***Titanic Survival Prediction Blog & Article Details.***

1. ***Problem Definition:***

We will analyse the Titanic data set and make two predictions.

One prediction to see which passengers on board the ship would survive and then another prediction to see if we would’ve survived.

Description: This program predicts if a passenger will survive on the titanic.

1. ***Data Analysis:***

Pclass: Passenger Class (1 = 1st; 2 = 2nd; 3 = 3rd)

Survived: Survival (0 = No; 1 = Yes)

Name: Name

Sex: Sex

Age: Age

Sibsp: Number of siblings/spouses aboard

Parch: Number of parents/children aboard

Fare: Passenger fare (British pound)

Embarked: Port of embarkation (C = Cherbourg; Q = Queenstown; S = Southampton)

adult male: A male 18 or older (0 =

No, 1=Yes)

Age is fractional if less than 1. If the age is estimated, is it in the form of xx.5

Sibsp

The dataset defines family relations in this way:

Sibling= brother, sister, stepbrother, stepsister

Spouse= husband, wife (mistresses and fiancés were ignored)

Parch

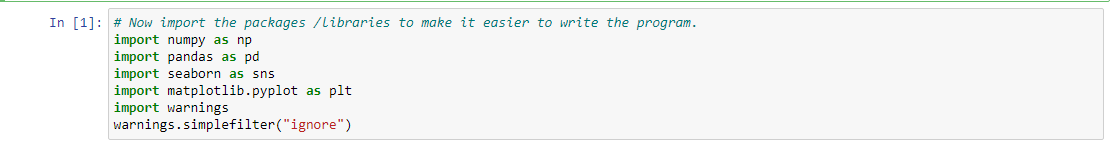
The dataset defines family relations in this way:

Parent= mother, father

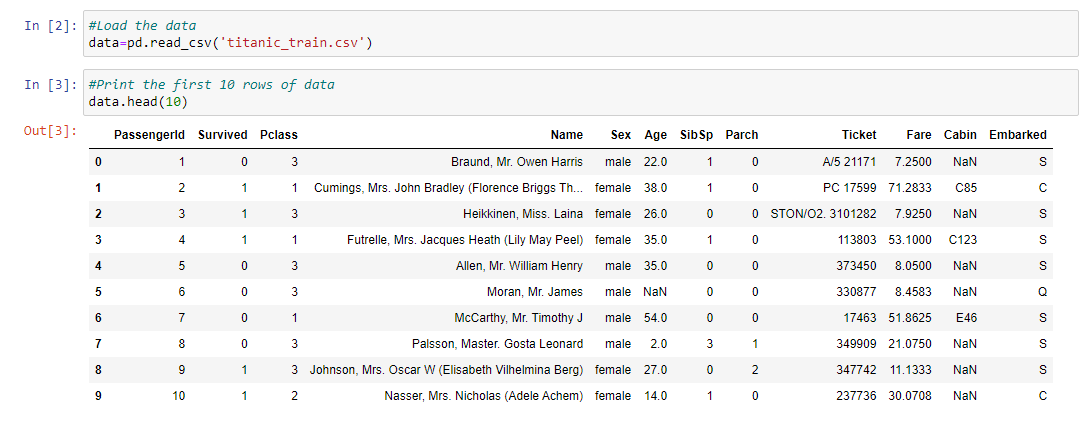
Child= daughter, son, stepdaughter, stepson

