EXPLORATORY DATA ANALYSIS AND PREDICTIVE MODEL BUILDING OF MS ADMISSION DATASET USING MACHINE LEARNING: LOGISTIC REGRESSION, DECISION TREE, RANDOM FOREST, SUPPORT VECTOR MACHINE, NAÏVE BAYES CLASSIFIER.

by

HRITTIK BANERJEE

[Reg. No.: 213001818010054, Roll No.: 30018021054]

SOUMITRO MUKHERJEE

[Reg. No.: 213001818010030, Roll No.: 30018021030]

BISHAL CHAKRABORTY

[Reg. No.: 213001818010027, Roll No.: 30018021027]

Department of Applied Statistics

School of Applied Science and Technology

ACKNOWLEDGMENT

We would like to express our profound and deep sense to our guide **prof. (Dr.) Indrani Mukherjee** and **prof. Debjit Konai** for their unending help, guidance, and suggestions without which this project would not have been a reality. They have acted as our philosopher, and guide. We owe great indebtedness for his untiring effort throughout the period of our research work.

We express our sincere thanks to the **prof. Prashanta Narayan Dutta** for his help in every corner of our study and we are gratefully indebted to **prof. (Dr.) Sukhendu Samajdar**, our respected director sir for his great inspiration and encouragement for our work.

We thank our teammate, **Mr. Soumitro Mukherjee**, for helping throughout the coding part and sparing his valuable time for project discussion and code writing.

We especially thank our teammate, **Mr. Bishal Chakraborty**, for sparing his valuable time for our academic project discussion.

Lastly, we thank all those concerned persons who have been directly or indirectly responsible for the completion of our project.

1 INTRODUCTION:

As we are studying MS, we know the amount of pressure someone has to go through during MS admissions, there are different requirements for the admission process and after meeting them all one will get a chance into the MS course of a certain University and In this project, we have taken the dataset of graduate admissions and our goal is to find out some important features from this dataset through Explanatory Data Analysis (EDA) and then we applied machine learning classifiers on this data to observe which model best fits this data. We've applied five ML classifiers namely, Logistic Regression, Decision Tree, Random Forest, Support Vector (SVC), and Naive Bayes. Before moving to the next part of our project let's know more about the dataset.

1.1 DATASET DISCUSSION:

	Serial No.	GRE Score	TOEFL Score	University Rating	SOP	LOR	CGPA	Research	Admission Status
0	1	337	118	4	4.5	4.5	9.65	1	1
1	2	324	107	4	4.0	4.5	8.87	1	1
2	3	316	104	3	3.0	3.5	8.00	1	0
3	4	322	110	3	3.5	2.5	8.67	1	1
4	5	314	103	2	2.0	3.0	8.21	0	0

In the above picture, we are able to see a glimpse of the data set, in which there are 9 columns,

- The first one is for serial no.
- 2nd is for Graduate Record Examinations (GRE) scores ranging between 290 to 340.
- 3rd one is for the Test of English as a Foreign Language (TOEFL) scores ranging between 92 to 120.
- 4th column represents the university rating 1 to 5.
- 5th column represents the Statement of Purpose (SOP) ranging between 1 to 5.
- 6th column represents the Letter of Recommendation (LOR) strength ranging between 1 to 5.
- 7th column represents the CGPA score of the graduate students.
- 8th column represents the Research status of the students if they have researched on any topic then they have been assigned the value 1 otherwise 0 value is assigned.
- 9th column represents Admission status i.e. if the candidate gets admitted to the respective university, then he is assigned the value 1 otherwise 0 value is assigned to that row.

2 METHODOLOGIES:

Here, at the beginning of our data analysis, we have done **EDA (tabular and graphical representation)** by that we observed the columns and the data in a graphical and more precise way then we moved toward the ML classifiers. A big part of machine learning is classification — we want to know what class (a.k.a. group) an observation belongs to. The ability to precisely classify observations is extremely valuable for various applications like predicting whether a particular incident will occure or not.

I. ***MIN-MAX STANDARDIZATION:*** We have used this scalarization technique for our data.
The mathematical formulation,

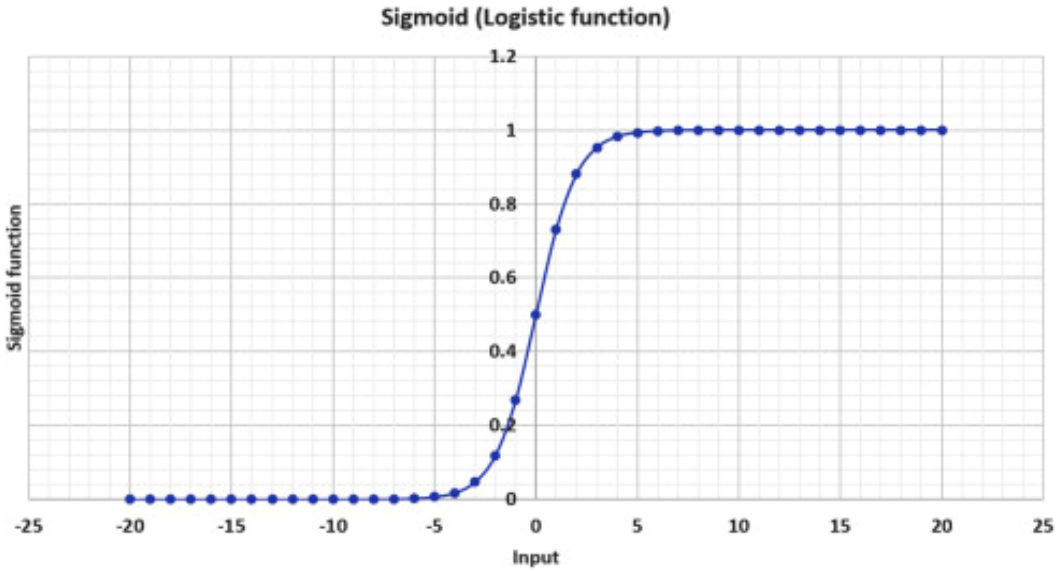
$$x_{scaled} = \frac{x - min\,(x)}{max(x) - min\,(x)}$$

