Assignment 1 – Titanic Problem

Report

Task – Titanic Challenge

Overview:

The Titanic Challenge is based on the historic sinking of passenger ship famously named RMS Titanic in the middle of the Atlantic on April 15, 1912.

The Kaggle challenge provides 891 Passenger data as the training set which included many fields such as whether the passenger survived or not.

The goal is to predict the survival of 418 Passengers given as the test set for this problem based on features such as age, sex, passenger class etc.

To solve this problem, we can use various machine learning models. I chose Logistic Regression, Kernel SVM and Random Forest to solve the titanic problem.

Main libraries I used for this task –

- Pandas – for data manipulation and ingestion.
- Matplotlib – for data visualization.
- Numpy – for multidimensional array computing.
- Sklearn – for machine learning and predictive modelling.

Datasets:

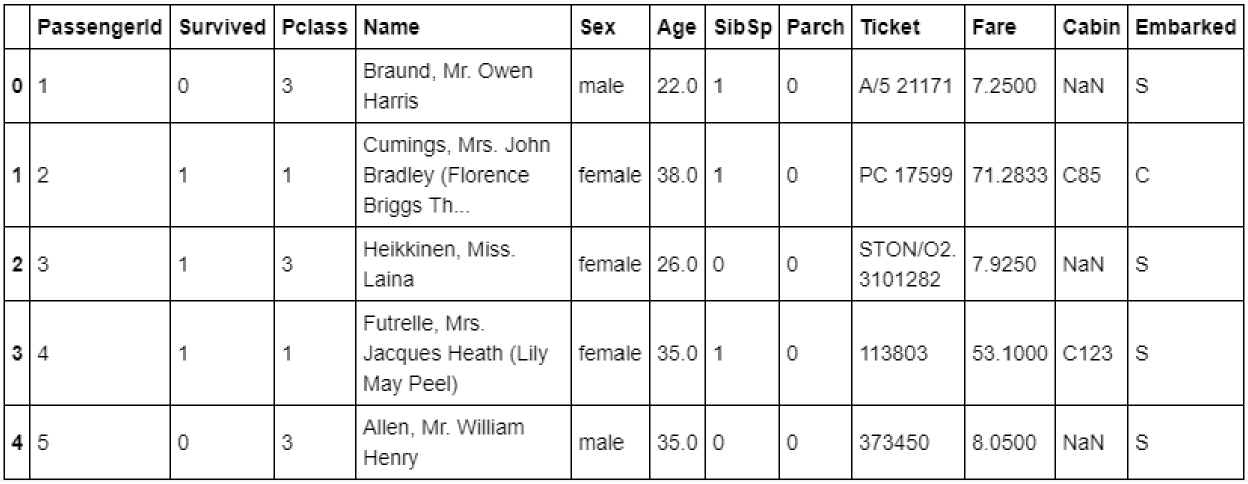
The training dataset has 891 rows and 12 columns.

Below is a brief information about each columns of the dataset:

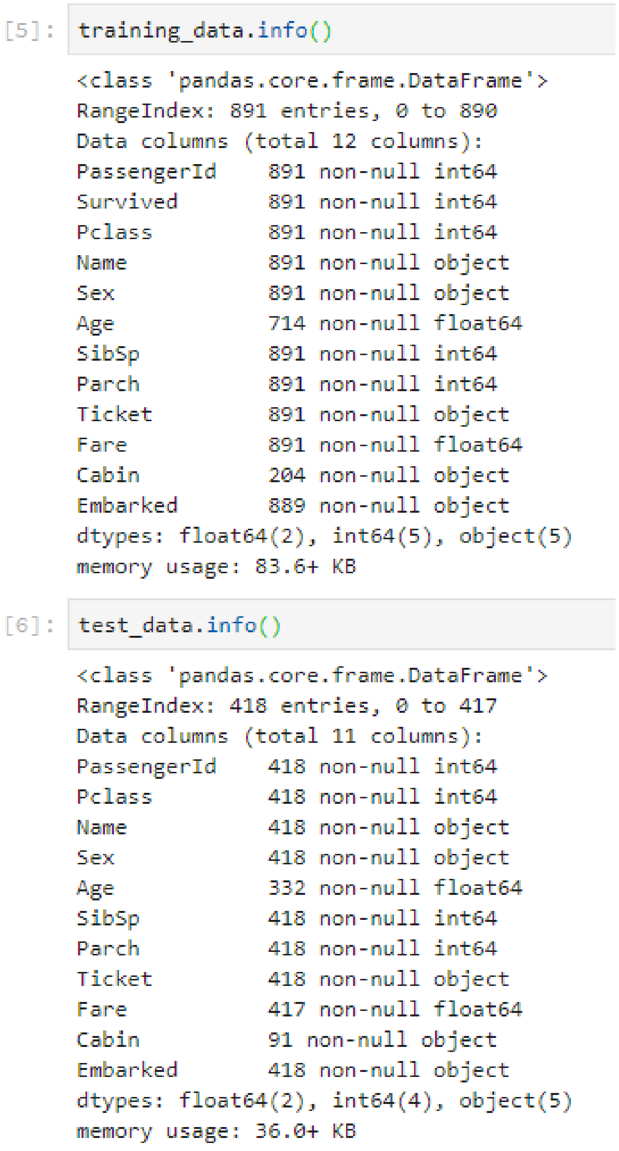
1. PassengerId: A unique index for passenger rows.
2. Survived: Shows if the passenger survived or not. 1 for “survived” and 0 “not survived”."