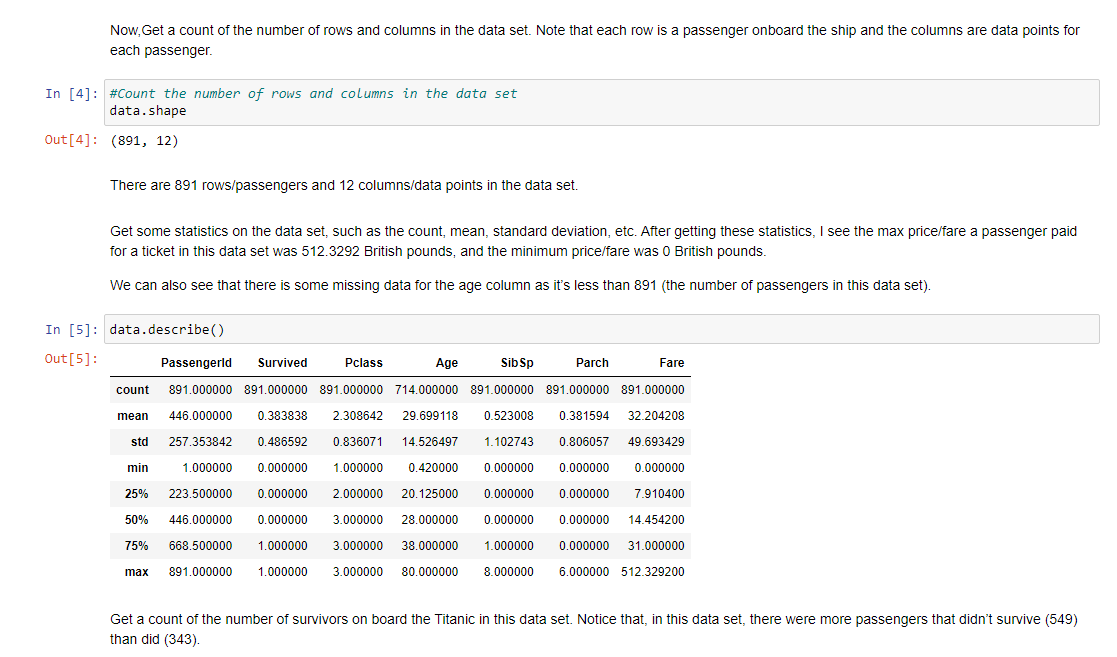
Some children travelled only with a nanny, therefore parch=0 for them.

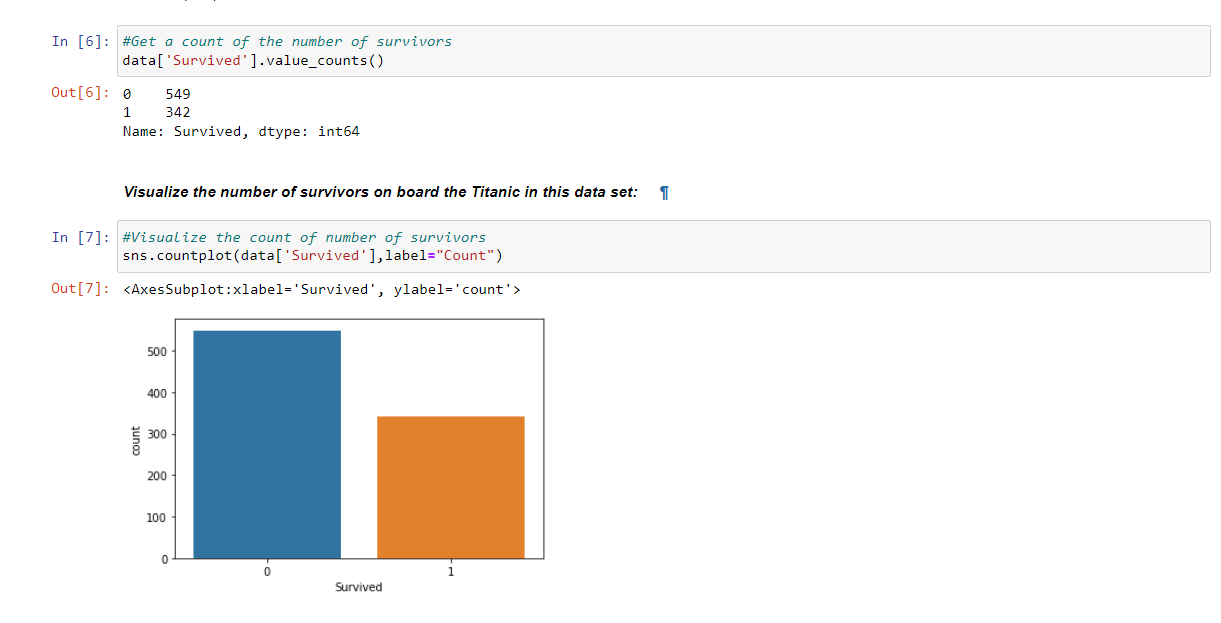
* ***Now import the packages /libraries to make it easier to write the program.***



