Instructor: Abhimanyu Gupta

Task: Assignment 1

Student: Adam Behun

Titanic – Machine Learning from Disaster

My objective was to predict which passengers of the Titanic Disaster were likely to survive and which were not, based on available information about the passengers. To do this, I was provided with two datasets: train.csv and test.csv. The train.csv dataset contained 891 rows (passengers) and their information, including the "ground truth" about whether they survived the accident. On the other hand, the test.csv dataset had similar information but did not reveal the "ground truth" (survival rate) for each passenger. My task was to predict the likelihood of survival of the 418 passengers in the test.csv dataset.

To achieve this objective, I used supervised learning machine learning techniques to identify patterns in the train.csv dataset. After that I was able to plug in a new dataset and predict the values for ‘Survived’ column using different algorithms. The project was divided into four parts: Exploratory Data Analysis, Feature Engineering, Model Selection, and Results Interpretation.

I started the project by importing both datasets into my local machine. The data was in a .csv format and was relatively small, not too complex, and easy to work with. I then started with an exploratory data analysis to understand the train.csv dataset and find patterns that could possibly help me engineer some features to help training my model. I found that the dataset had 891 rows and 12 columns, with missing values in the 'Age', 'Cabin', and 'Embarked' columns. I also found that 62% of all passengers had died in the accident and that there was a clear relationship between the 'Pclass' and 'Survived' variables, with passengers in higher classes being more likely to survive. I discovered that the 'Name' column could be used after extracting titles and that there was a clear relationship between the newly created 'Title' variable and 'Survived'. I also found that passengers with more letters in their 'Name' variable were more likely to survive, which could indicate a higher status. I noted that 65% of the passengers were male and that 74% of female passengers survived while only 19% of male passengers did. I also found that the correlation between 'SibSp', 'Survived', and 'Parch', 'Survived' was not that strong, so I created a 'Fam\_Size' variable to better understand the relationship between family size and survival rate of individuals.

Moving on to the feature engineering part of the project, I dropped columns that provided little predictive value. Namely, these were ‘Ticket’, ‘Cabin’, ‘PassengerId’, and ‘Died’ (was created to better visualize the relationship within sex and survival rate). I also converted 'Sex', 'Embarked', and 'Title' columns into numerical representations. My thinking was that it is going to be easy to work with it once it is a numerical values. I also computed a median for male and female 'Age' values and imputed these values to the missing places in the train dataset. I dropped rows with missing 'Embarked' values and created a new variable 'Fam\_Size' to calculate the family size. Next, I split the 'Name' column, created a new column 'Title', and extracted the titles of different passengers. Doing this I found out that there are 17 different titles for representing various characteristics of the passengers on-board. I have concluded that there were 6 "social classes": 1. Officer, 2. Royalty, 3. Miss, 4. Mrs., 5. Mr., 6. Master and I have matched the previous titles provided (17) to these new titles (6). After I had the ‘Title’ column ready, I dropped the name column which provided little value (most likely). Another technique to improve my model’s accuracy is to normalize or rescale the 'Age' and 'Fare' columns for the values to fall between 0 and 1. After all these steps were done, I had a clean dataset, and I was ready to fit an ML model to it.

I then tried Logistic Regression, Decision Tree, and Random Forest algorithms to solve this problem and predict who is going to survive and who is not. My best scores from Kaggle.com for the different algorithms were the following:

Logistic Regression = 0.787

Decision Tree = 0.775

Random Forest = 0.785

While Logistic Regression was the best model in terms of official Kaggle.com score, I cannot definitively conclude that it is the best model to use with this dataset. I think this because I used slightly different techniques to prepare the dataset for different models, and there is definitely room for improvement in terms of feature engineering.