One important thing to keep in mind when using the MinMax Scaling is that it is highly influenced by the maximum and minimum values in our data so if our data contains outliers it is going to be biased. MinMaxScaler rescales the data set such that all feature values are in the range [0, 10]. This is done feature-wise in an independent way. The MinMaxScaler scaling might compress all inliers in a narrow range.

II. ***LOGISTIC REGRESSION:*** Logistic regression, despite its name, is a classification model rather than a regression model. Logistic regression is a simple and more efficient method for binary and linear classification problems. It is a classification model, which is very easy to realize and achieves very good performance with linearly separable classes. It is an extensively employed algorithm for classification in industry. The logistic regression model, like the Adaline and perceptron, is a statistical method for binary classification that can be generalized to multiclass classification. Scikit-learn has a highly optimized version of logistic regression implementation, which supports multiclass classification tasks *(Raschka, 2015)*. Logistic regression is a process of modeling the probability of a discrete outcome given an input variable. The most common logistic regression models a binary outcome; something that can take two values such as true/false, yes/no, and so on. Multinomial logistic regression can model scenarios where there are more than two possible discrete outcomes. Logistic regression is a useful analysis method for classification problems, where you are trying to determine if a new sample fits best into a category. As aspects of cyber security are classification problems, such as attack detection, logistic regression is a useful analytic technique. Logistic regression is another powerful supervised ML algorithm used for binary classification problems (when target is categorical). The best way to think about logistic regression is that it is a linear regression but for classification problems. Logistic regression essentially uses a logistic function defined below to model a binary output variable (Tolles & Meurer, 2016). The primary difference between linear regression and logistic regression is that logistic regression's range is bounded between 0 and 1. In addition, as opposed to linear regression, logistic regression does not require a linear relationship between inputs and output variables. This is due to applying a nonlinear log transformation to the odds ratio.

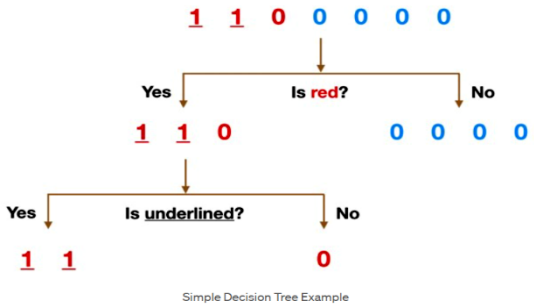
$$\boldsymbol{Logistic\ Function} = \frac{1}{1 + e^{-x}}$$

In the logistic function equation, x is the input variable. Let's feed in values -20 to 20 into the logistic function. As illustrated in Fig. 5.17, the inputs have been transferred to between 0 and 1.



III. ***DECISION TREE:*** Decision Trees (DTs) are a non-parametric supervised learning method used for classification and regression. The goal is to create a model that predicts the value of a target variable by learning simple decision rules inferred from the data features. A tree can be seen as a piecewise constant approximation. For instance, in the example below, decision trees learn from data to approximate a sine curve with a set of if-then-else decision rules. The deeper the tree, the more complex the decision rules and the fitter the model.

Construction of Decision Tree: A tree can be "learned" by splitting the source set into subsets based on an attribute value test. This process is repeated on each derived subset in a recursive manner called recursive partitioning. The recursion is completed when the subset at a node all has the same value of the target variable, or when splitting no longer adds value to the predictions. The construction of decision tree classifier does not require any domain knowledge or parameter setting, and therefore is appropriate for exploratory knowledge discovery. Decision trees can handle high dimensional data. In general decision tree classifier has good accuracy. Decision tree induction is a typical inductive approach to learn knowledge on classification.

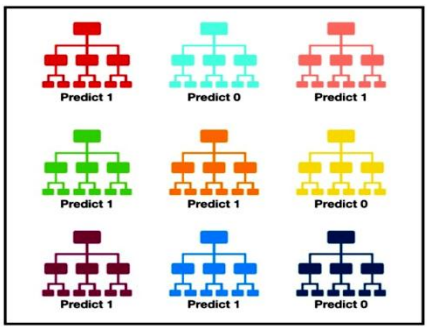


Simple Decision Tree Example

Decision Tree Representation: Decision trees classify instances by sorting them down the tree from the root to some leaf node, which provides the classification of the instance. An instance is classified by starting at the root node of the tree, testing the attribute specified by this node, then moving down the tree branch corresponding to the value of the attribute as shown in the above figure. This process is then repeated for the subtree rooted at the new node.