1. ***EDA Concluding Remark:***

******

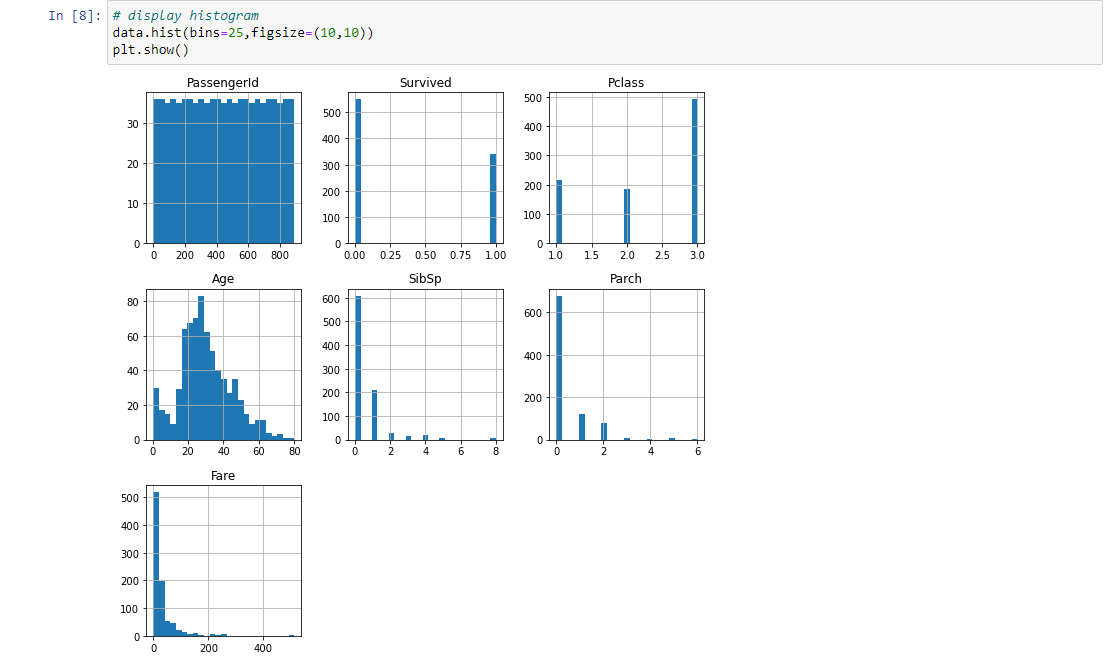




From the charts below, Females are most likely to survive from the chart gender.

Third class is most likely to not survive by chart Pclass. If you have 0 siblings or spouses on board, you are not likely to survive according to chart Sibsp.

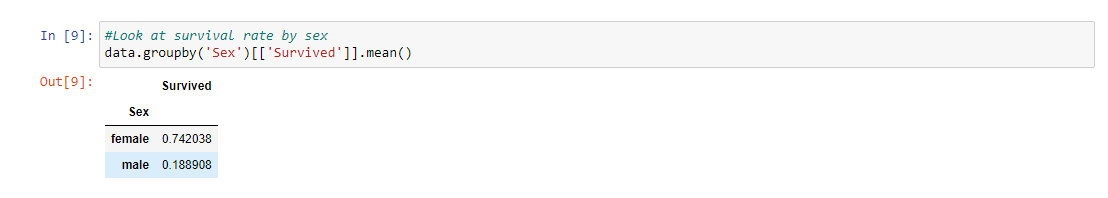
If you have 0 parents or children on board, you are not likely to survive according to the parch chart.



