UCS 1603: INTRODUCTION TO MACHINE LEARNING

MINI PROJECT

ON

Loan Eligibility Prediction

DONE BY:

Shrinisha N (18 5001 148)
Sriprabha AR (18 5001 167)
Suba Shree V S (18 5001 171)
Varsha R (18 5001 189)

PROBLEM STATEMENT

Loans are the core business of banks. The loan companies grant a loan after an intensive process of verification and validation.

Validation is based on a set of criteria such as Loan amount, dependents, marital status, applicant income, credit history and etc., which takes a huge amount of time. The entire process is done and validated manually all these years.

We have built a predictive model to predict if an applicant is able to repay the lending company or not based on information from the loan application. By automating this process, the model will be very useful for banks and other organizations to speed up their verification and validation process.

PROPOSED METHODOLOGY

LIBRARIES USED:

- Pandas
- Numpy
- Matplotlib & Seaborn
- Scikitlearn

DATASETS:

1. Train file :

Used for training the model, i.e. our model will learn from this file. It contains all the independent variables and the target variable.
Contains 12 independent and 1 target variable (LoanStatus)

2. Test file:

Contains all the independent variables, but not the target variable. We will apply the model to predict the target variable for the test data.
Contains only the 12 independent variables. Target variable is predicted using our trained model.

DATA:

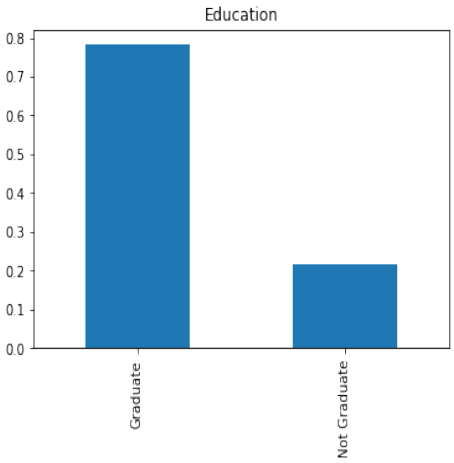
- Categorical features: These features have categories (Gender, Married, Self_Employed, Credit_History, Loan_Status)
- Ordinal features: Variables in categorical features having some order involved (Dependents, Education)
- Numerical features: These features have numerical values (ApplicantIncome, Co-applicantIncome, LoanAmount, Loan_Amount_Term)

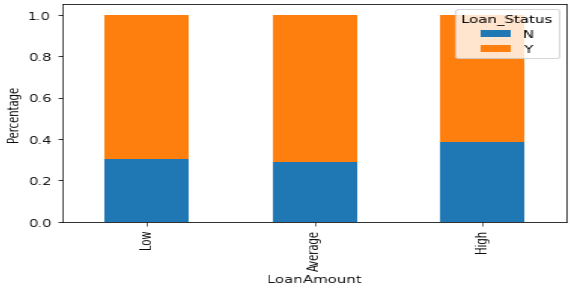
ATTRIBUTES	NON-NULL COUNT	DATA TYPE
Loan_ID	614	object
Gender	601	object
Married	611	object
Dependents	599	object
Education	614	object
Self_Employed	582	object
ApplicantIncome	614	int64
CoapplicantIncome	614	float64
LoanAmount	592	float64
Loan_Amount_Term	600	float64
Credit_History	564	float64
Property_Area	614	object
Loan_Status	614	object