The decision tree in above figure classifies a particular morning according to whether it is suitable for playing tennis and returning the classification associated with the particular leaf. (In this case Yes or No).

IV. ***THE RANDOM FOREST CLASSIFIER:*** Random Forest, like its name implies, consists of a large number of individual decision trees that operate as an ensemble. Each individual tree in the random forest spits out a class prediction and the class with the most votes becomes our model's prediction (see figure rhs).



Visualization of a Random Forest Model Making a Prediction

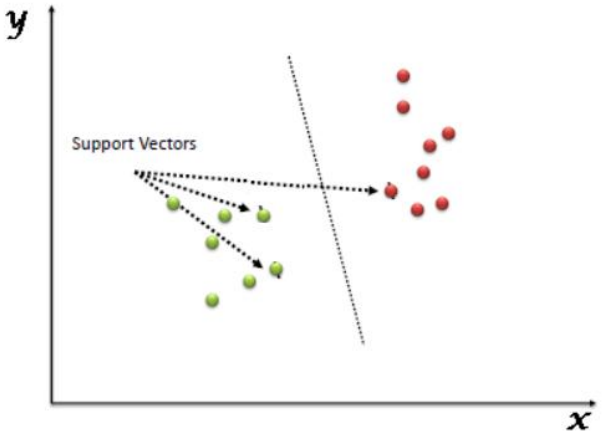
The fundamental concept behind random forest is a simple but powerful one — the wisdom of crowds. In data science speak, the reason that the random forest model works so well is:

A large number of relatively uncorrelated models (trees) operating as a committee will outperform any of the individual constituent models.

The low correlation between models is the key. Just like how investments with low correlations (like stocks and bonds) come together to form a portfolio that is greater than the sum of its parts, uncorrelated models can produce ensemble predictions that are more accurate than any of the individual predictions. The reason for this wonderful effect is that the trees protect each other from their individual errors (as long as they don't constantly all err in the same direction). While some trees may be wrong, many other trees will be right, so as a group the trees are able to move in the correct direction. So the prerequisites for random forest to perform well are:

1. There needs to be some actual signal in our features so that models built using those features do better than random guessing.
2. The predictions (and therefore the errors) made by the individual trees need to have low correlations with each other.

V. ***SUPPORT VECTOR MACHINE:*** "Support Vector Machine" (SVM) is a supervised machine learning algorithm that can be used for both classification or regression challenges. However, it is mostly used in classification problems. In the SVM algorithm, we plot each data item as a point in n-dimensional space (where n is a number of features you have) with the value of each feature being the value of a particular coordinate. Then, we perform classification by finding the hyper-plane that differentiates the two classes very well (look at the snapshot).



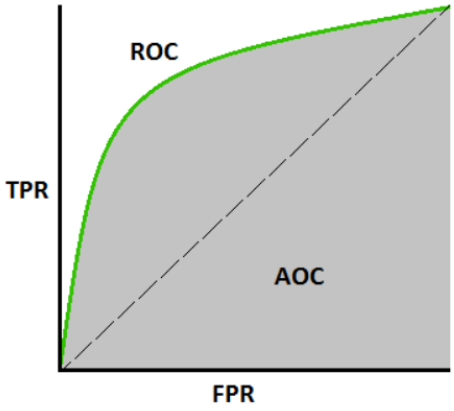
SVMs can be used to solve various real-world problems:

1. SVMs are helpful in text and hypertext categorization, as their application can significantly reduce the need for labeled training instances in both the standard inductive and transductive settings. Some methods for shallow semantic parsing are based on support vector machines.
2. Classification of images can also be performed using SVMs. Experimental results

show that SVMs achieve significantly higher search accuracy than traditional query refinement schemes after just three to four rounds of relevance feedback. This is also true for image segmentation systems, including those using a modified version SVM that uses the privileged approach as suggested by Vapnik.
3. Classification of satellite data like SAR data using supervised SVM.
4. Hand-written characters can be recognized using SVM.
5. The SVM algorithm has been widely applied in the biological and other sciences. They have been used to classify proteins with up to 90% of the compounds classified correctly. Permutation tests based on SVM weights have been suggested as a mechanism for interpretation of SVM models.[15][16] Support-vector machine weights have also been used to interpret SVM models in the past.[17] Posthoc interpretation of support-vector machine models in order to identify features used by the model to make predictions is a relatively new area of research with special significance in the biological sciences.

VI. *NAÏVE BAYES CLASSIFIER:*

Naive Bayes is a simple technique for constructing classifiers: models that assign class labels to problem instances, represented as vectors of feature values, where the class labels are drawn from some finite set. There is not a single algorithm for training such classifiers, but a family of algorithms based on a common principle: all naive Bayes classifiers assume that the value of a particular feature is independent of the value of any other feature, given the class variable. For example, a fruit may be considered to be an apple if it is red, round, and about 10 cm in diameter. A naive Bayes classifier considers each of these features to contribute independently to the probability that this fruit is an apple, regardless of any possible correlations between the color, roundness, and diameter features.
In many practical applications, parameter estimation for naive Bayes models uses the method of maximum likelihood; in other words, one can work with the naive Bayes model without accepting Bayesian probability or using any Bayesian methods.
Despite their naive design and apparently oversimplified assumptions, naive Bayes classifiers have worked quite well in many complex real-world situations. In 2004, an analysis of the Bayesian classification problem showed that there are sound theoretical reasons for the apparently implausible efficacy of naive Bayes classifiers. Still, a comprehensive comparison with other classification algorithms in 2006 showed that Bayes classification is outperformed by other approaches, such as boosted trees or random forests. An advantage of naive Bayes is that it only requires a small number of training data to estimate the parameters necessary for classification.