1. ***Pre-Processing Pipeline:***

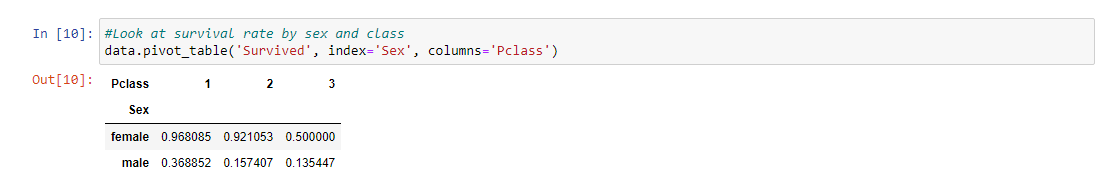
Next, we want to take a look at the survival rate by gender.

From the table below, we can see that about 74.2% of females survived and about 18.89% of males survived.



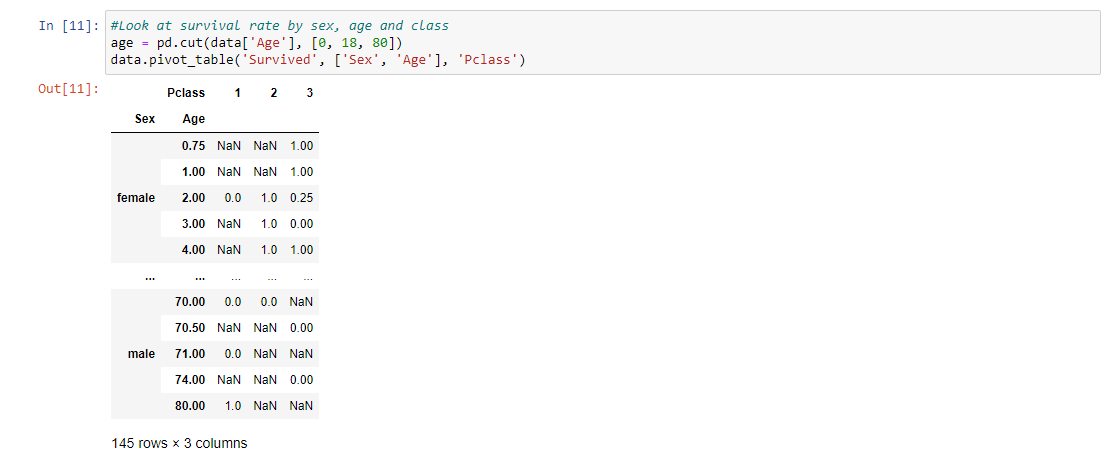
Look at the survival rate by gender and class.

From the pivot table below, we see that females in first class had a survival rate of about 96.8%, meaning the majority of them survived. Males in third class had the lowest survival rate at about 13.54%, meaning the majority of them did not survive.



Note that, in this data set, the oldest person is aged 80, so that will be our age limit.

We can see from the table below that women in first class that were 18 and older had the highest survival rate at 97.2973%, while men 18 and older in second class had the lowest survival rate of 7.1429%.



Check which columns contain empty values (NaN, NAN, na). Looks like columns age, embarked, deck, and embarked town are missing some values.