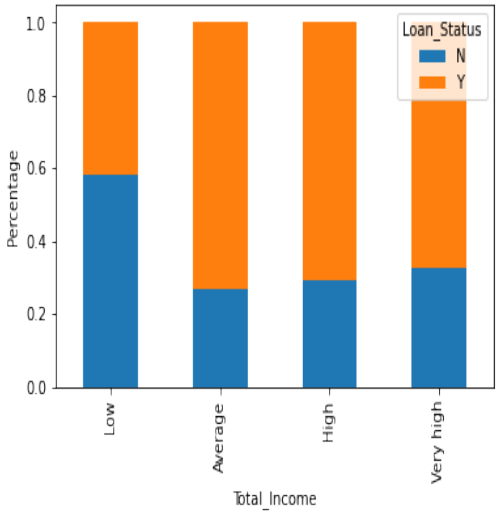
Data types: Object(string), int, float

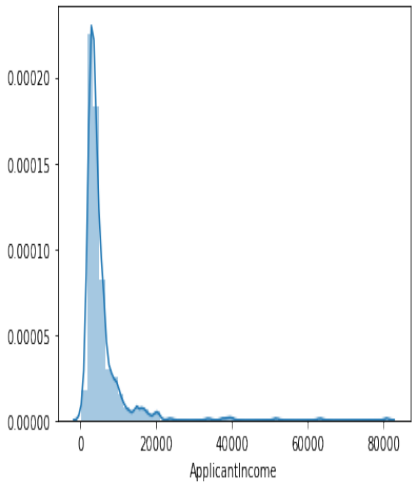
EXPLORATORY DATA ANALYSIS:

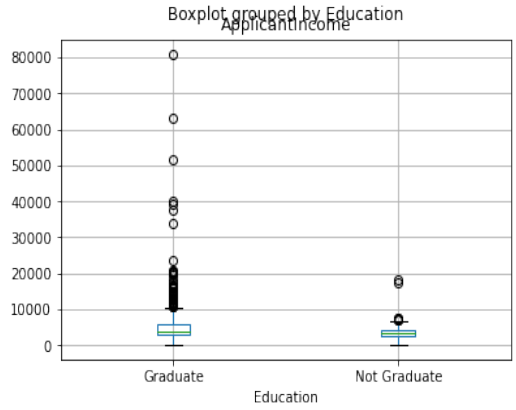
Performed data analysis using various visualizations tools such as graphs, barcharts, stacked barcharts, histograms and boxplots. Both univariate and bivariate analysis have been performed. Few images of the analysis are included below:

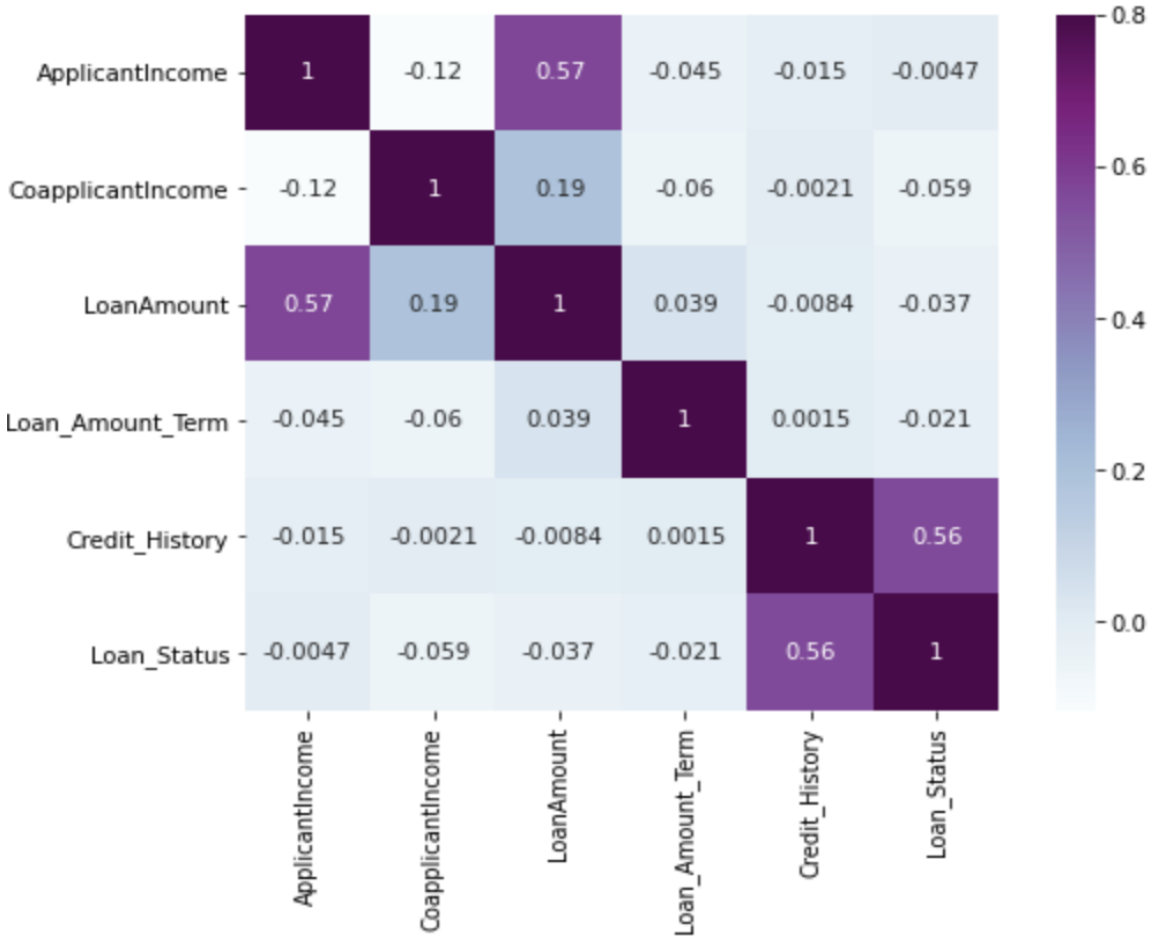






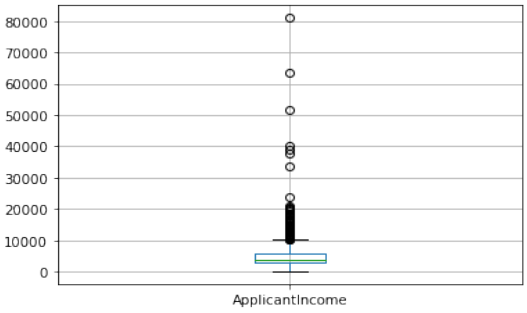


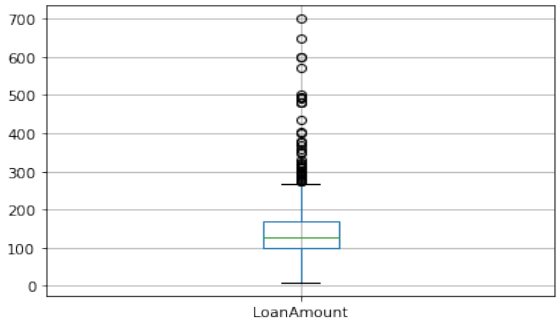




HANDLING OUTLIERS:

Presence of outliers would affect the mean and standard deviation of an attribute. The skewness in the data was removed by doing the log transformation. It does not affect the smaller values much but reduces the larger values. Hence we get a distribution similar to normal distribution.





Before Log Normalization:

	ApplicantIncome	CoapplicantIncome	LoanAmount
0	5849	0.0	NaN
1	4583	1508.0	128.0
2	3000	0.0	66.0
3	2583	2358.0	120.0
4	6000	0.0	141.0
...	...	...	...
609	2900	0.0	71.0
610	4106	0.0	40.0
611	8072	240.0	253.0
612	7583	0.0	187.0
613	4583	0.0	133.0

After Log Normalization:

	ApplicantIncome	CoapplicantIncome	LoanAmount	LoanAmount_log	TotalIncome	TotalIncome_log
0	5849	0.0	NaN	NaN	5849.0	8.674026
1	4583	1508.0	128.0	4.852030	6091.0	8.714568
2	3000	0.0	66.0	4.189655	3000.0	8.006368
3	2583	2358.0	120.0	4.787492	4941.0	8.505323
4	6000	0.0	141.0	4.948760	6000.0	8.699515
...	...	...	...	...	...	...
609	2900	0.0	71.0	4.262680	2900.0	7.972466
610	4106	0.0	40.0	3.688879	4106.0	8.320205
611	8072	240.0	253.0	5.533389	8312.0	9.025456
612	7583	0.0	187.0	5.231109	7583.0	8.933664
613	4583	0.0	133.0	4.890349	4583.0	8.430109

HANDLING MISSING VALUES:

Methods to fill missing values:

- Numerical Variables: Replace by mean or median
- Categorical Variables: Replace by mode

ATTRIBUTES	BEFORE	AFTER
Loan_ID	0	0
Gender	13	0
Married	3	0
Dependents	15	0
Education	0	0
Self_Employed	32	0
ApplicantIncome	0	0
CoapplicantIncome	0	0
LoanAmount	22	0
Loan_Amount_Term	14	0
Credit_History	50	0
Property_Area	0	0
Loan_Status	0	0
LoanAmount_log	22	0
TotalIncome	0	0
TotalIncome_log	0	0

MODEL BUILDING:

We have used few supervised learning classification algorithms to build our model since our target variable is a categorical variable (Eligible or Not Eligible)

The dataset is didvided into independent and dependent(target) variables – X and y
We have divided the dataset for training and testing in the ratio of 80 : 20 using the train_test_split function from sklearn library.

We have preprocessed our dataset and converted all the categorical data (gender, married, education, etc.) into numeric format for boththrain and test data. Label encoding was done using LabelEncoder( ) function of sklearn library.

We have also scaled the data by normalizing the range of independent variables using the StandardScaler function of sklearn

Algorithms Used:

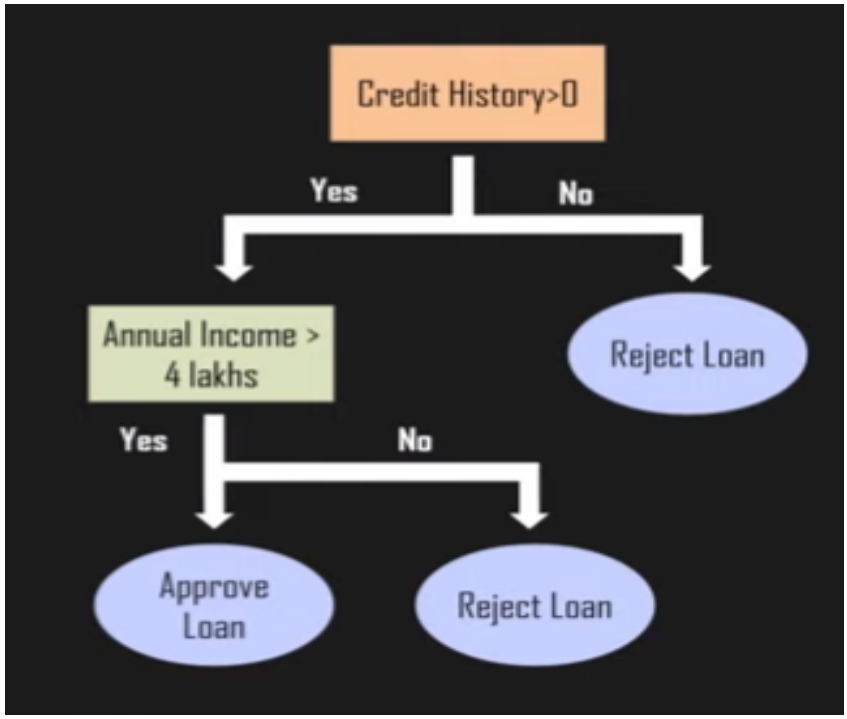
- Decision Tree Classifier
- Naive Bayes Classifier

We have choosen the best out of the two based on the accuracy obtained. And the same algorithm was used to predict the loan status in our test dataset.