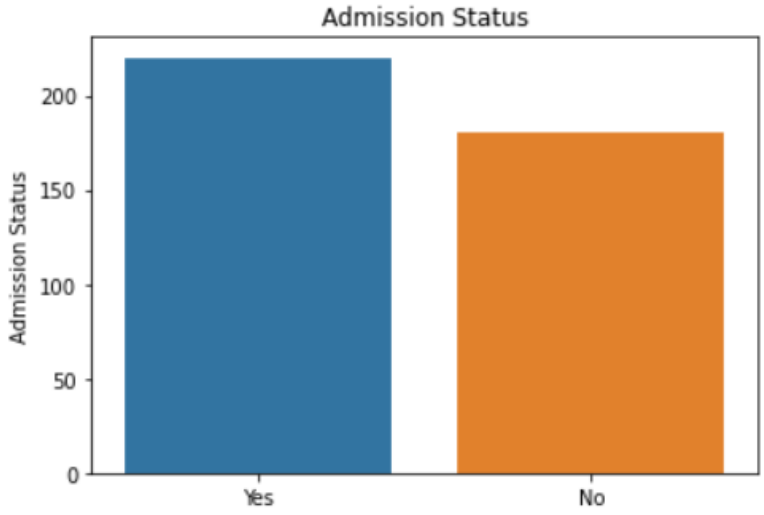
AUC - ROC Curve [Image 2] (Image courtesy: My Photoshopped Collection)

VII. ***ROC-AUC CURVE:*** In Machine Learning, performance measurement is an essential task. So when it comes to a classification problem, we can count on an AUC - ROC Curve. When we need to check or visualize the performance of the multi-class classification problem, we use the AUC (Area Under The Curve) ROC (Receiver Operating Characteristics) curve. It is one of the most important evaluation metrics for checking any classification model's performance. It is also written as AUROC (Area Under the Receiver Operating Characteristics)

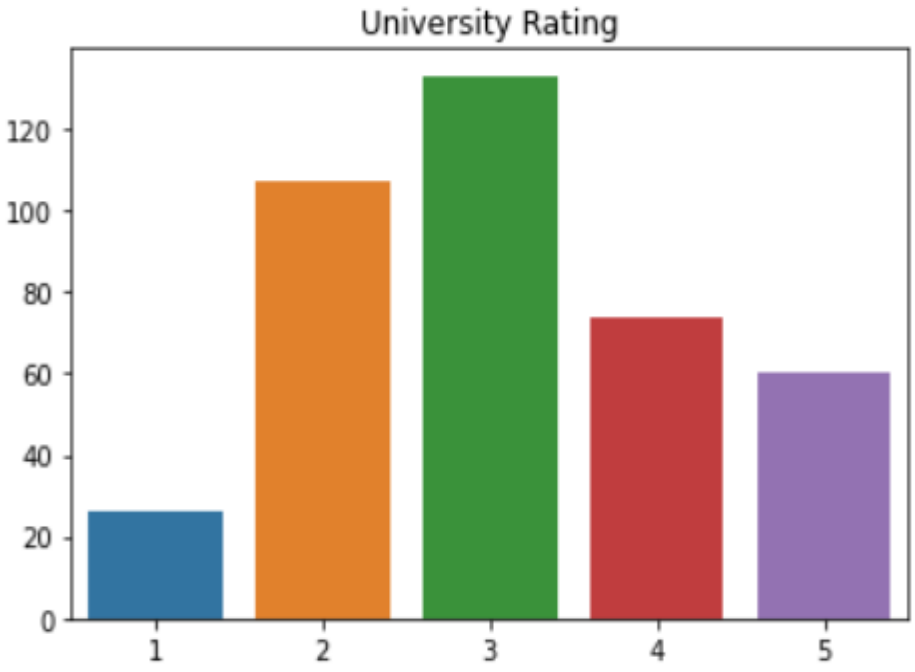
AUC - ROC curve is a performance measurement for the classification problems at various threshold settings. ROC is a probability curve and AUC represents the degree or measure of separability. It tells how much the model is capable of distinguishing between classes. Higher the AUC, the better the model is at predicting 0 classes as 0 and 1 classes as 1. By analogy, the Higher the AUC, the better the model is at distinguishing between patients with the disease and no disease. The ROC curve is plotted with TPR against the FPR where TPR is on the y-axis and FPR is on the x-axis.

3 DATA ANALYSIS:

First, we are interested in the column Admission Status, so we are seeing the bar plot of this column.