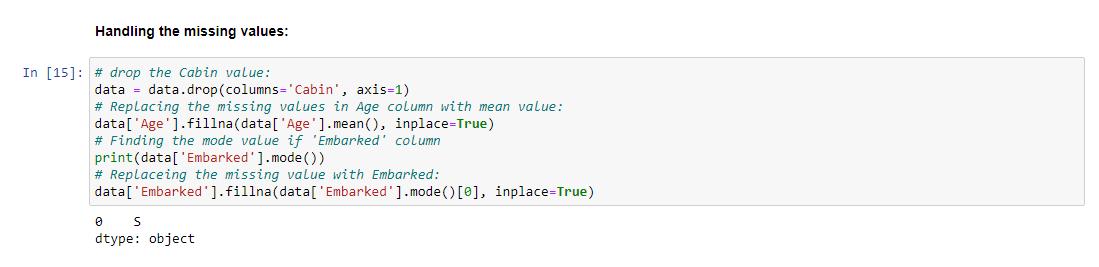
All the other columns are not missing any values.



Next, we will drop the redundant columns that are non-numerical and remove rows with missing values.

We also decided to drop the column called deck because it's missing 688 rows of data which means 688/891 = 77.22% of the data is missing for this column.

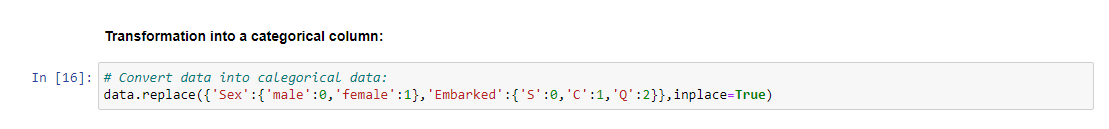




Now let us check if there are still any cells remaining empty.

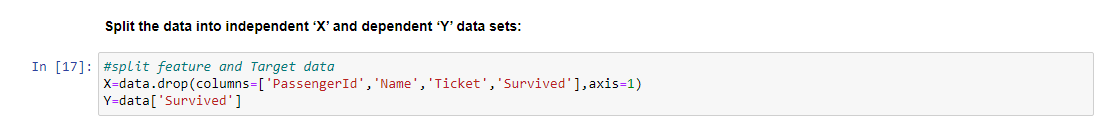
Running the isnull() command again, we get the satisfactory output, that no such empty cells are present.

We have already noticed from the table, there are two columns that contain string-type values: The "Sex" column and the "Berth" column.



Now if we run the titanic\_data.head() command again, we find that the values have been replaced successfully.

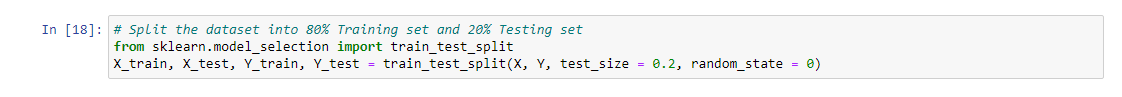
We also see, that there are few columns, which are not of much importance in this process. Let us get rid of them.



Here, X is the feature variable, containing all the features like Pclass, Age, Gender, Embarked, etc. excluding the Survived column.

Y, on the other hand, is the target variable, as that is the result that we want to determine, i.e, whether a person is alive.

Now, we will be splitting the data into four variables, namely, X\_train, Y\_train, X\_test, Y\_test.



Let's understand the variables :-

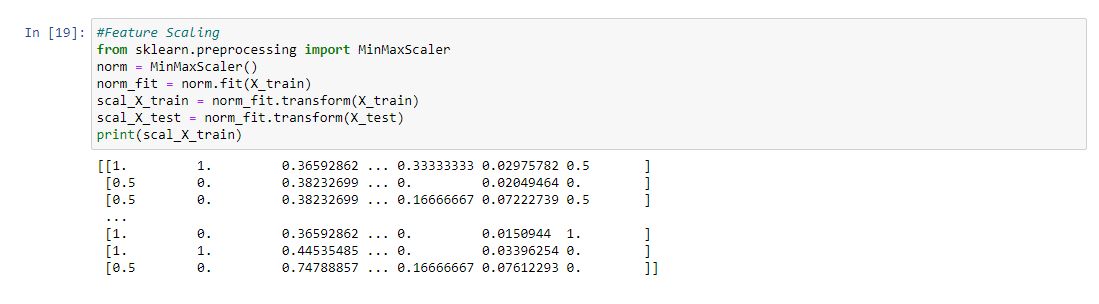
X\_train: contains a set of values from variable ' X '

Y\_train: contains the output (whether the person is alive or dead) of the corresponding value of X\_train.

