KNN Classification Report

In the exercise, I only used the features deck level, gender, and age to determine survival status(target) of each passenger. I realized that the other features (PassengerId, Name) will not help me that much in the predictions.

The dataset had missing values. For my baseline model, I removed all observations with missing values using *df.dropna().* This cut down the dataset observations from 1307 to 734. I trained this data, and it gave me an accuracy of 77.8%. I realized that this way took away almost half of the data, which affected my model.

I then decided to drop Na values with regard to a particular column and I used the most importa nt column which was the target using df.dropna(subset=['Survived']). This gave me more data to train as the data was now cut down to 1282. I trained the data which gave me an accuracy of about 81%.

As a preprocessing step, I implemented a column transformer which imputed missing values on the numerical feature (Age) and used a standard scaler to normalize the values. Furthermore, th e transformer used a OneHotEncoder on the categorical features (PClass and Sex) so that they become numerical. OneHotEncoding transforms strings into numbers so that we can apply our Machine Learning algorithms without any problems.

Lastly, on choosing the value of K to be 4, I made my model pipeline and tested it using a wide range of k values and I obtained the highest accuracy of 82.3% when k=4.

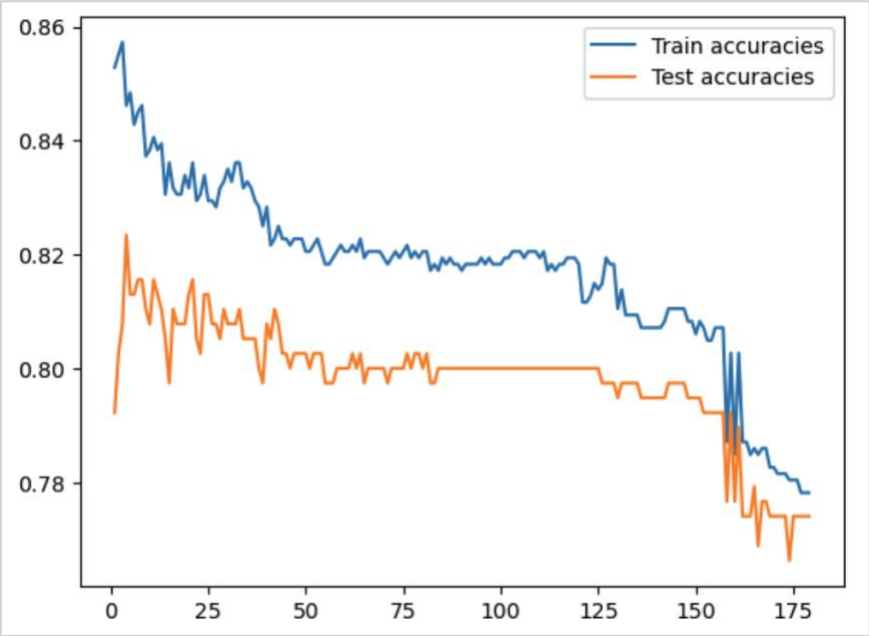

Figure: Choosing K value