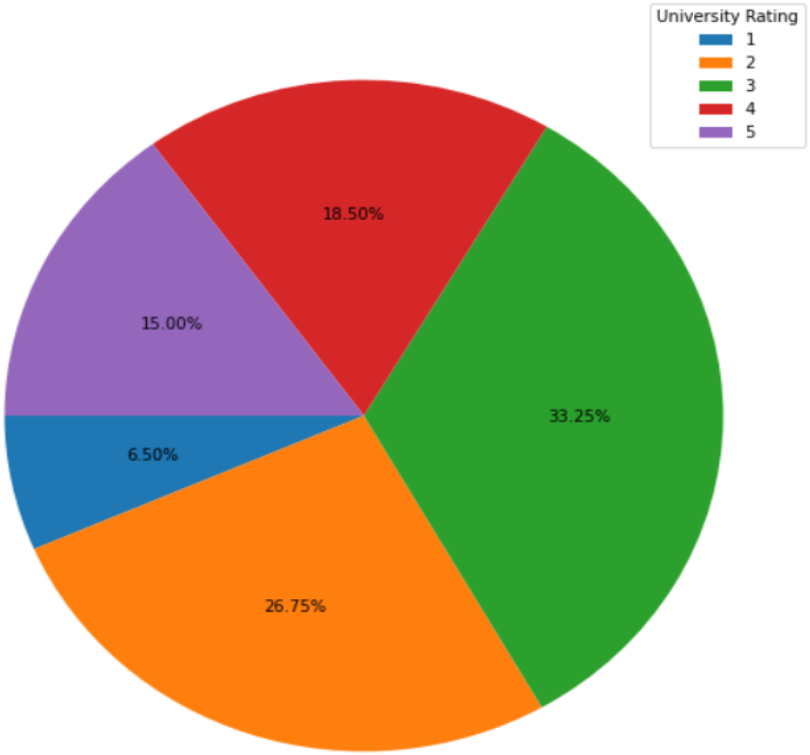


Bar plot of University Rating.

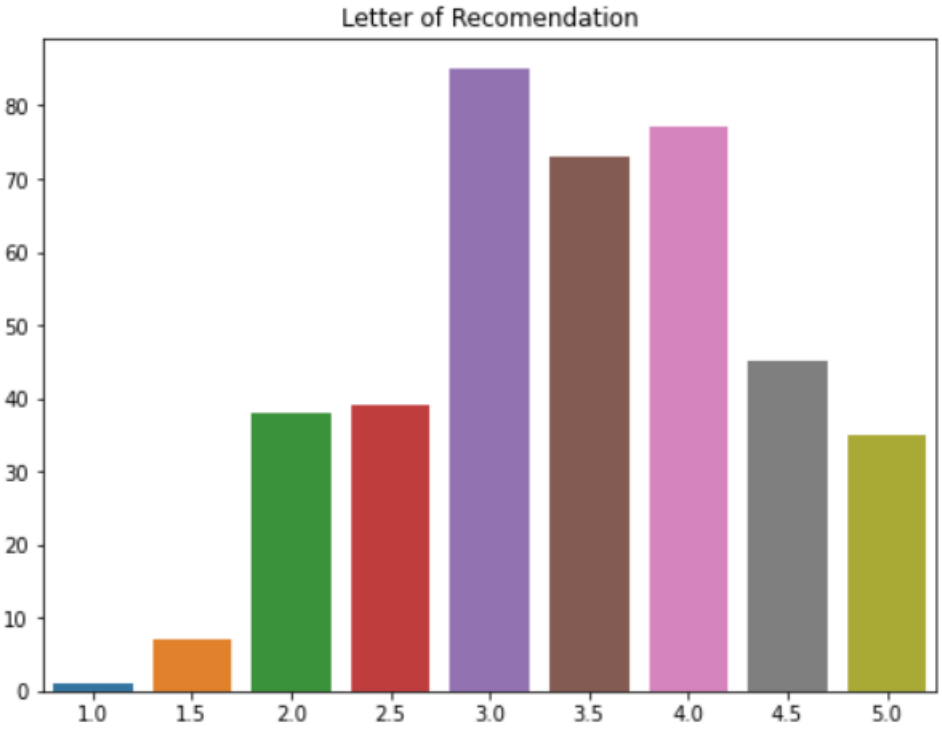


Here, we are observing that 3 has the most frequency.

This is the pie chart of University Rating which shows 2 and 3 has most frequency.

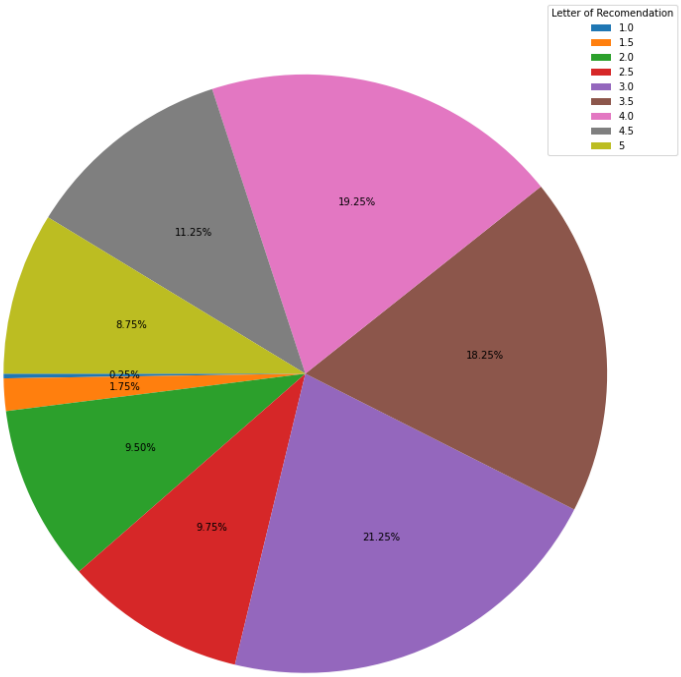


Now, providing the bar plot of Letter of Recommendation.