X\_test: contains a set of values from variable ' X ', excluding the ones from X\_train.

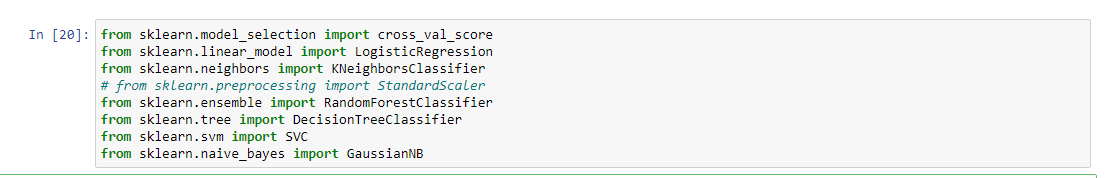
Y\_train: contains the output (whether the person is alive or dead) of the corresponding value of X\_test.

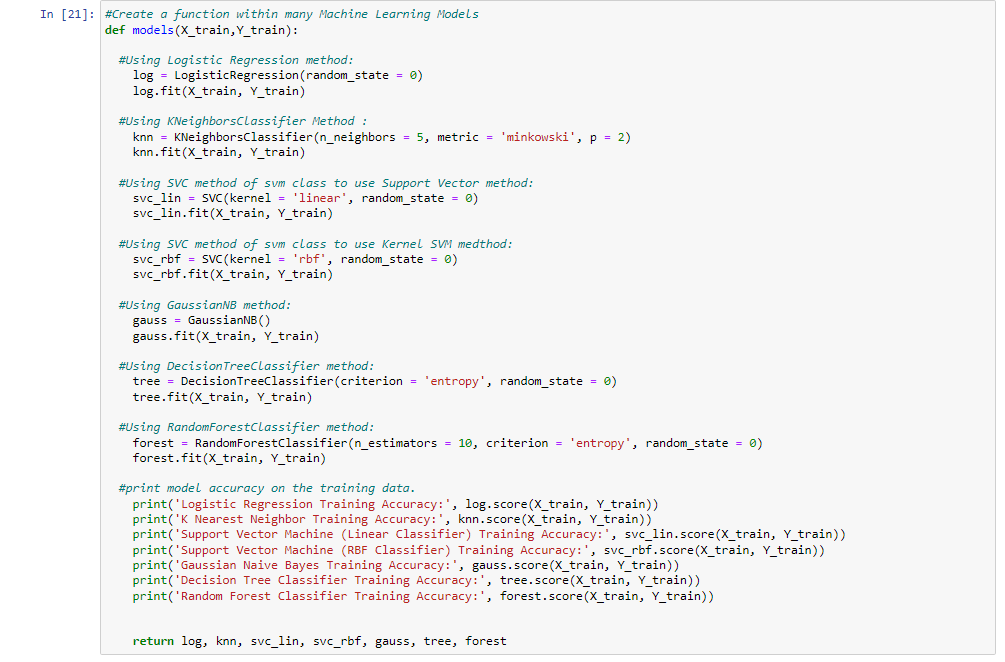
test\_size: represents the percentage ratio of X\_train:X\_test (Here 0.2 means that the data will be segregated in the X\_train and X\_test variables in a 80:20 ratio). You can use any value you want. A value <0.3 is preferred

