**Problem Statement:**

Use machine learning to create a model that predicts which passengers survived the Titanic shipwreck.

**Dataset:**

The data has been split into two groups: training set (train.csv) test set (test.csv)

I used the train dataset for the entire process once and then the train and test dataset or another run.

Link: <https://www.kaggle.com/competitions/titanic/overview>

**Data Preprocessing:**

1. Remove columns that do not impact the survival such as the ones with a high missing data percentage (‘’Cabin) and variables such as ‘Name’
2. Replaced the na values with the mode for variables that had a very low percentage of na values(‘Embarked’)
3. Replaced the na values with the mean for variables that had a medium percentage of na values(‘Age’)
4. Used One Hot Encoding to encode categorical variables

75% of the data was used for training the model and the remaining 25% was used for testing when only the train set was used.

**Models and Techniques Used:**

1. Decision Trees

Using the train set as both test and train: 70% accuracy

1. kNN

Using the train set as both test and train: 55% accuracy

1. Kernel SVM

Using the train set as both test and train: 69% accuracy

1. Logistic Regression

Using the train set as both test and train: 78.9% accuracy

1. Random Forest Classification

Using the train set as both test and train: 80% accuracy

1. SVM

Using the train set as both test and train: 78% accuracy

1. LightBGM

Using the train set as both test and train: 80.26%

1. XGBoost

Using the train set as both test and train: 83%

1. Naive Bayes

Using the train set as both test and train: 78%

Most of the models managed to get around 80% but the decision trees and the random forest were overfitted meaning that more variation is needed.

On the actual test dataset, XGBoost had a 76% accuracy

**Final Results:**

Ensemble methods like Random Forest, LightBGM, and XGBoost demonstrated the best performance, with accuracies ranging from 80.26% to 83% when using the train set for both training and testing