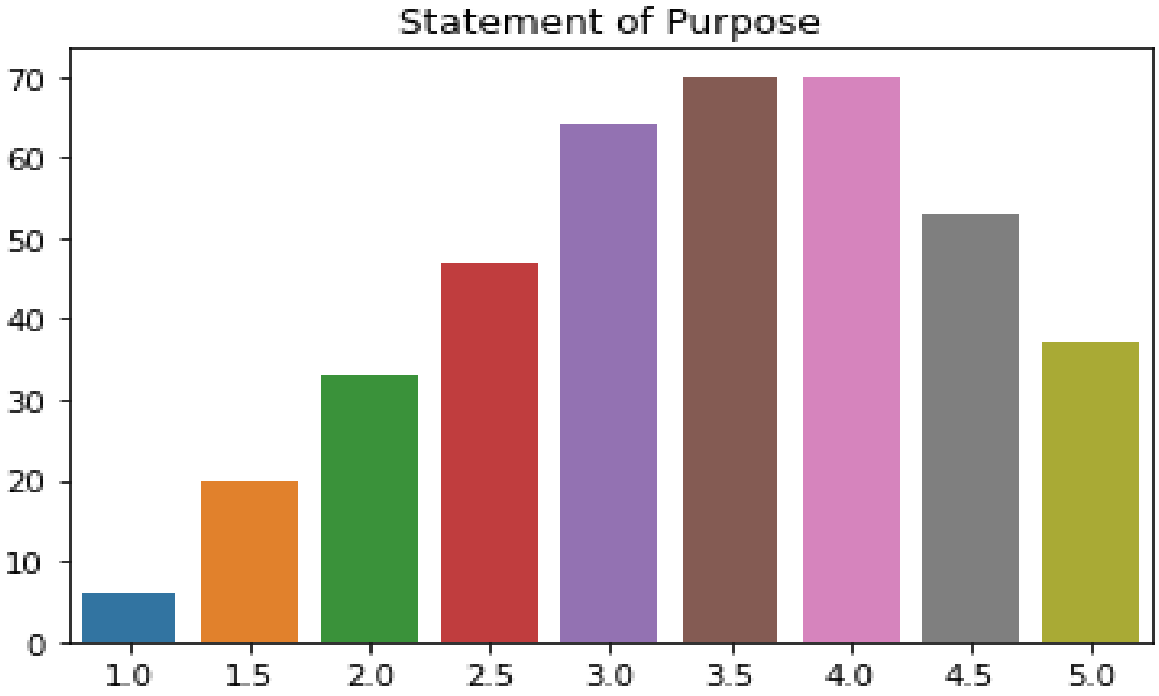
most of the observation lie in the region 3 to 4.

Pie chart of Letter of Recommendation.



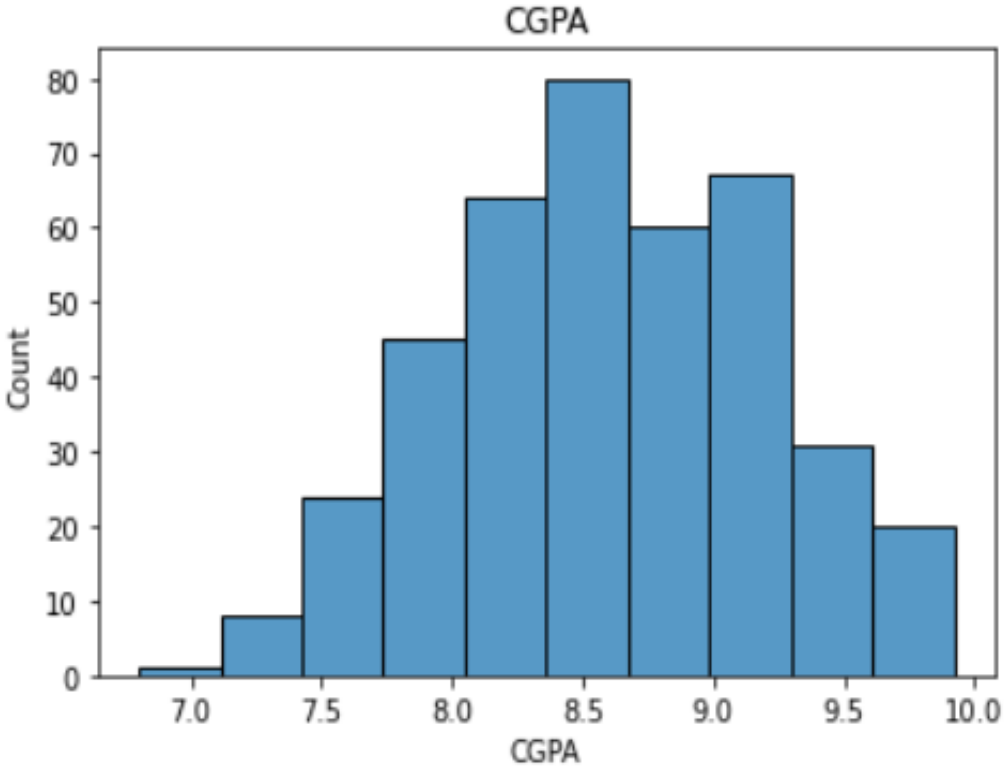
Here we can observe 3 has the highest frequency.

