MATH4995
Project 1 - Titanic: Machine Learning from Disaster

HARJONO, Natasha Valerie
TSUI, Ying Tsz
WAHYU, Zoya Estella
Kaggle Group Name: MATH4995_Harjono_Tsui_Wahyu

1. Introduction

We have chosen the 'Titanic: Machine Learning for Disaster' as our first project. In this project we will use machine learning techniques to create a model that predicts which passengers survived the Titanic shipwreck. In this report, we will discuss how we process the given data and choose a model to solve this prediction problem.

2. Data Overview

There are two datasets in csv format which is, the training set and the test set.
The training set has 891 rows with 12 columns with one observation identifier (PassengerId), one target variable (Survived), and 10 predictor variables. Meanwhile, the test data has 418 rows with 11 columns, without the target variable column.

3. Data Preprocessing

3.1. Data Visualization

We want to show the relationship between each feature and the survival rate also we want to figure out if there are any correlation between one feature to another.

3.1.1. Pclass

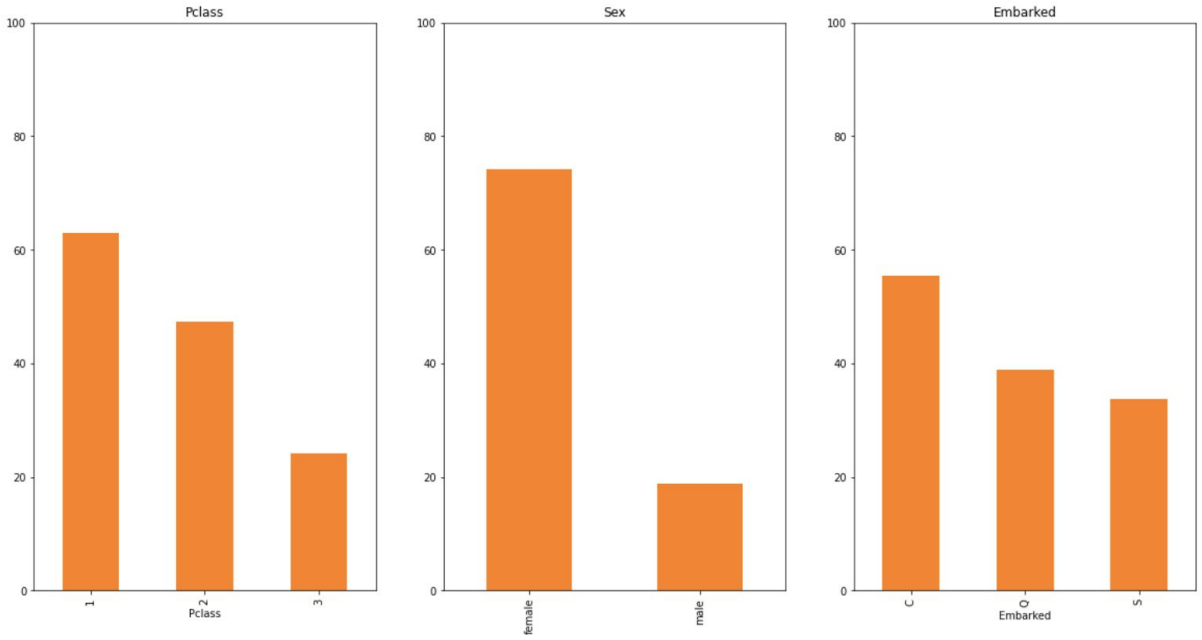
We plot the person class against survived to see if being in a certain class increases or decreases their survival chance. We can see that people from class 1 are very likely to survive with over 0.6 chance which is then followed by class 2 with less than 0.5 and class 3 around 0.2.

3.1.2. Sex

We plot sex against survived to help determined whether being female or male affect their survival rate. This plot shows that females are very likely to survive with over 0.7 probability rate. While the male survival chances are very low with only approximately 0.2 probability rate.

3.1.3. Embarked

We plot the Embarked against survived to see if where a person gets on the ship influences their survival rate. The plot shows that people that embarked on C have the highest survival chance followed by Q then S.