1. Decision Tree Classifier

Step 1: Pick best attribute from dataset
Step 2: Ask a relavent question
Step 3:Divide the dataset based on answer
Step 4: Goto step 1 and recursively make new decision trees using subsets of dataset created

Attribute selection measures: Entropy, information gain, gini index, gain ratio



Function: DecisionTreeClassifier( ) from sklearn.tree

Accuracy: 0.6910569105691057

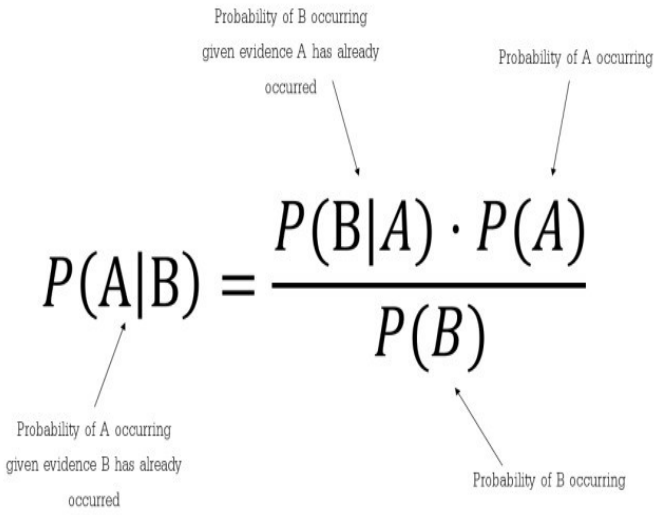
2. Naive Bayes Classifier

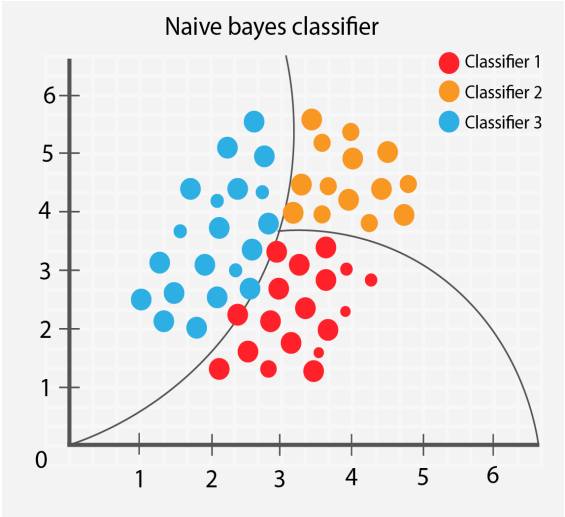
We have used this simple probabilistic classifier based on Bayes Theorem. Its a fast, accurate and reliable classification algorithm with strong naive (independent) assumptions between features

Assumption made:
Each feature makes an

- Independent
- Equal

contribution to the outcome





Function: GuassianNB( ) from sklearn.naive_bayes

Accuracy: 0.8292682926829268

ALGORITHM	ACCURACY
Decision Tree	0.69
Naive Bayes	0.83 ✔

On Comparing both the models, the accuracy of Naive Bayes Algorithm is better than Decision Tree Classifier.
Hence we have used Naive Bayes Classification for predicting the target variable in our test data set.
All the above methods such as EDA, Handling missing data and outliers, preprocessing and model building was used to predict the eligibility of loan for a given applicant.

CONCLUSION

In our project, Loan Eligibility Prediction, we have classified a given applicant into whether they are eligible or not eligible for a loan based on the information from their loan applications. We have carefully and systematically analysed every feature in the dataset and have built the models using Decision Tree and Naive Bayes Algorithms. We have also handled the missing values and outliers and have used only the normalized data in building the models. The Loan Status (if eligible or not) for the test dataset was predicted using Naive Bayes Algorithm since it gave higher accuracy when compared to Decision Tree Classifier. Our project would be very useful for banks and various other finance companies to automate the tedious process of background checking and validation.