REPORT OF KAGGLE SUBMISSION

Carnegie Mellon University Africa

Submitted By: Peace Ekundayo Bakare

Course: Data, Inference, and Applied Machine Learning

Date: 6th December, 2024

Libraries Used:

```
import numpy as np # linear algebra
import pandas as pd # data processing, CSV file I/O (e.g. pd.read_csv)
import seaborn as sns
import matplotlib.pyplot as plt
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import classification_report, accuracy_score
from sklearn.preprocessing import LabelEncoder, StandardScaler
from sklearn.model_selection import train_test_split
```

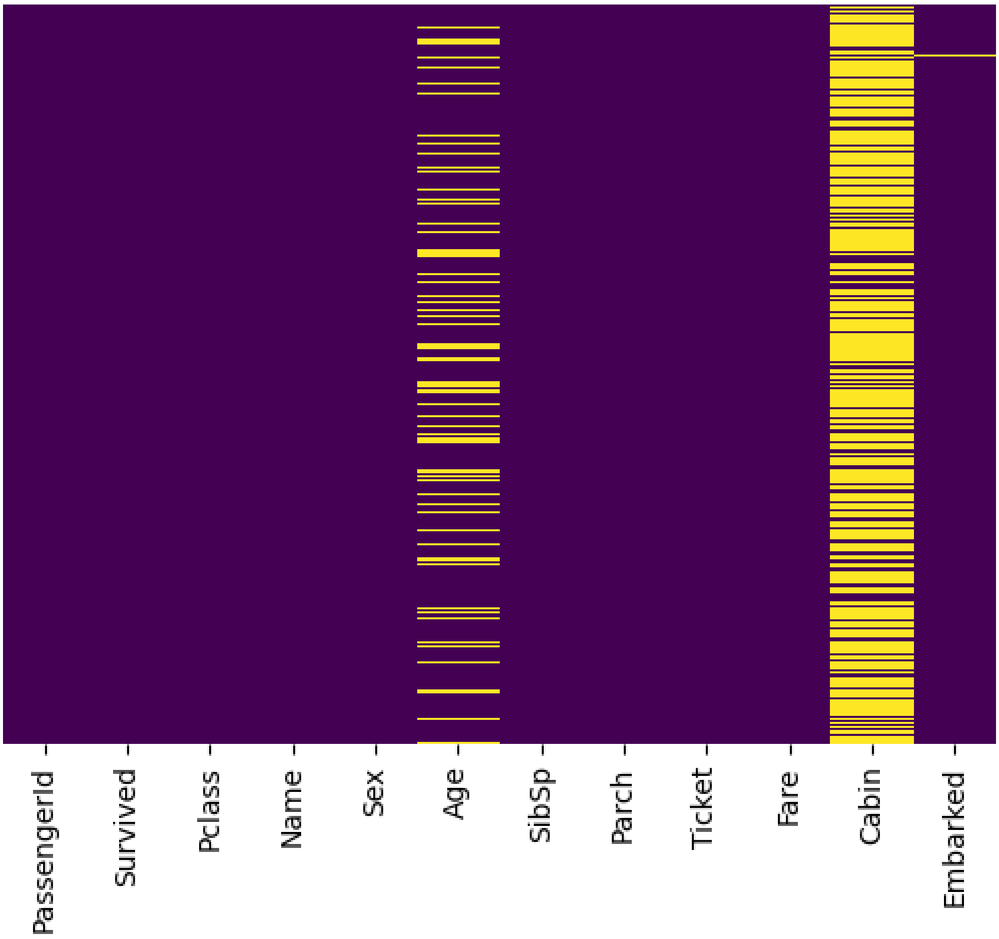
The Kaggle datasets are train.csv and test.csv.

Step 1: Load the Dataset

I loaded the train.csv file using *read_csv()* method of the pandas library.