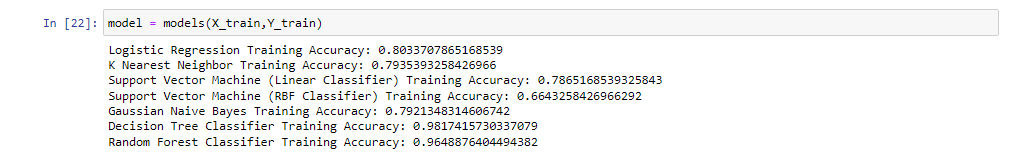


1. ***Building Machine Learning Models:***

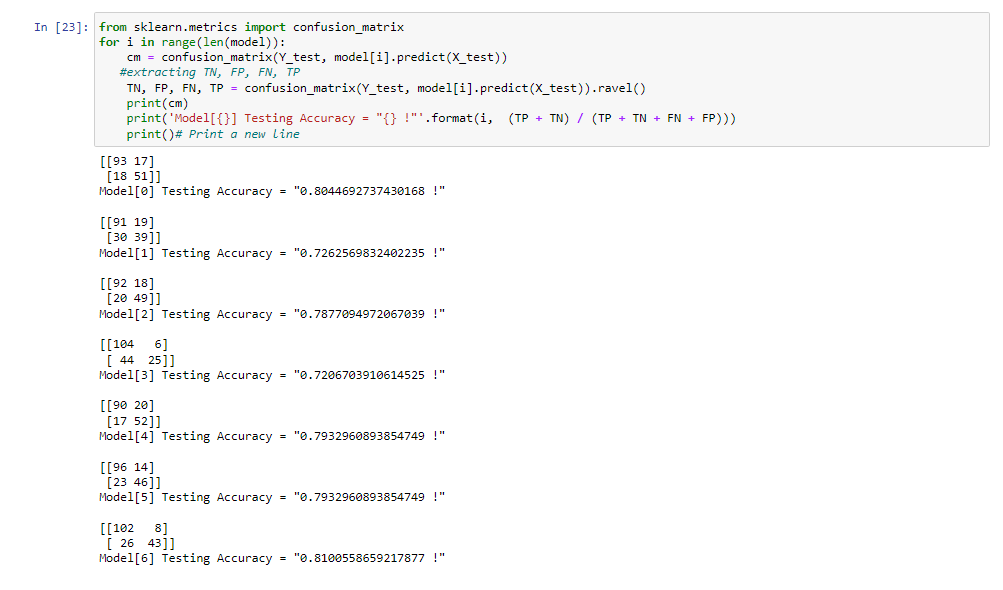
Create a function that has within it many different machine learning models that we can use to make our predictions.





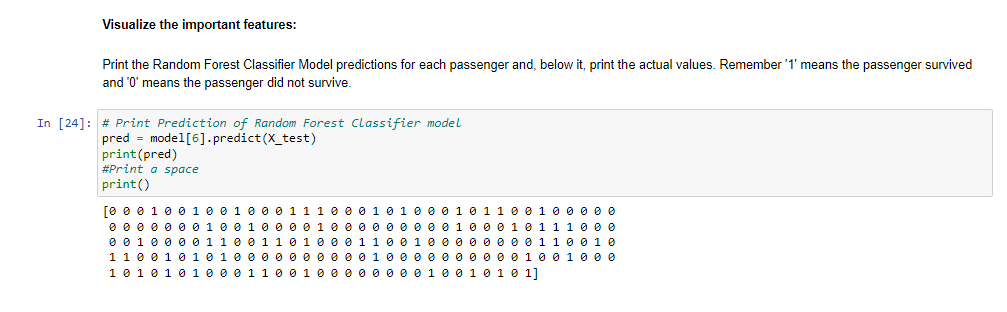


Show the confusion matrix and accuracy for all the models on the test data. The model that was most accurate on the test data is the model at position 0, which is the Logistic Regression Model with an accuracy of 81.11%, according to

