REPORT FOR THE ADMISSION PROJECT

I. Introduction and Description of the Admission Project:

The Admission Project aims to predict the response of an accepted applicant to the University offer using a Machine Learning classification model. The selection of such model is based on a multitude of both linear and non-linear methods as well as the various components of an applicant's profile. As a result, it does not only serve as a predictive model for applicants' decision but also as a tool to extract insights from the gathered data, which, in turn, will assist the University in increasing yield. The application of Machine Learning in Admissions is very similar to that of data-driven technology in predicting consumers' behavior, with the students being the customers and University education being the product. In order to make the best deal out of this product, the University needs to balance between the use of its resources whilst attracting the most eligible candidates, thus the importance of understanding the motivation behind applicants' decision.

II. Description of the Original Dataset:

The data set provided by the Office of Admissions covers multiple facets of a student that need to be taken into consideration for admission, which are reflective of their circumstances and personal needs. It has a total of 15,143 rows and 69 columns, with the first 10,000 rows belonging to the *train* data set and the remaining 5143 rows making up the *test* data set. The data has been amassed from all accepted applicants over five consecutive Fall semesters from 2017 to 2021.

Out of the 69 columns, 66 of which are predictor variables, or objective factors that drive the decision of applicants. The other 3 columns are ID number, train.test, and Decision – ID represents the index number of each observation, train.test identifies the grouping of observations, and Decision is the response variable. Of the 66 explanatory variables, 36 are of numeric type whilst the remaining 30 are of categorical type (the data type classification of each variable is subject to change during the data cleaning process).

III. Discussion on the Cleaning Process:

I started off the cleaning process by creating a table named Gen_Sum, which stands for General Summary, to summarize the data type, the number and the percentage of missing data for each variable in

the dataset. I then proceeded to clean the data on a variable-by-variable basis using a variety of steps as follows:

1. Handling missing values:

I employed different approaches to handle missing values based on data types and the amount of meaningful data available in each column. Columns with over 95% of the values as NAs, such as Sport.2.Sport and School.2.Class.Rank..Numeric., are removed because they are unlikely to generate insights during the modeling process. For other variables, missing values are either imputed using related and relevant columns if they are available, or simply classified as unspecified if there is no data to infer from.

2. Correcting typos or measurement errors:

All variables have no typos and spelling errors with the only exception being the Merit.Award column. In this variable, a Merit Award encoded as TT12.5 was mistakenly entered as TT125. I replaced all observations with value "TT125" in the data set with the correct level "TT12.5" to avoid confusion and further errors in interpretation.

3. Dealing with outliers:

I identified outliers by plotting the distribution of variables to see points that have abnormal distance from the rest of the data. The distribution of numeric variables is examined by using the boxplot or the density plot, whereas for the categorical variables I used the bar plot.

4. Grouping levels:

Variables with a large number of levels are overly specific and detailed, leading some of the levels to having a small number of observations. As a result, they are unable to explain for the variability in the outcome variable and fail to contribute to the classification models. Grouping similar or related levels allows for less confusion as well as higher interpretability.

5. Data type conversion:

All numeric variables with integer data type are converted to numeric data and all categorical variables with character data type are converted to factor (ordered or unordered) data. With numeric variables whose range of values is divided into intervals (which are used to replace exact data points), I converted them to ordered factors because their quantitative ranges represent qualitative levels.

6. Transformations:

Transformations are applicable to only numeric variables and are determined by the distribution of the data as well as its skewness. Data with high skewness (skewness > 1 or skewness < -1) are transformed with an appropriate transformation whilst data with moderate and slight skewness are maintained.

7. Creating new variables:

New variables are created as combinations of related existing variables in an attempt to better account for the response variables.

8. Removing variables:

Variables with a value or a level making up at least 95% of the data are removed as they will not provide meaningful insights in predicting Decision.

IV. Comparison of models – Best model selection:

Model	Kappa - Test
Logistic Regression	0.5371796
KNN	0.3117081
Simple Tree	0.2008635
Pruned Tree	0.2008635
Bagging	0.4919075
Random Forest	0.5252375
Boosting	0.540082

Based on Kappa score and interpretability, the best model in my opinion is Logistic Regression. Although Boosting results in the highest Kappa score with cut-off point 0.42, Logistic Regression allows for better interpretation as the summary of the model lists out the weight (coefficient) assigned to each predictor as well as their statistical significance (p-value). All remaining models have a lower Kappa score than Logistic Regression with the figure for Random Forest only slightly lower than that of Logistic Regression whilst the numbers for KNN, Simple Tree, and Pruned Tree are far below Logistics Regression's Kappa score. As a result, Logistic Regression is the best model among all classification models built.

V. Final Conclusion – Personal Reflection:

1. What I learned:

The biggest takeaway for me from this project is the ability to make sensible decisions during the data cleaning process. Such ability does not only require me to understand the nature of the data set I'm working with and the information they convey, but also to be able to justify for the decisions I make. I also learned that striking a balance between the amount of cleaning and the usefulness of the insights extracted from the data is extreme – if the cleaned data do not increase the predictive and explanatory value of the model, reconsideration of cleaning methods is crucial.

2. Obstacles I have encountered and my solutions:

a. Missing level in *test* dataset: When I ran the Logistic Regression model after the initial cleaning, R returned an error message because many levels of the Permanent.Country variable were absent from the *test* data set. As a result, I grouped the countries in Permanent.Country by continents and eventually removed the variable because one of the levels makes up over 95% of the data.

b. Errors with the kNN model: I received error messages for the kNN models due to the fact that the parameters I passed to the function were not of the correct data structure. Converting the parameters from *list* to *data frame* helped to run the code smoothly.

3. Overall reflection on the experience:

Overall, this project has been a learning opportunity for me not only in terms of Machine Learning techniques but also in terms of the mindset and sensibility necessary for people working in the analytics field. It has been a truly unique experience for me to make decisions not following any rules other than Dr. Zhu's guidance and my personal judgment, which gave me an enormous sense of responsibility.