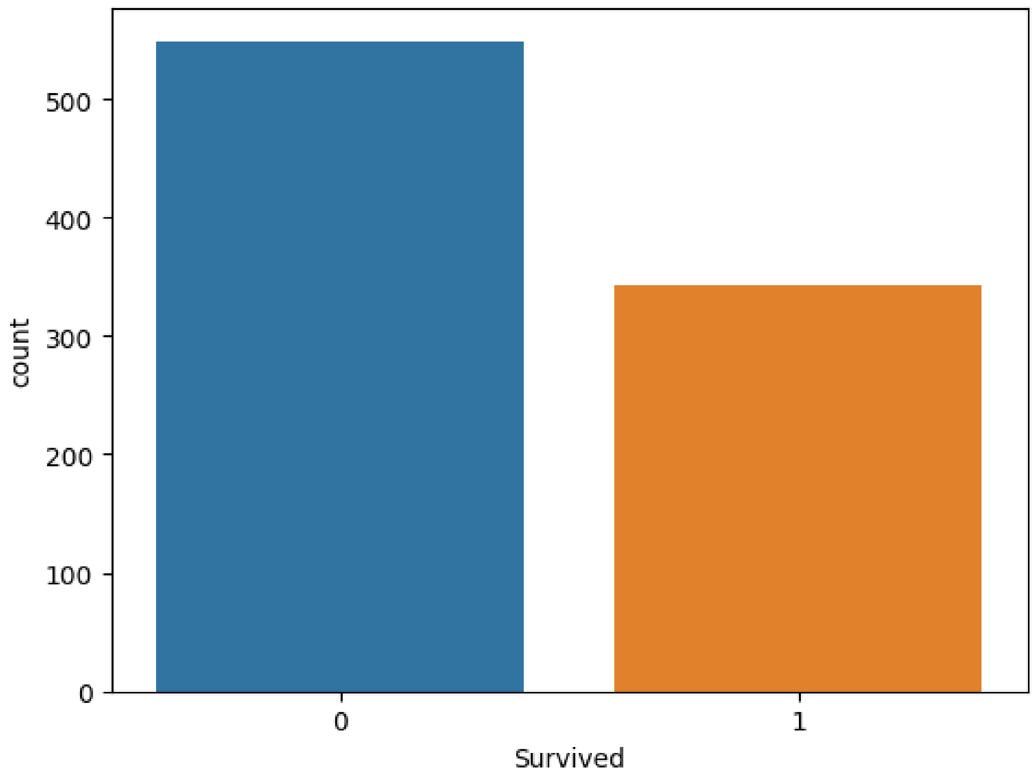
Step 2: Data Engineering – Understanding the Dataset

I displayed a Heatmap to understand the missing values in the dataset. The result shows that the Age column has some NaN values as well as the Cabin column. I will soon handle the NaN values.