3. Pclass: Ticket class. 1 stands for First class ticket. 2 stands for Second class ticket. 3 stands for Third class ticket.
4. Name: Passenger's name. Name also contain title. "Mr" for man. "Mrs" for woman. "Miss" for girl. "Master" for boy.
5. Sex: Passenger's sex. It's either Male or Female.
6. Age: Passenger's age. "NaN" values in this column indicates that the age of that particular passenger has not been recorded.
7. SibSp: Number of siblings or spouses travelling with each passenger.
8. Parch: Number of parents of children travelling with each passenger.
9. Ticket: Ticket number.
10. Fare: How much money the passenger has paid for the travel journey.
11. Cabin: Cabin number of the passenger. "NaN" values in this column indicates that the cabin number of that particular passenger has not been recorded.
12. Embarked: Port from where the particular passenger was embarked/boarded.

- <u>Pandas allows to take a look at this data –</u>



	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	NaN	S
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.2833	C85	C
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250	NaN	S
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000	C123	S
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500	NaN	S



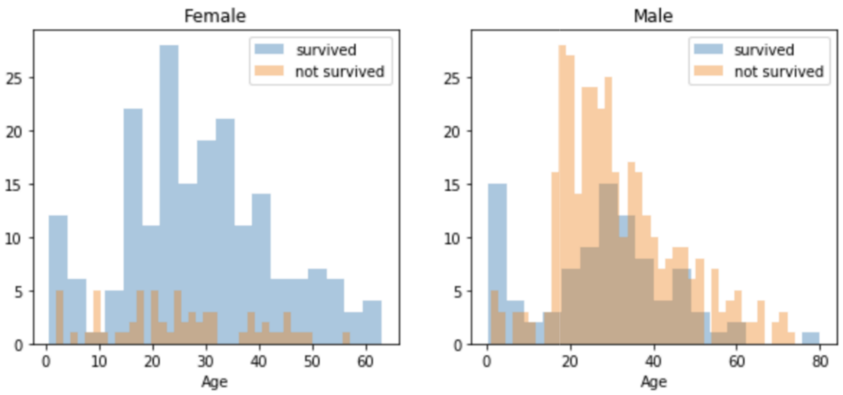
```
[5]: training_data.info()
```

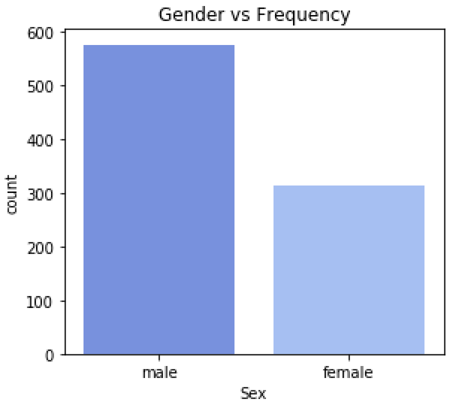
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 891 entries, 0 to 890
Data columns (total 12 columns):
PassengerId    891 non-null int64
Survived       891 non-null int64
Pclass         891 non-null int64
Name           891 non-null object
Sex            891 non-null object
Age            714 non-null float64
SibSp          891 non-null int64
Parch          891 non-null int64
Ticket         891 non-null object
Fare           891 non-null float64
Cabin          204 non-null object
Embarked       889 non-null object
dtypes: float64(2), int64(5), object(5)
memory usage: 83.6+ KB
```

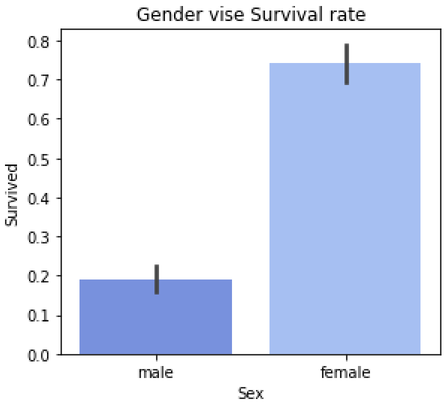
```
[6]: test_data.info()
```

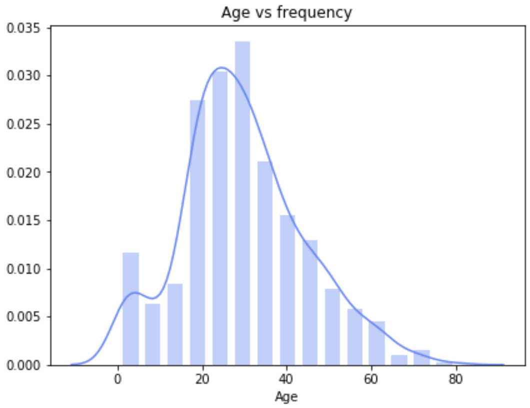
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 418 entries, 0 to 417
Data columns (total 11 columns):
PassengerId    418 non-null int64
Pclass         418 non-null int64
Name           418 non-null object
Sex            418 non-null object
Age            332 non-null float64
SibSp          418 non-null int64
Parch          418 non-null int64
Ticket         418 non-null object
Fare           417 non-null float64
Cabin          91 non-null object
Embarked       418 non-null object
dtypes: float64(2), int64(4), object(5)
memory usage: 36.0+ KB
```

