**Titanic Data Analysis & Modeling**

**Table of Contents**

[1. Summary 2](#_Toc123929997)

[2. Benchmarking of Other Solutions 3](#_Toc123929998)

[2.1. Notebook 1 3](#_Toc123929999)

[2.2. Notebook 2 4](#_Toc123930000)

[3. Data Description and Initial Processing 5](#_Toc123930001)

# Summary

The White Star Line owned the Titanic, which was constructed at the Harland and Wolff shipyard in Belfast, Ireland. At the time of its debut in 1912, it was the biggest ship in the world and was believed unsinkable owing to its unique design and use of watertight compartments. The Titanic was designed to be a luxury vessel, and it provided her passengers with a gymnasium, swimming pool, library, and luxurious eating and lodging choices.

On April 10, 1912, the Titanic departed Southampton, England on her first trip to New York City. Before crossing the Atlantic, the ship made stops at Cherbourg, France and Queenstown, Ireland to take up extra passengers. On the evening of April 14, the ship hit with an iceberg and began to sink. More than 1,500 passengers and crew members perished in the Titanic tragedy, despite the crew's attempts to keep the ship afloat.

Significant in history, the sinking of the Titanic has been the topic of several novels, films, and other works of art. The accident drew attention to the need for stronger safety rules in the shipping sector and led to the creation of the International Convention for the Safety of Life at Sea (SOLAS), which defined minimum requirements for ship construction, equipment, and operation.

Kaggle, a platform for data science challenges, introduced the Titanic competition in 2012 as an introduction machine learning task for novices. The competition is based on the sinking of the Titanic, and the objective is to estimate which people survived based on parameters including age, fare, and passenger class. The dataset for the contest includes passenger data such as name, age, gender, ticket class, price, and embarkation point. There is also a column stating whether or not each passenger survived.

The challenge statement for the competition is to utilise this data to develop a machine learning model capable of reliably predicting which passengers survived the accident based on the presented attributes. The model is then assessed utilising a distinct set of passenger test data. The competition's assessment criterion is the proportion of accurate predictions provided by the model, and the objective is to obtain the greatest possible score.

Participants are needed to register a Kaggle account and download the dataset from the competition page in order to compete. The dataset is given as two CSV (Comma Separated Value) files: one for training and one for testing the model. The training set contains passenger data for 891 passengers, whereas the test set has passenger data for 418 people.

# Benchmarking of Other Solutions

I selected following two notebooks from the competition as benchmarking. These notebooks are explained below.

## Notebook 1

The first notebook that I selected was built by *“marinchenko”.* The work done by this person was very simple and he used decision tree algorithm for the classification of passengers. You can find this notebook from the following URL.

<https://www.kaggle.com/code/marinchenko/decision-tree>

In his work, first of all he did some basic exploration on the data that includes the visualization of data by applying different conditions. For example checking the records for the passengers whose fare was more than 20 USD. Then he checked the statistical properties of different features.

After that he directly started feature generation and generated some features by applying different conditions on existing features. Those features include National\_FR, National\_FN etc. Then he encoded the categorical features using one hot encoding technique but then he performed binning on the data fare feature and created three new features low, medium and high. By doing this he created 10 new features in the dataset.

But in the end, he selected only 6 features manually for further process. Those features are…

* PClass
* Sex
* Fare
* Embarked\_Q
* Embarked\_S
* Embarked\_C

And then on these features he builds decision tree with max\_depth of 3 and random\_state of 2018. Then he trained the model on training data and generated the predictions for test data that were submitted for evaluation. After evaluation on testing data he achieved 77.99% accuracy.

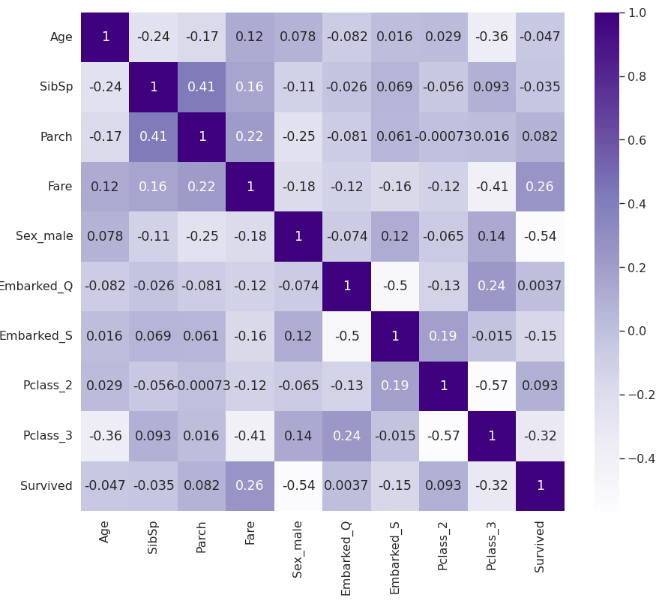
## Notebook 2

Second notebook that I selected was built by *“Farzad Nekouei”.* This notebook was very optimized and full of information. You can find this notebook from the following given link.