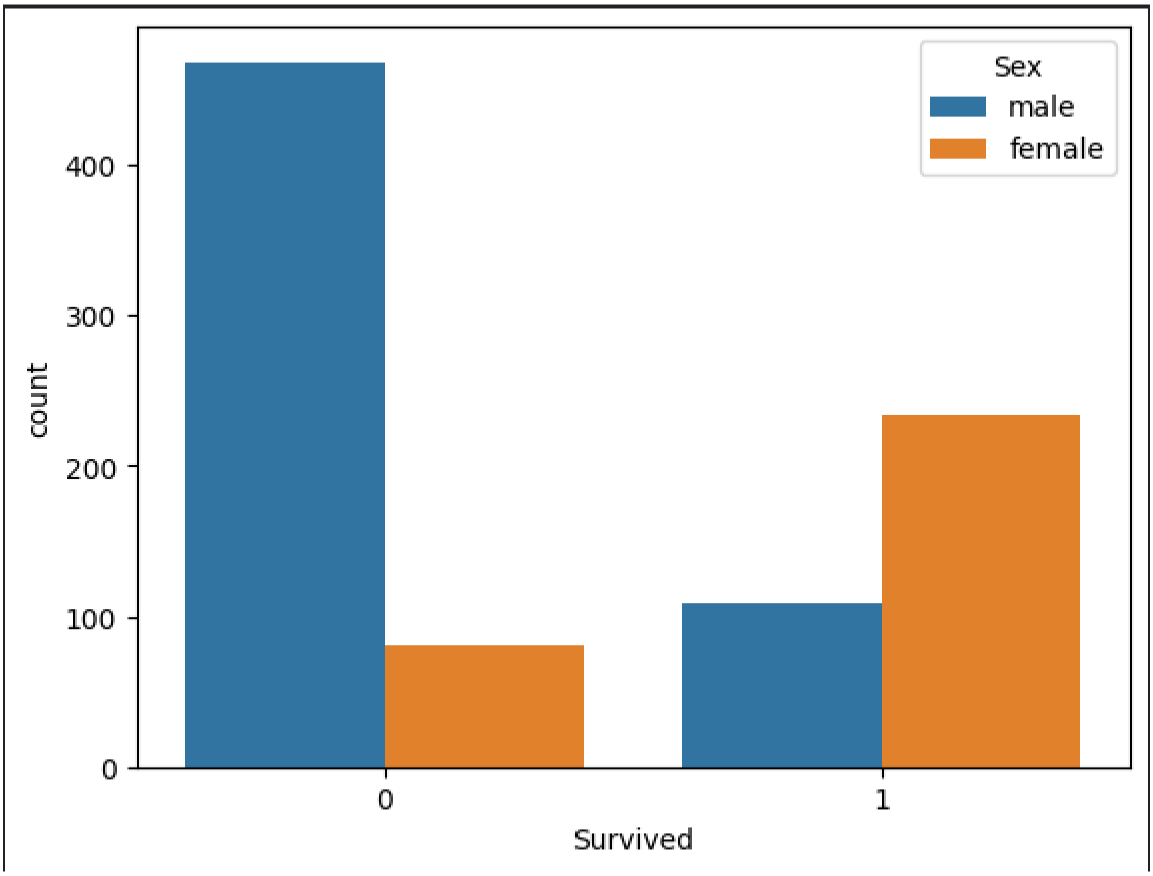


I wanted to understand the statistics behind people who survived and those who died (the deceased), so I plotted countplots.

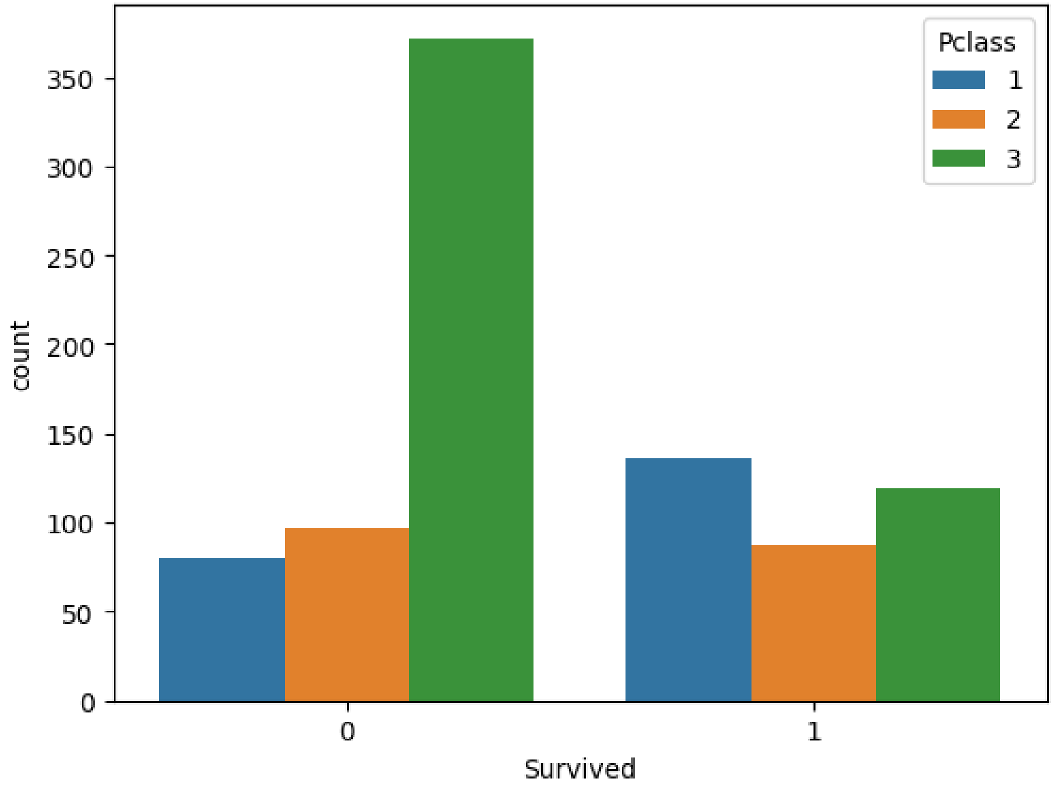
The graph below shows the people that survived and the deceased people. About 350 people survived while about 580 people died.



