Admission Forecasting

**Data sets:** Two data sets to work with,

Primary data set: Complete admission data, used to construct the model.

Second data set: Contains predictors only, used to make predictions for admission.

**Process Flow in RStudio:**

We’ll start by loading the necessary libraries for our analysis along with setting the working directory for our files.

A white background with green text

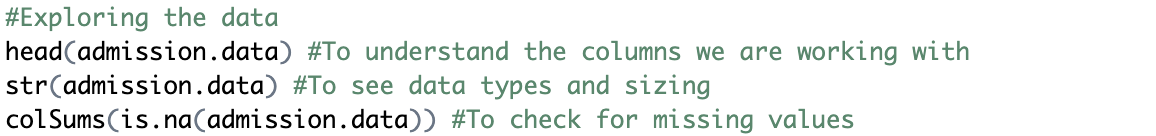
Description automatically generated

Loading the dataset into a variable which represents the type of data we are dealing with is usually ideal. In our scenario the data corresponds to admission, as shown by the chosen name.



Now we will explore the data in hope of gaining a better understanding of

1. Headers,
2. Data types,
3. Range of values,
4. Missing values if any



We have four variables, three predictors and one response variable,

A white background with blue text

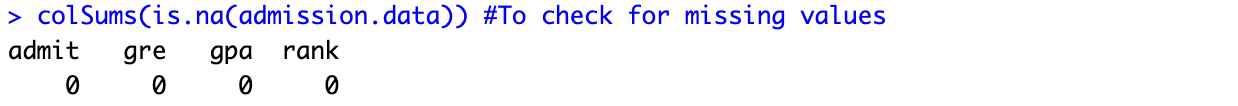
Description automatically generated

We have 400 objects, three integer type and one numeric data, which seems the right fit so we do not need any Intervention,

A close-up of a number

Description automatically generated

We also confirmed that there are no missing values in our data set,



A close-up of a math equation

Description automatically generated

This plot helps us visualize the amount of overlap between GRE scores for admission as well as the distribution of the scores. Also, only couple outliers can be observed.

A graph with blue squares

Description automatically generated

A white background with black and green text

Description automatically generated

This plot shows the GPA across admission status, one observable outlier.

A graph showing a green box

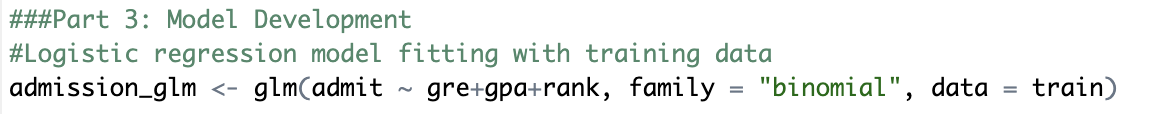
Description automatically generated with medium confidence

After understanding the data, we will now split the data and also seed it for reproducibility,

A computer code with text

Description automatically generated

As we are ready with our data set for modeling, we will construct the equation. We can observe from data that admission status is a result of a combination between the other three variables. So, “admit” becomes our response variable and the remaining three our predictors. We will be using training data set for constructing the model,



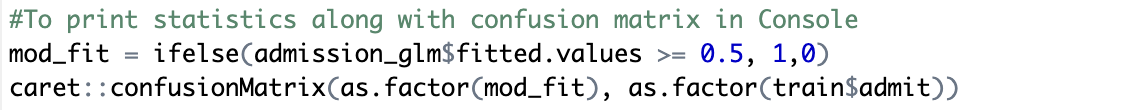


From the significance codes we can see that rank is the only significant variable influencing admission. Also, by looking at the accompanying negative sign for the coefficient of rank variable, we can say that its Inversely related to our response. That means higher the rank the closer the possibility for rejected admission. (as it nears 0)

A screenshot of a computer

Description automatically generated

We are defining 50% as our decision mark and will look at the performance of the model with training data by the use of Confusion matrix and statistics,



We can see that accuracy of the model is 69.64%, with True Positives being dominant (also shown by 0.9022 Sensitivity value). We could see a P value as 0.09 (>0.05), meaning that it’s not completely significant. We can say that the model predicts rejected admissions well but needs improvement in detecting successful admits.

A screenshot of a computer

Description automatically generated

For Visualization purposes, let’s look at the Confusion matrix clearly.

A close-up of a computer code

Description automatically generated

A yellow and blue squares with black numbers

Description automatically generated

We will now use test data to validate our model and evaluate the performance in similar fashion,

A screenshot of a computer program

Description automatically generated

From the results we can see that Accuracy and Sensitivity seem to be maintaining a similar level,

A screenshot of a computer

Description automatically generated

Confusion matrix for the test data predictions,

A yellow and blue squares

Description automatically generated

Now as we have evaluated the accuracy and statistics of our constructed model, we will make predictions for the new data set using earlier built model. Firstly, we’ll load the data set and explore the data. In our case, we had to explicitly mention the data type for GPA to be numeric due to an error caused while reading the data.

A close-up of a number

Description automatically generated

Str () function to see the data structure of the dataset, and we see that our conversion has been successful.

A close-up of a white background

Description automatically generated

As the new data is prepared, we will fit that into our model and get the predictions stored into a column and add it into the new data set,

A close-up of a computer screen

Description automatically generated

This is the head of the new data set along with predictions.

A white background with black and white clouds

Description automatically generated

From our Initial Boxplots, we can observe that having a GRE score around 300 makes it difficult to get admission. GPA plot showed the median for success at 3.55, two of the candidates are rejected. Now as Rank plays a significant role, we can most certainly say that the model performed well.

Green Box: R code, Blue Box: Output

Reference and Credits: This Project has been done during my Coursework, so pardon the comments, mentions or references which do not seem relevant to the functionality of the code itself.