[*https://www.kaggle.com/code/farzadnekouei/titanic-logistic-regression-95-accuracy*](https://www.kaggle.com/code/farzadnekouei/titanic-logistic-regression-95-accuracy)

In this notebookFarzad explored the data in the start and then he started data preprocessing. First of all he checked missing values and found that in three features missing values are present. One is age, second is Cabin and third is Embarked feature. For age he took median with respect the Pclass and imputed it in the missing values. And for the Embarked feature he used most frequent value of that feature to replace it with missing values in that feature. In Cabin feature almost, 77 percent values were missing so he dropped this column. Similar thing he did for both training and testing data for imputing missing values.

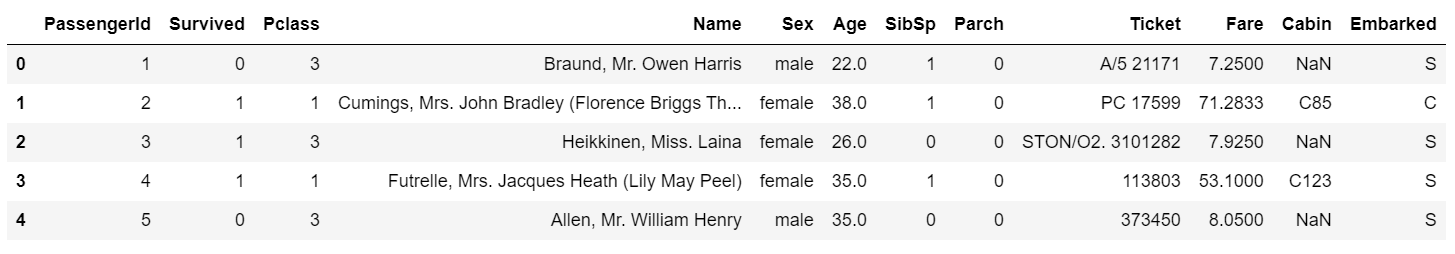
Then he did feature selection and first of all he dropped PassengerID and Name features from the data. Then he also dropped Ticket feature because there were large number of unique values in that feature. Then he performed EDA on the data. And then he build correlation matrix that can be seen below.



Then he built Logistic Regression Model and achieved and accuracy of 94.6 percent. Then the tuning of hyperparameters of this model was also done and found the optimal values for these parameters. And then he builds model on tuned parameters and achieved accuracy of 95. 4 percent.

# Data Description and Initial Processing

The data for the Titanic competition on Kaggle consists of passenger information such as name, age, gender, ticket class, fare, and port of embarkation. The data is provided in the form of two CSV (Comma Separated Values) files: one for training the model and one for testing the model. The training set consists of passenger information for 891 passengers, and the test set consists of passenger information for 418 passengers. Test dataset was having 12 columns and training data was having 11 features. First look of the dataset is given below.



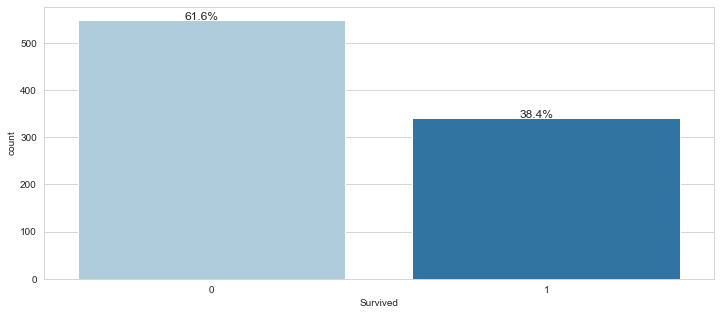
The training set includes a variety of features that can be used to build a machine learning model. Some of the key features include:

* PassengerId: a unique identifier for each passenger
* Pclass: the ticket class of the passenger (1 = first class, 2 = second class, 3 = third class)
* Name: the name of the passenger
* Sex: the gender of the passenger
* Age: the age of the passenger in years
* SibSp: the number of siblings or spouses the passenger had on board
* Parch: the number of parents or children the passenger had on board
* Ticket: the ticket number of the passenger
* Fare: the fare paid by the passenger
* Cabin: the cabin number of the passenger
* Embarked: the port of embarkation (C = Cherbourg, Q = Queenstown, S = Southampton)

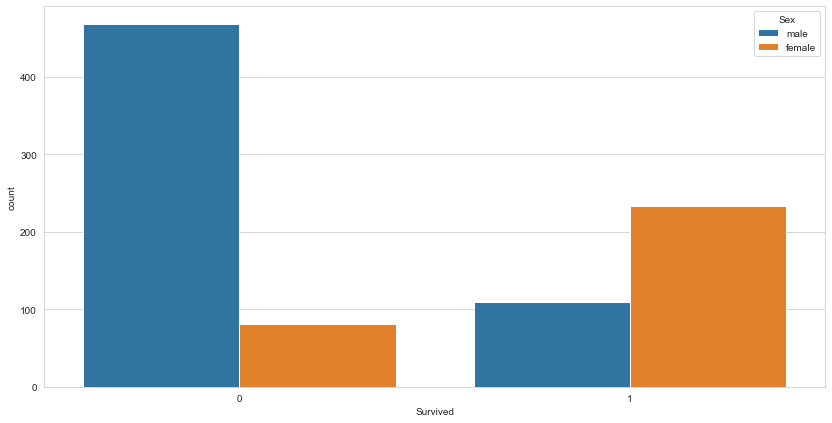
The test set also includes these features, with the exception of the "Survived" column, which is the target variable for the competition.

Overall, the data for the Titanic competition is a good representation of real-world data, as it includes both numerical and categorical features, as well as missing values and other challenges that are commonly encountered when working with data. The data is also relatively small, which makes it a good choice for beginners who are just getting started in machine learning.

Then I performed some explorations on the data that are given and explained below.

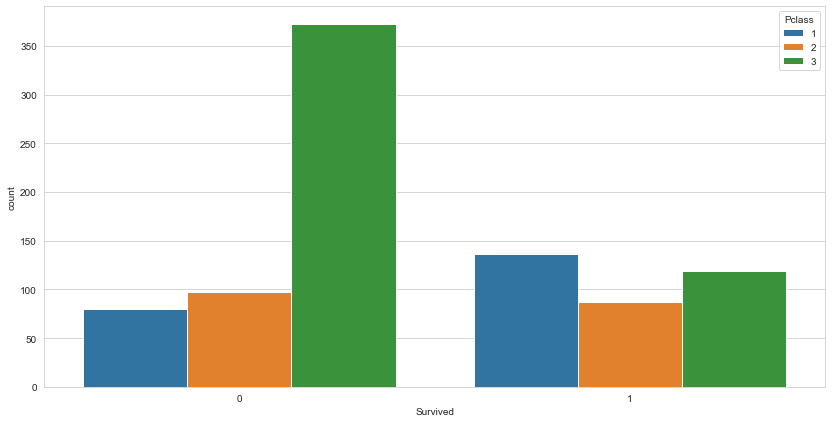


This plot was made for target feature and we can see that 61 percent data is for people who were not survived. So, the data among the classes is imbalanced.



In the Titanic dataset, it is apparent that a larger proportion of female passengers survived the disaster compared to male passengers. This can be seen by examining the relationship between the sex of the passengers and their survival status.

In the context of the Titanic dataset, it appears that a higher proportion of passengers who travelled in first class survived the disaster compared to those in lower classes. This can be seen in the following graph.



Identifying whether features contain null values is the initial stage of preprocessing. In this instance, three attributes have null values: "Cabin," "Age," and "Embarked." It is vital to manage these null values with care and consideration, since they can drastically influence the model's performance if not handled appropriately.

77% of the values for the "Cabin" feature are null, making it extremely troublesome. In this instance, it is decided to eliminate the "Cabin" function completely. This is not always the best course of action, and it is essential to examine the potential repercussions of removing a feature with so many null values. In this instance, however, the "Cabin" characteristic is regarded less significant for the model's predictions and is so eliminated.

The null values for the 'Age' feature are handled differently. Instead of discarding the feature, the null values are filled with the feature's mean value for each value. This implies that the mean age for each passenger class (1st class, 2nd class, etc.) is determined, and these data are utilised to fill in the "Age" feature's blanks. This method of imputing null values is widespread since it helps preserve the data's structure and connections.

