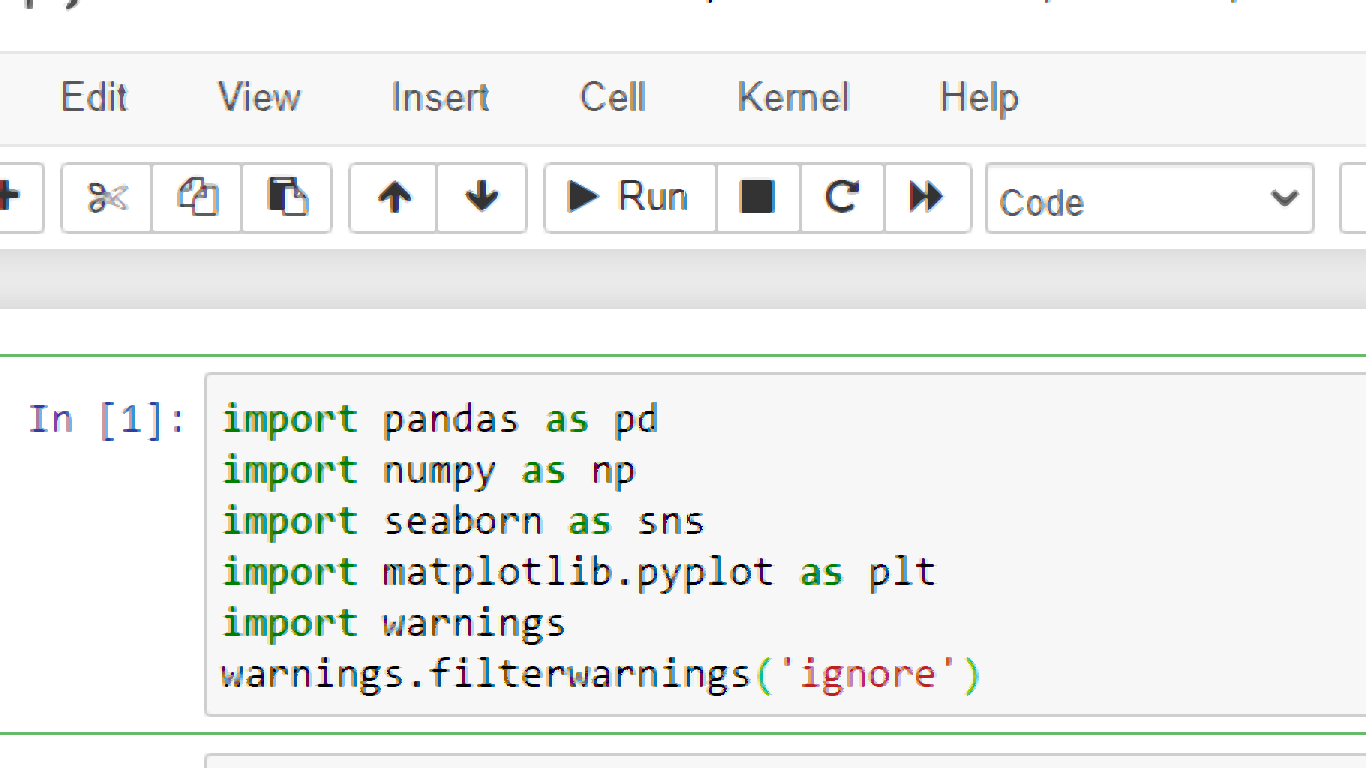
# **Predicting the Survival of Titanic Passengers**

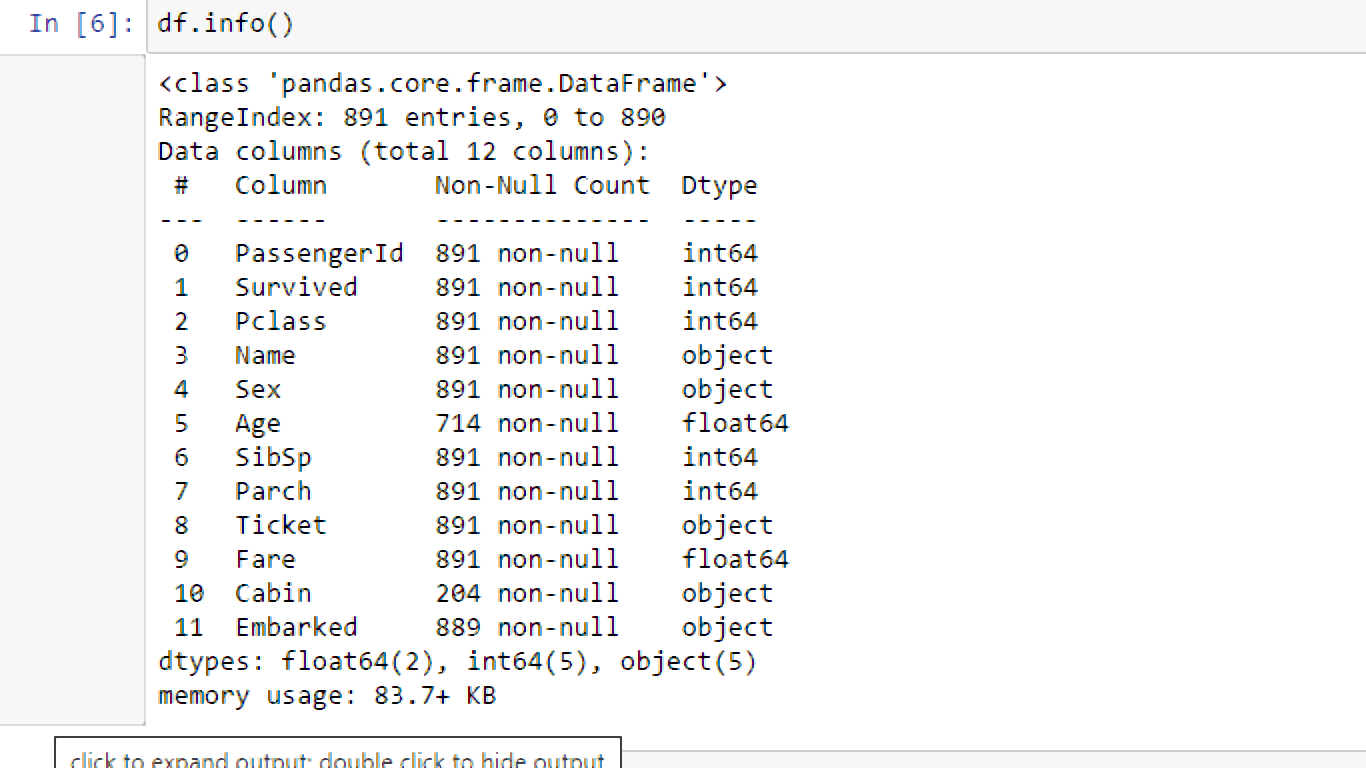
I will go through the whole process of creating a