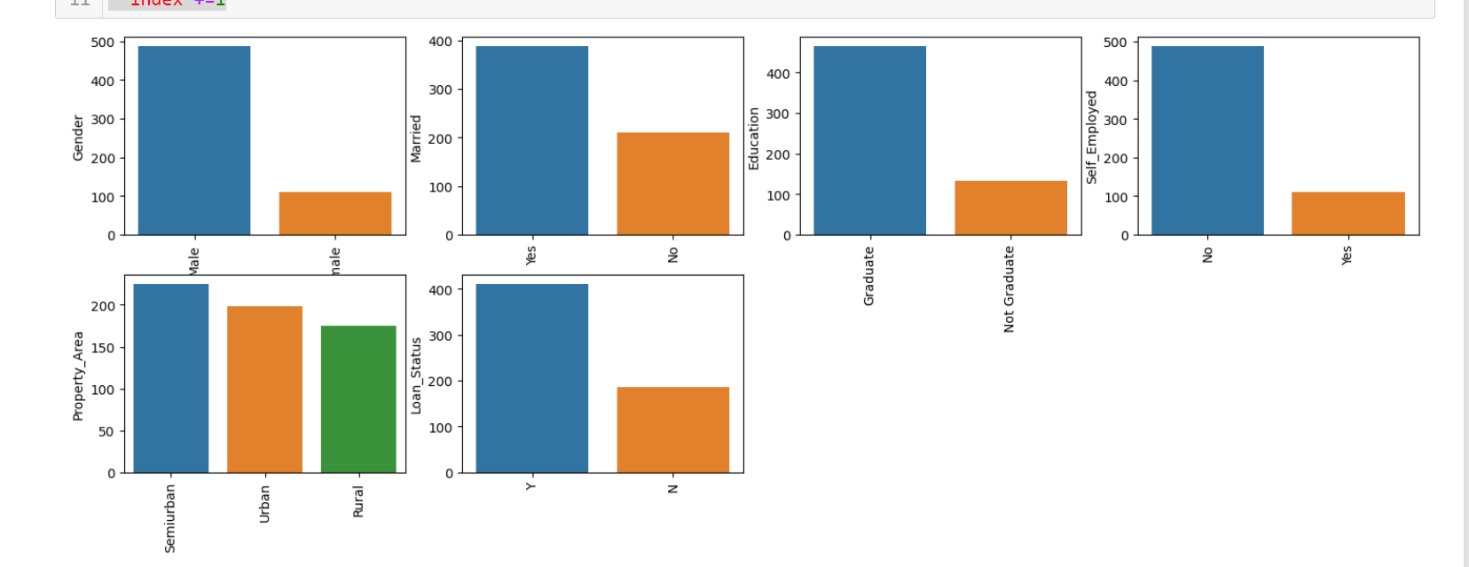
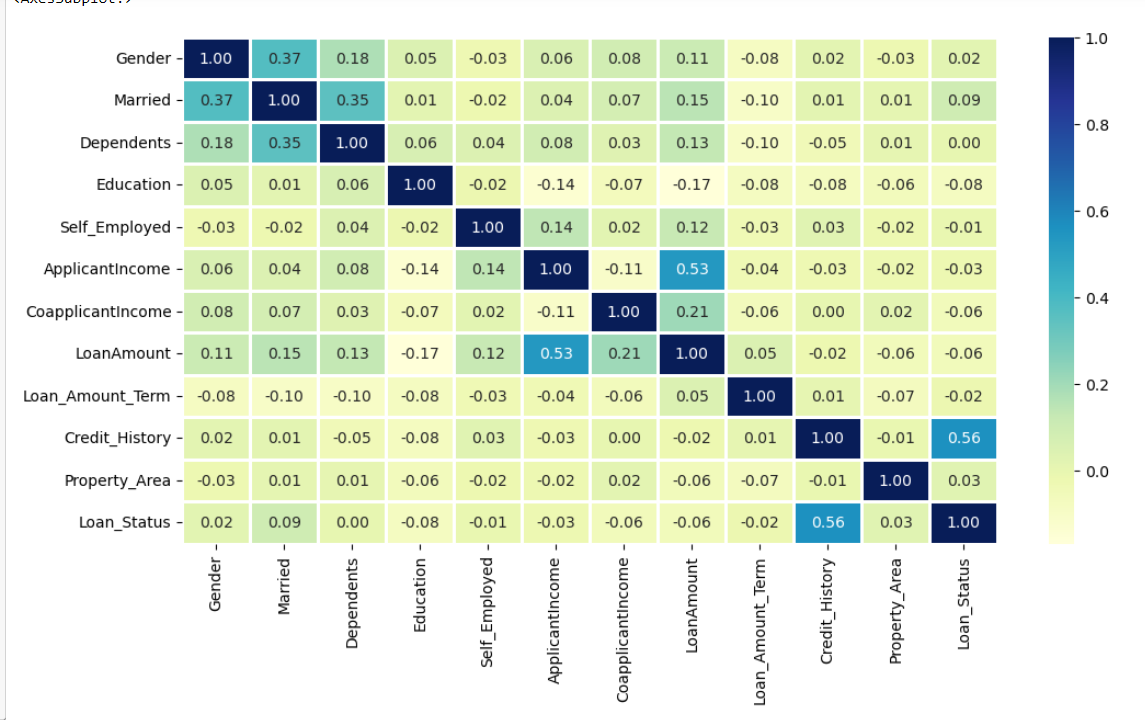
**Objective:**