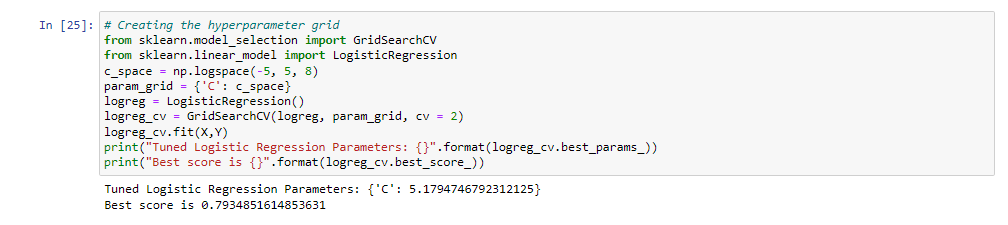


The model that I will use to predict if I would’ve survived, will be the model at position 6, the Random Forest Classifier.

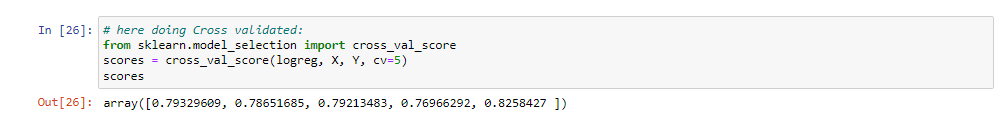
I chose that model because it did second-best on the training and testing data and has an accuracy of 80.41% on the testing data and 97.53% on the training data.



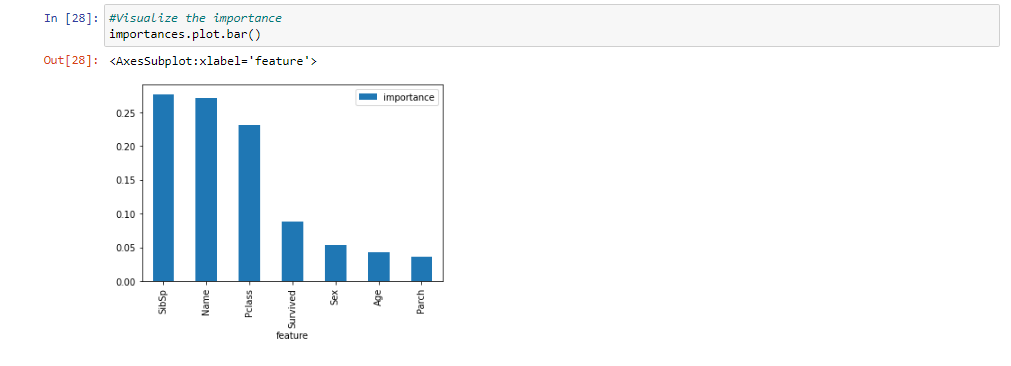
Now we see the hyperparamter tuning for the logreg\_cv.best\_params\_.

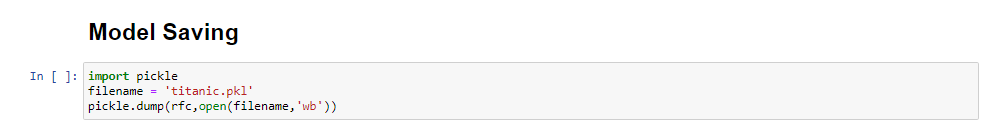


Show the Cross validated and, which is the Logistic Regression Model with Best score is 0.7934851614853631, according to









1. ***Concluding Remarks:***

We started with the data exploration where we got a feeling for the dataset, checked about missing data and learned which features are important.

During this process we used seaborn and matplotlib to do the visualizations.

During the data pre-processing part, we computed missing values, converted features into numeric ones, and created a few new features.

Afterwards we started training 8 different machine learning models, we looked into confusion matrix, picked one of them (Logistic Regression) and applied cross validation on it.

Then we discussed how random forest works, look at the importance it assigns to the different features and tuned into hyperparameter values.

Lastly, than Save the models and Visualize the importance datasets.