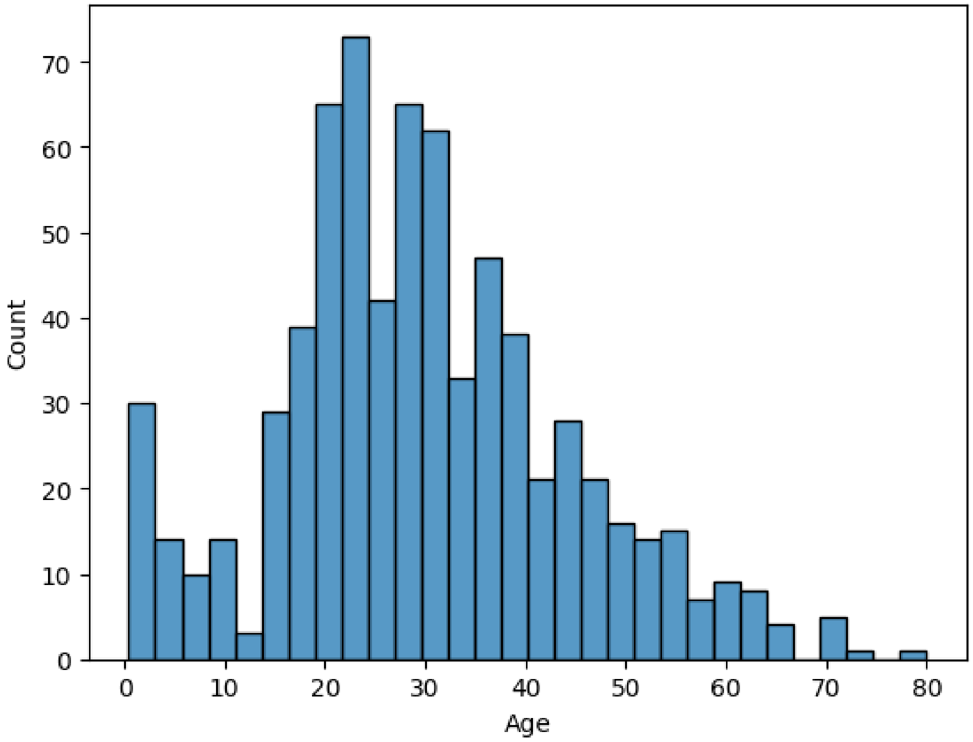
I also plotted to show the proportion of male and female that survived and those who are deceased, using the "Sex" column



I also plotted to show the proportion of people in each of the classes for those who survived and those who are deceased, using the “Pclass” column.