The null values in the "Embarked" feature are populated with the mode value, which is the value that occurs most frequently in the feature. This is a straightforward and efficient method for dealing with null values in categorical features, and it can assist verify that the data is full and ready for modelling.

After handling null data, it is necessary to remove the "PassengerID" and "Name" features. These characteristics are eliminated because they are deemed irrelevant to the model's predictions. The "Sex" and "Embarked" features are then one-hot encoded, which involves converting them into a series of binary columns, with a column for each unique value in the feature. This is a frequent method for categorical data, as it enables the model to efficiently learn from these characteristics.

Using oversampling, the minority class (in this case, the class with fewer samples) is duplicated and added to the dataset in order to balance the class distribution.

# Modeling

Modeling is the process applying machine learning algorithms on the dataset. So here I selected following features of data to used in the machine learning models.

* PClass
* Age
* SibSP
* Parch
* Fare
* Sex
* Embarked

But before building the model I split my dataset into training and evaluation part by the ratio of 70:30 percent. I used following machine learning algorithm to test out.

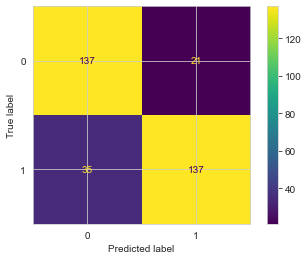
* KNN
* Logistic Regression
* Decision Tree
* Random Forest

And for the evaluation of models I used following evaluation metrics.

* Accuracy
* Precision
* Recall
* F1 Score

## Logistic Regression

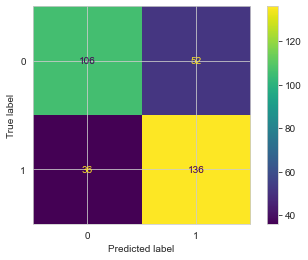
The accuracy of this model on the evaluation data was 83.03 percent. And the confusion matrix of the model is given below,



Total number of records in the evaluation data are 330 and from these 330 records 60 records were miss classified. The precision for class 0 is 80 percent and for class 1 it is 87 percent. That’s mean the model is little biased towards class 1.

## KNN

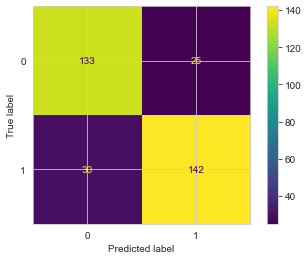
Accuracy of KNN model with default parameters on evaluation data is 73.3 percent that was low from Logistic regression model. And the confusion metric is given below,



More than 85 records were miss classified and model was fair among both classes.

## Random Forest

Random Forest was also having same accuracy to logistic regression that was 83.33 percent. And in confusion metric 55 records were miss classified.

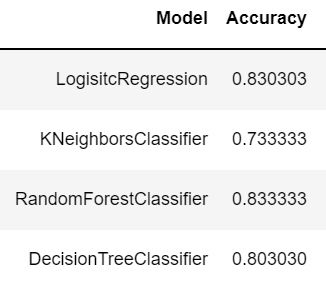


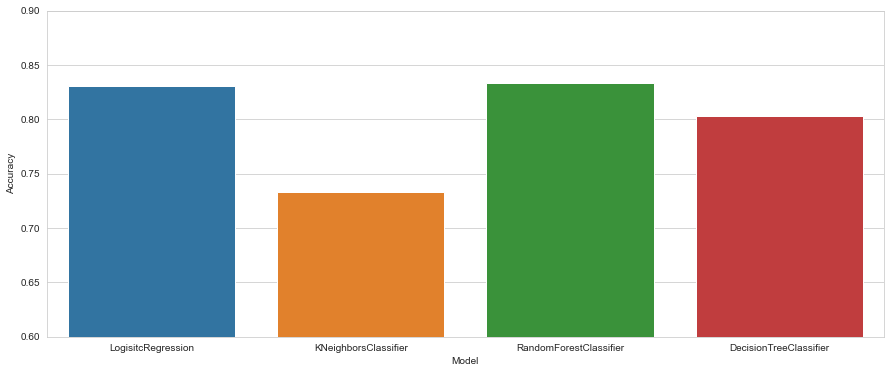
Model was fair among both classes.

## Decision Tree

Decision was having high accuracy than KNN but less than logistic and random forest model. Its accuracy was around 80 percent.

# Comparison of all Models





# Appendix

https://github.com/umut28035/Game-AI-Term-Project.git