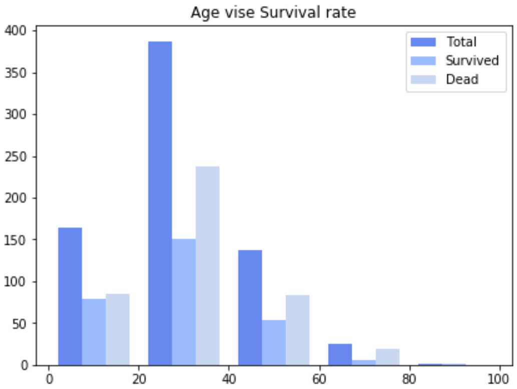
- **Feature Correlation Visualizations –**











Preparing the Data :

Feature Extraction –

- We select appropriate features to train our classifiers and drop the ones which do not have any significant contribution to the prediction of survival directly
- We also convert the categorical features into numerical form.
- We also take care of null valued features using mean values.

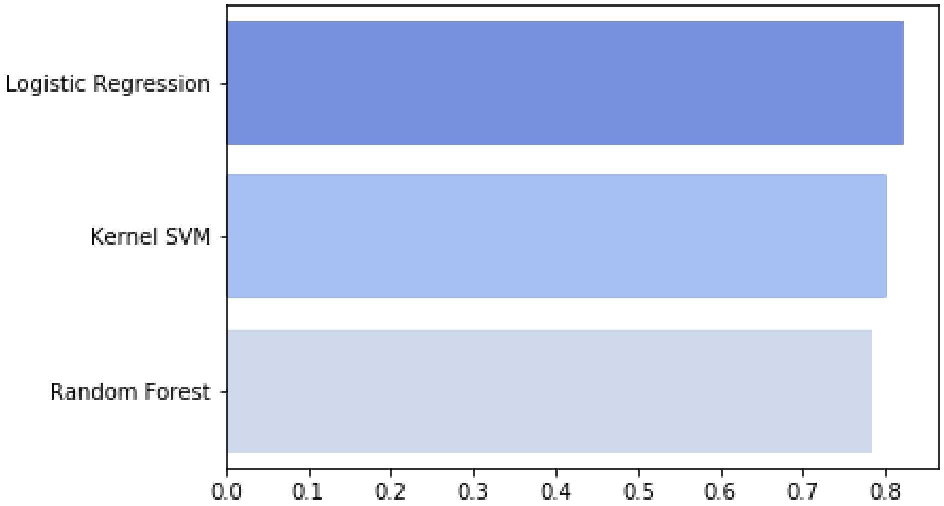
Cleaning the data for classification –

- We split the dataset into dependent and independent features.
- Further we split the datasets into training and testing sets.
- We also use feature scaling.

Model Evaluation and Selection :

Models used to solve the titanic problem

1. Logistic Regression
2. Kernel SVM
3. Random Forest



We evaluate these models based on model accuracy plots and will apply grid search on the final model for parameter fine tuning.

For Random Forest model I used Information gain entropy to improve the performance metrics of the model.

We will use Precision and Recall , ROC AUC curve for further model evaluation.

Result of model –

I chose Random Forest as the final model for this prediction task. On average it has 81% Accuracy.

I attached the correct format submission file I used at Kaggle with this assignment.