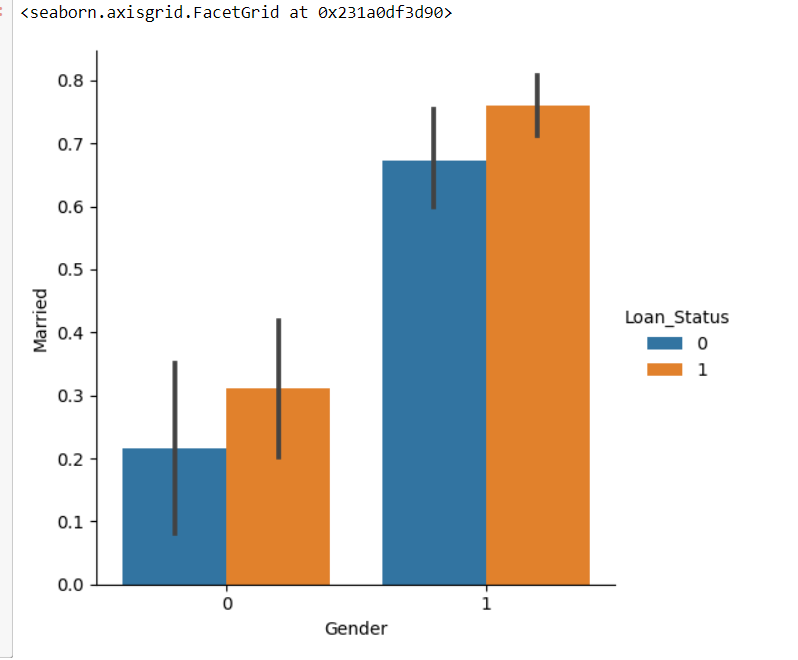
To predict whether a candidate’s profile is relevant or not for loan approval using features like Marital Status, Education, Applicant Income, Co-applicant Income, Credit History, etc.

**Data Importing and Preprocessing:**

* All the required and necessary libraries and the dataset are imported in Python.
* The number of columns of object datatype are found as Categorical variables, which are 7.
* The column Loan\_ID is dropped using .drop() function, as the column is completely unique and not correlated with any of the other column.

**Data Visualization:**

* A barplot is plotted for the six categorical variables – Gender, Married, Education, Self\_Employed, Property\_Area and Loan\_Status. It shows which value is dominating in the dataset. 
* We will use Label Encoder for all the columns containing categorical values as they are binary. This will change the categorical values into int datatype.
* The heatmap is plotted which shows the correlation between the numeric variables. It shows high correlation between Loan Amount and ApplicantIncome. It also shows that Credit\_History has a high impact on Loan\_Status. 
* Then, a Catplot is used to visualize the plot for the Gender, and Marital Status of the applicant.

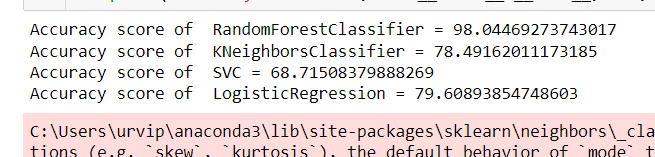
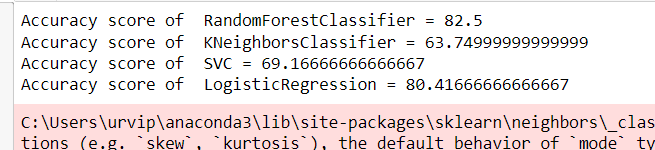


* The presence of missing values in the dataset is checked and it is found that there are no missing values in the dataset.

**Splitting the dataset:**

The entire dataset is split into training set and test set respectively.

**Model Training and Evaluation:**

* Since it is a classification problem, we will be using the following models:
* K Neighbors Classifiers
* Random Forest Classifiers
* Support Vector Classifiers (SVC)
* Logistics Regression
* To predict the accuracy, the accuracy score function from scikit-learn library is used.
* The accuracy of the different models is found to be: 
* Prediction on the test set: 

**Conclusion:**

* The Random Forest Classifier is giving the best accuracy with an accuracy score 82.5% for the test set.
* The K Neighbors Classifier gives the least accuracy with an accuracy score of 63.74% for the testing dataset.
* To get much better results, we can also use ensemble learning techniques like Bagging and Boosting.