Bar plot of Statement of Purpose.



Here we observe, most of the observations lie between 2.5 to 4.5.

Histogram of CGPA.

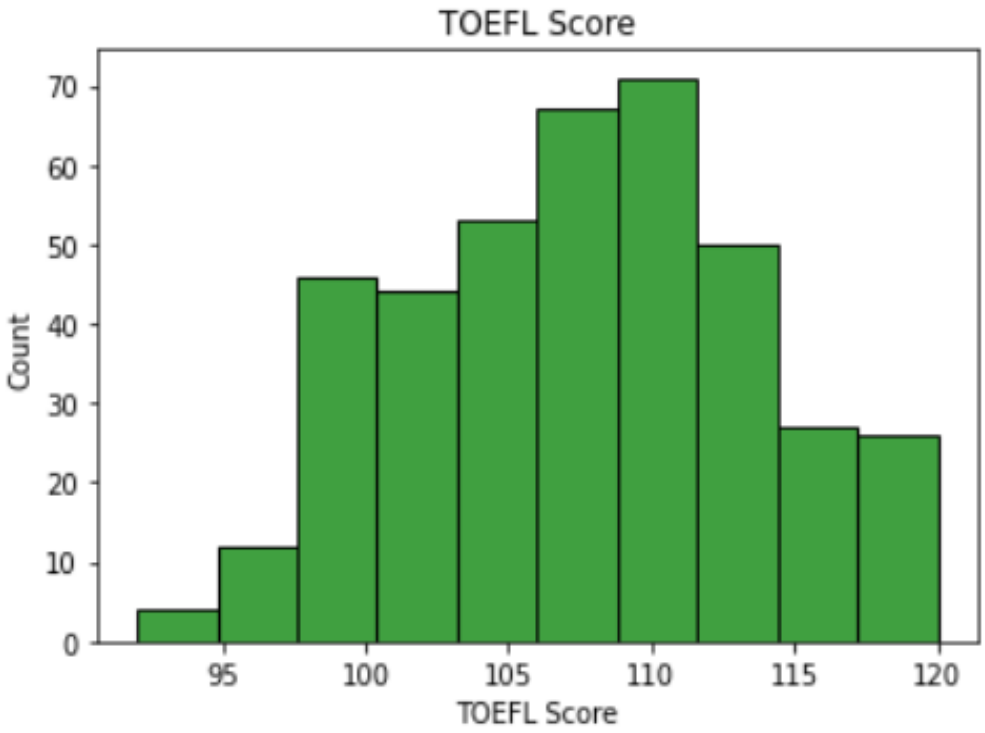


Here we can observe that 8.5 has the most frequency.

Histogram of GRE.

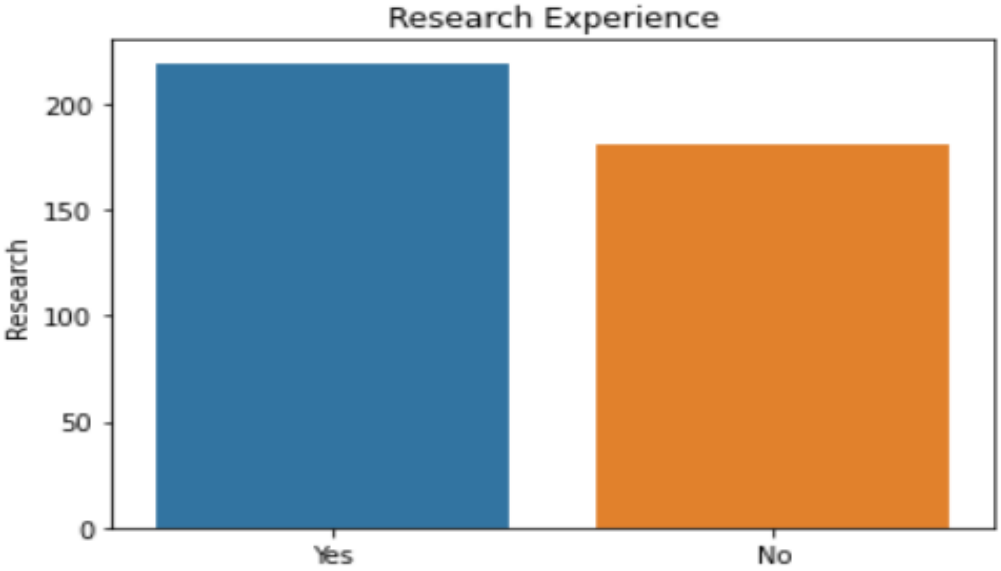
It can be observed that the range 310 to 330 has the highest no. of observation.

Histogram of TOEFL.



110 has the highest frequency.

Bar plot Research Experience.



Max no of observation has research Experience(approx.220).

From this table we can clearly observe that admission chance depends upon **CGPA**, **GRE Score**, **TOEFL Score**.