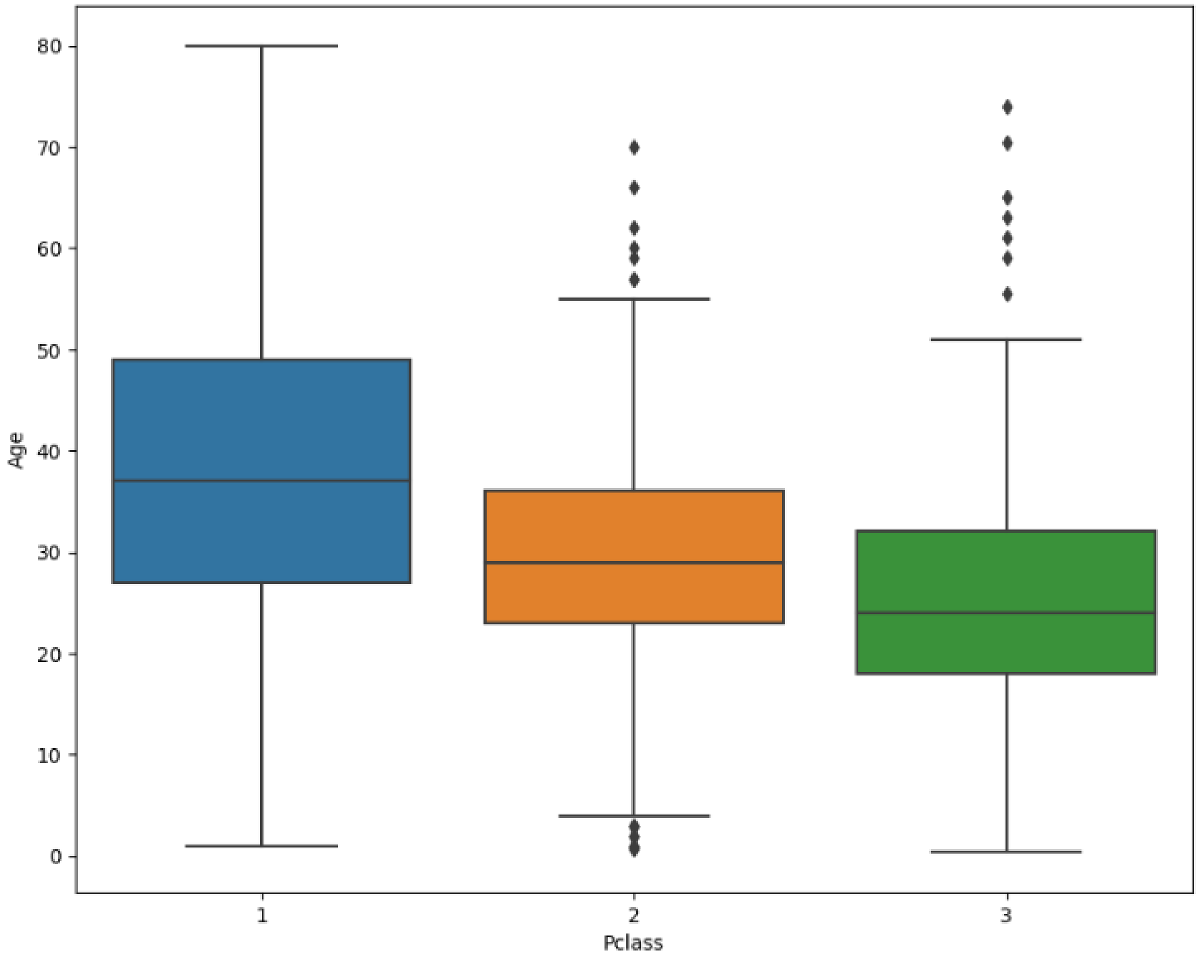


In addition, to understand the age distribution, I plotted a histplot.