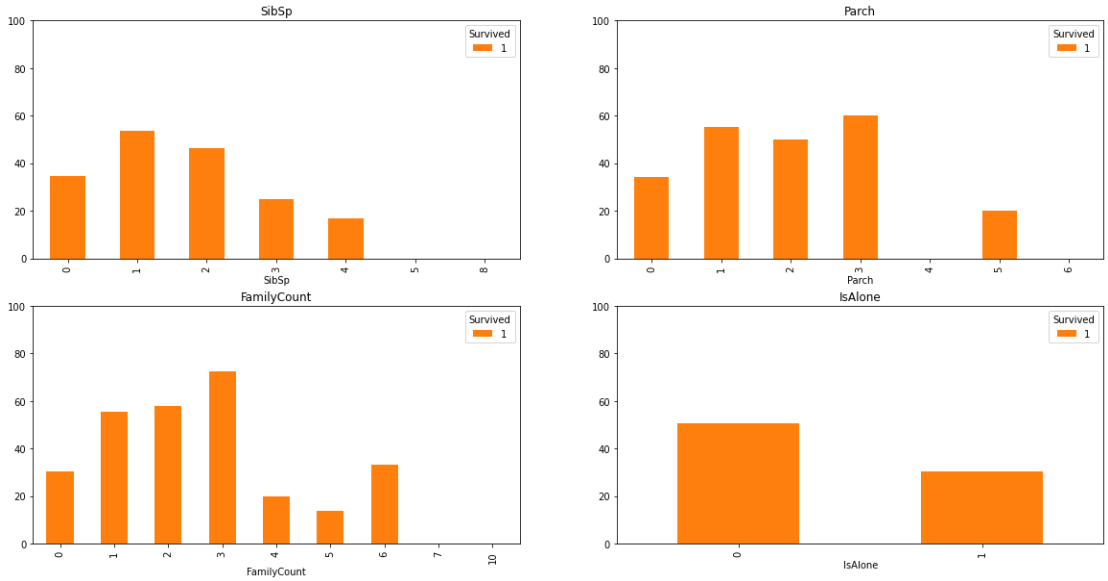


3.1.4. Sibsp and Parch/ Family

We want to see whether the number of people who travel together affect the survival rate and what happened to those who travel alone. Therefore, we created 2 new variables (more explanation can be found on feature engineering section):

FamilyCount: No. of Parents/Children + No. of Siblings/Spouses

IsAlone: travels alone = 1, travels in groups = 0

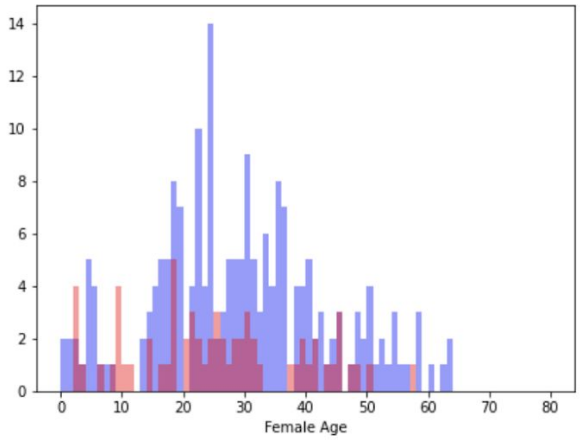


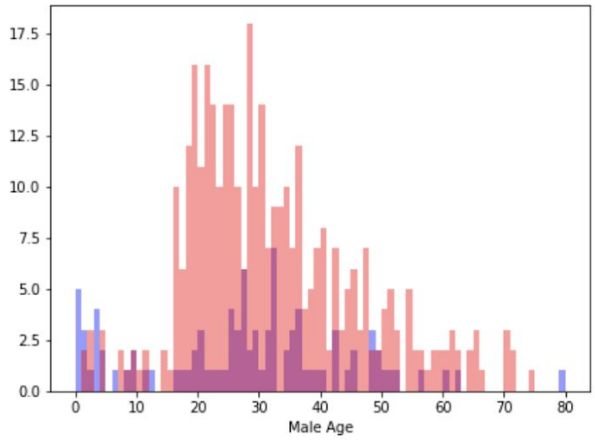
We then plot them all against survived. We can see that those who travel with 1-3 people have a higher chance to survive, followed by people who travel alone, and lastly those who travel with more than 3 people.

3.1.5. Age

We plot age against survived to see if the age of a person can help determine their survival rate or not. Also, to be more precise, we divided the age into 2 categories which are female and male. From the plot, we can see that:

- Children below the age of 10 have a high survival rate.
- Females with age range 15-36 and above 50 have the highest chance to survive.
- Males with age more than 55 have a slim chance of surviving.





3.2. Missing Values

3.2.1. Fare

There are 1 missing value in Fare. Since the number is very small, we use the column's median to fill it.

3.2.2. Embarked

There are 2 missing values in Embarked. We first find the Class of the null data, so that we can compare it with the median of the fare that belongs to the same class as the missing data for each Embarked point. We will choose the Embark point which median fare value is the closest to the fare of the missing person.

3.2.3. Cabin

There are a total of 1014 missing values from the training and test data. It accounts to more than two-thirds of the test data. We do not want to make any assumptions about the missing values. Therefore, we marked them with "X".

3.2.4. Age

There are a total of 233 missing data from the training and test data. Due to the large amount of missing data, in order to figure out the missing ages, we build a model using Random Forest Regressor. The features we use include Fare, Parch, SibSp, Class, Embarked, and Title.