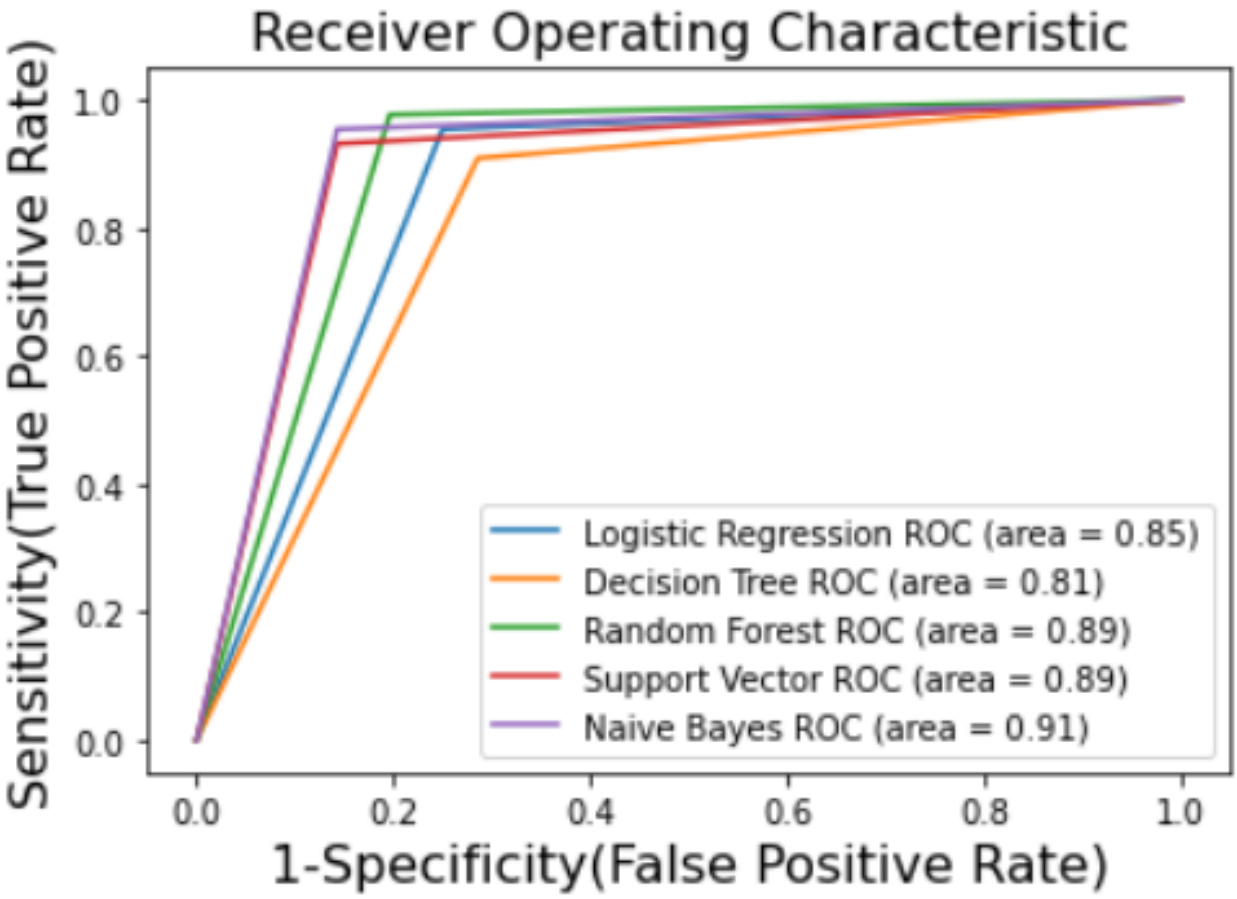
Correlation of Admission Status with	**Correlation Coefficient Value**
GRE Score	0.686138
TOEFL Score	0.672465
University Rating	0.638983
SOP	0.612152
LOR	0.557481
CGPA	0.737307
Research	0.519441

Now, we are going to fit different ML classifier to this dataset.
We have standardized the data using **Min-Max** Scalarization.

ML Classifiers	**Accuracy**	**Precision**	**Recall**	**F1-Score**	**ROC-AUC Score**
Logistic Regression	0.84	0.75	0.954545	0.84	0.852272
Decision Tree	0.8	0.71428	0.9090	0.8	0.81168
Random Forest	0.88	0.796296	0.977272	0.8775510	0.89042
Support Vector	0.89	0.836734	0.931818	0.88172	0.89448
Naïve Bayes	0.9	0.84	0.954545	0.863617	0.905844

Now, we are using **ROC-AUC** Score to decide the best model fitted in the data. Now from the above table we observed that **Naïve – Bayes** Classifier has the *maximum score* so Naïve-Bayes Classifier is the best model for this data.

The ROC-AUC curve is shown below:



4 CONCLUSION:

After doing the EDA and fitting different ML classifiers to this dataset we have observed the data admission chance depends upon CGPA, GRE Score, TOEFL Score. After that we noticed The Naïve Bayes Classifier has the highest ROC-AUC score (0.91) i.e., it covers the maximum area under the curve we can say that this classifier fits the data best.

5 REFERENCES:

- https://www.geeksforgeeks.org/decision-tree/
- https://www.sciencedirect.com/topics/computer-science/logistic-regression
- https://towardsdatascience.com/understanding-random-forest-58381e0602d2#:~:text=The%20random%20forest%20is%20a,that%20of%20any%20individual%20tree.
- https://en.wikipedia.org/wiki/Support-vector_machine
- https://en.wikipedia.org/wiki/Naive_Bayes_classifier
- https://towardsdatascience.com/everything-you-need-to-know-about-min-max-normalization-in-python-b79592732b79
- https://towardsdatascience.com/understanding-auc-roc-curve-68b2303cc9c5