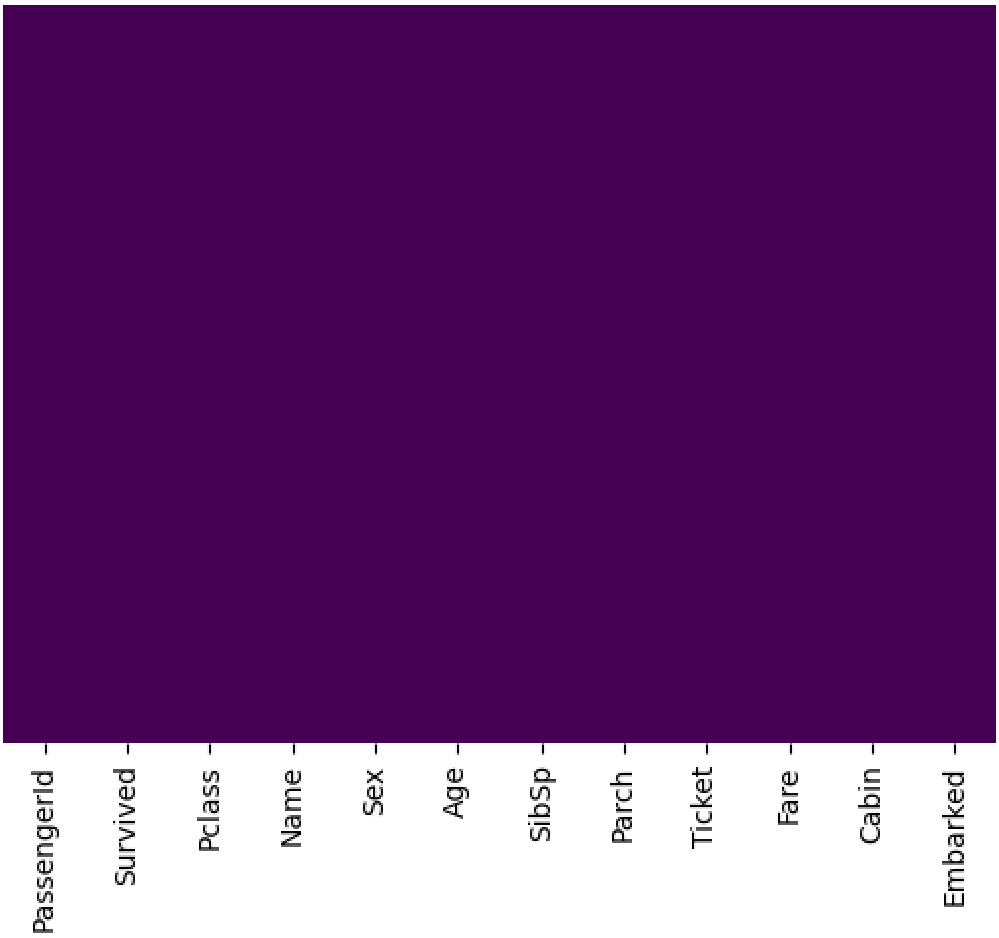


Step 3: Data Preprocessing

I examined the mean ages of the Pclass column and displayed a boxplot to identify the mean ages for each Pclass. I used the mean values to fill the NaN values in the "Age" column.



Also, I cleared all the NaN values in every column. To confirm that there is no NaN values in the "Age" column and other columns, I plotted the heatmap again to show the NaN values.



Part of the data preprocessing done is changing the data type of the Sex column to Numerical values where “male” is 0 and “female” is 1.

Step 4: Training of the Model

I represented the Predictor variables as X which contains features like ‘Pclass’, ‘Sex’, ‘Age’ and ‘Fare’. I represented the Target variable as y, which contains the ‘Survived’ feature.

Splitting of the data into training and testing set is performed using the *train_test_split()* function. After which the Logistic Regression model is initialized.

I fitted the model using the X_train and y_train, and predicted the y_pred using the fitted model.

Step 5: Model Evaluation

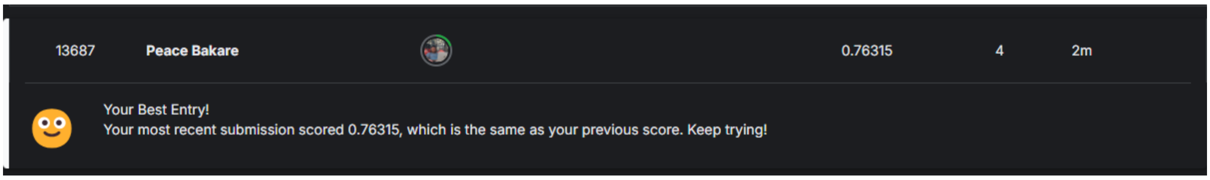
I calculated the model accuracy using the *accuracy_score()* function and got a value of **0.8537 – 85%.**

Step 6 – Testing the Model on the Test Dataset

I loaded the test dataset, performed the same preprocessing I did to the train.csv file, aligning it to the model training features.

I predicted the 'Survived' values for each of the Passengers on the test.csv dataset and saved the result to a file, *test_predicted_survived.csv* which was uploaded to Kaggle for the evaluation of the performance of the Logistic Regression model.

Kaggle Submit Result!



13687 **Peace Bakare** 0.76315 4 2m

Your Best Entry!
Your most recent submission scored 0.76315, which is the same as your previous score. Keep trying!