4. Feature Engineering

4.1. New Features

4.1.1. Title

Although the name of passengers does not have any influence over one's survival rate, there is a title in each name. We decided to change the "Name" column into "Title". Furthermore, some titles have similar meanings so we grouped some of them together. For example, "Mlle", "Ms", and "Miss" will be grouped in "Miss". All special titles such as "Don", "Rev", etc. will be combined into one group named "Rare". And lastly, we assume people that do not pay any fare as "Worker".

4.1.2. Deck

Deck is the first alphabet character of the Cabin column. This is used to increase the group size of each unique value of Deck.

4.1.3. FamilyCount

FamilyCount is a feature that collects the number of family members on board. It adds up the number of parents, children, siblings, and spouses, i.e. Parch+SibSp.

4.1.4. IsAlone

When a person travels alone (without any family member), the IsAlone value will be 1, otherwise, it will be 0.

4.1.5. FarePerPerson

We found the values in the Fare column are the same for people with the same ticket number. We then conclude that the Fare values are the fare paid for the group. In order to find the fare paid per person and store it to FarePerPerson we divide the Fare by the number of people in the group (assuming all groups are family) instead of by the number of people with the same ticket number because the training data do not include all people at the ship.

4.2. Finalizing the Data

We transform all of the categorical variables into indicator variables (binary) to later feed all data into the model.

In the end, after including the new features, there are 29 predictors in total: Sex, Age, SibSp, Parch, Fare, FamilyCount, IsAlone, FarePerPerson, Pclass_one,

Pclass_two, Pclass_three, Title_Master, Title_Miss, Title_Mr, Title_Mrs, Title_Rare, Title_Worker, Embarked_C, Embarked_Q, Embarked_S, Deck_A, Deck_B, Deck_C, Deck_D, Deck_E, Deck_F, Deck_G, Deck_T, Deck_X.

5. Data Processing

5.1. Model Selection

In order to maximize the accuracy of our prediction, we tried several models to find which one will yield the highest score.

```
Linear Regression model: 45.73%
Naive Bayes Classifier model: 80.13%
Random Forest model: 98.77%
Decision Tree model: 98.77%
KNN model: 81.71%
Logistic Regression model: 84.29%
```

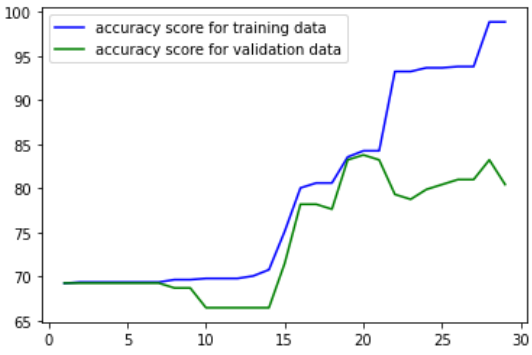
The result shows that the Random Forest and Decision Tree generate the highest score. To further investigate which model suits our data more, we use K-fold cross validation to divide our data into 5 permutation groups of training and validation data. We then fitted each group into both the Random Forest and the Decision Tree model. Then, we compared the final mean and standard deviation of the average accuracy scores across the 5 groups with each other.

```
          Score Mean +/- Std
Random Forest: 80.69 +/- 2.97 %
Decision Tree: 77.67 +/- 2.74 %
```

Since the mean of the Random Forest model is higher, we will use the Random Forest for our model.

5.2. Feature Selection

We have 29 predictors in order to obtain the results, however, not every predictor holds the same importance. We want to find the optimum numbers of predictors to be included in our model. First, we find the significance value of each feature and sort it from the least to the most significant. Then we fitted our data to the model while removing the feature one by one and comparing the scores.



The plot shows that using 20 features will generate the highest score. Therefore, we will use these 20 features of highest significance values. The final model will exclude

these 9 features: Deck: A, B, C, D, F, G, T; and Title: Worker, Rare.

5.3. Model Assessment

We assess the final model using K-fold cross validation (K=5) again. Then, we generate its confusion matrix and the results are as follows:

TN: 412	FP: 34
FN: 95	TP: 171

The overall error rate is 18.12%, with precision score of 83.41% and recall score of 64.29%. F1-score of the model is 72.61%.

6. Conclusion

We feed the test data to the finalized model and obtain an accuracy score of 78.95% with an error rate of 21.05%. This is very close to the overall error rate obtained in the model assessment.

7. Contribution

HARJONO, Natasha Valerie:

1. data cleaning
2. data visualization
3. feature engineering

TSUI, Ying Tsz:

1. data cleaning
2. data finalization
3. model selection
4. hyperparameter tuning

WAHYU, Zoya Estella:

1. data cleaning
2. feature selection
3. model assessment (decision tree vs. random forest, confusion matrix)

Link to the repository: https://github.com/zoyaew/MATH4995