machine learning model on the famous Titanic dataset, which is used by many people all over the world. It provides information on the fate of passengers on the Titanic, summarized according to economic status (class), sex, age and survival.



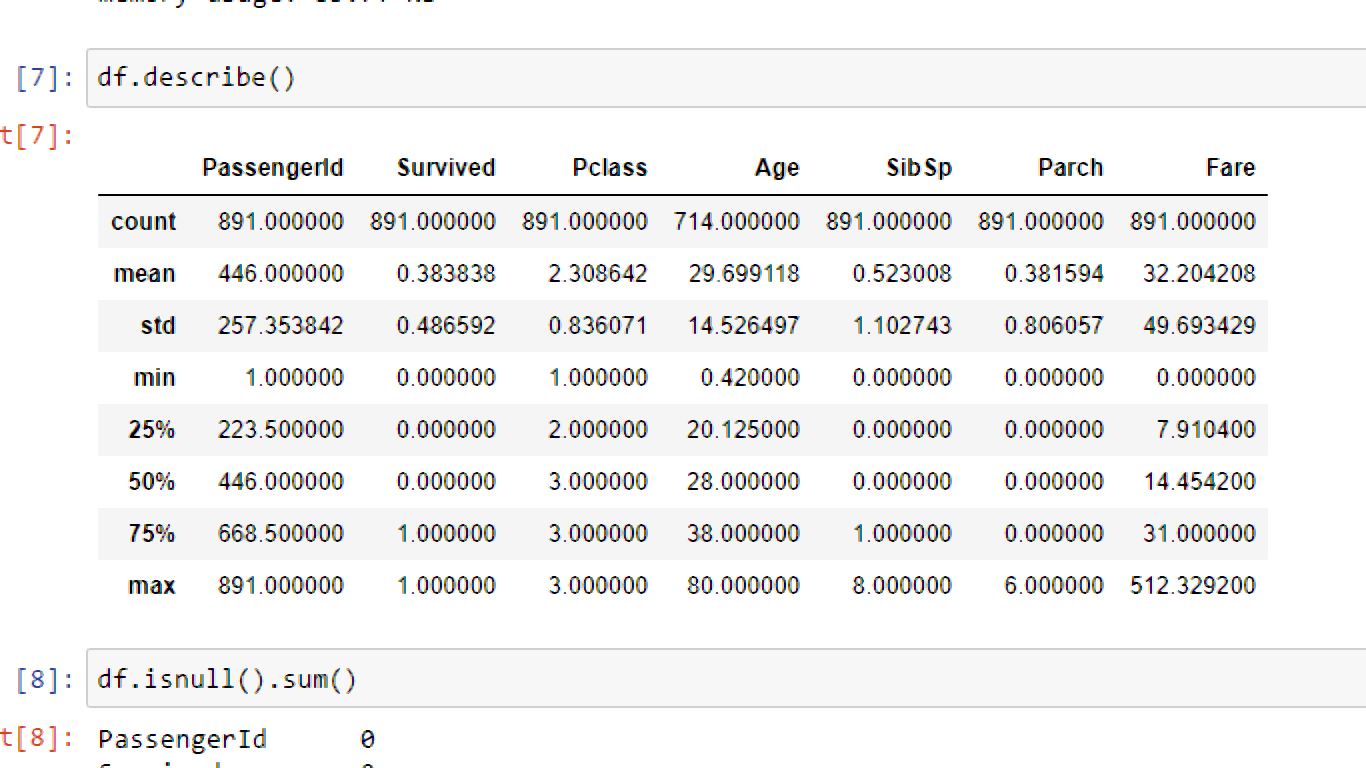
# Getting the Data



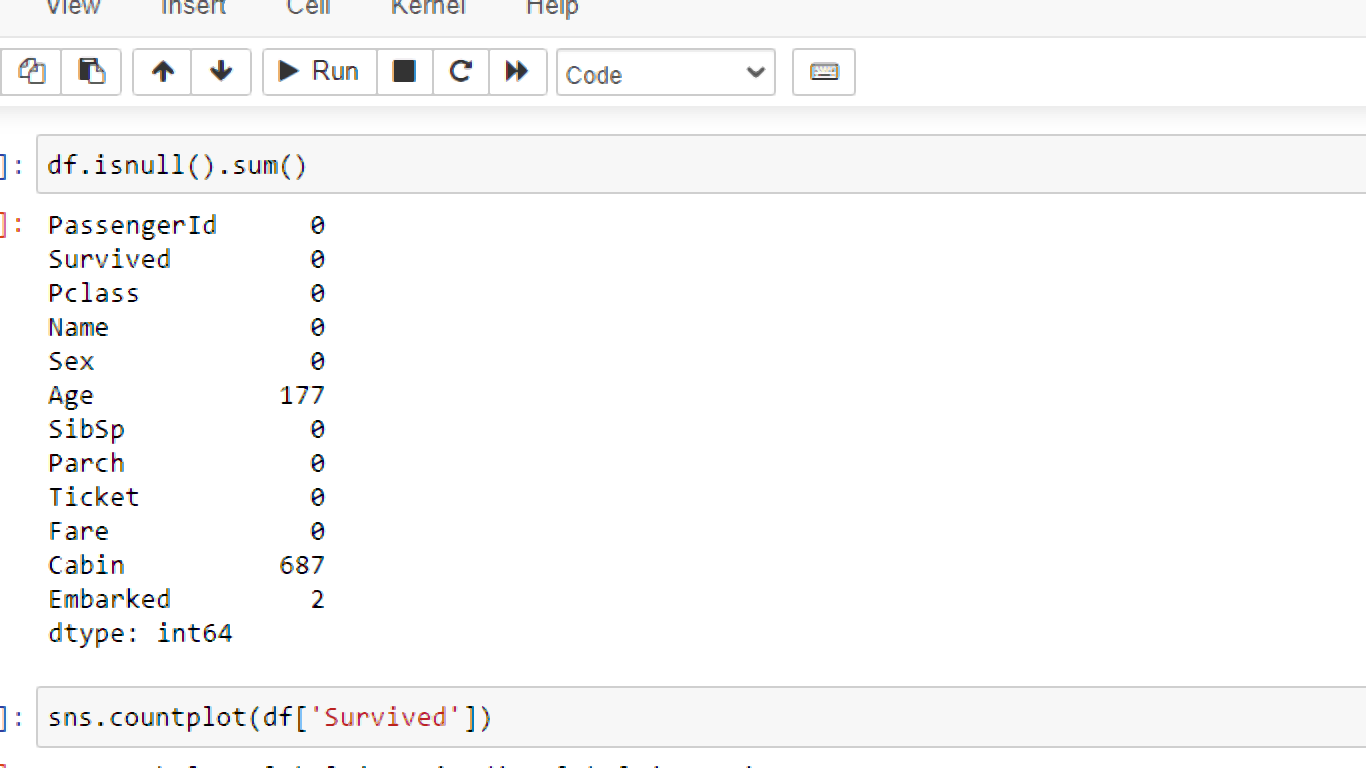
# Data Exploration/Analysis



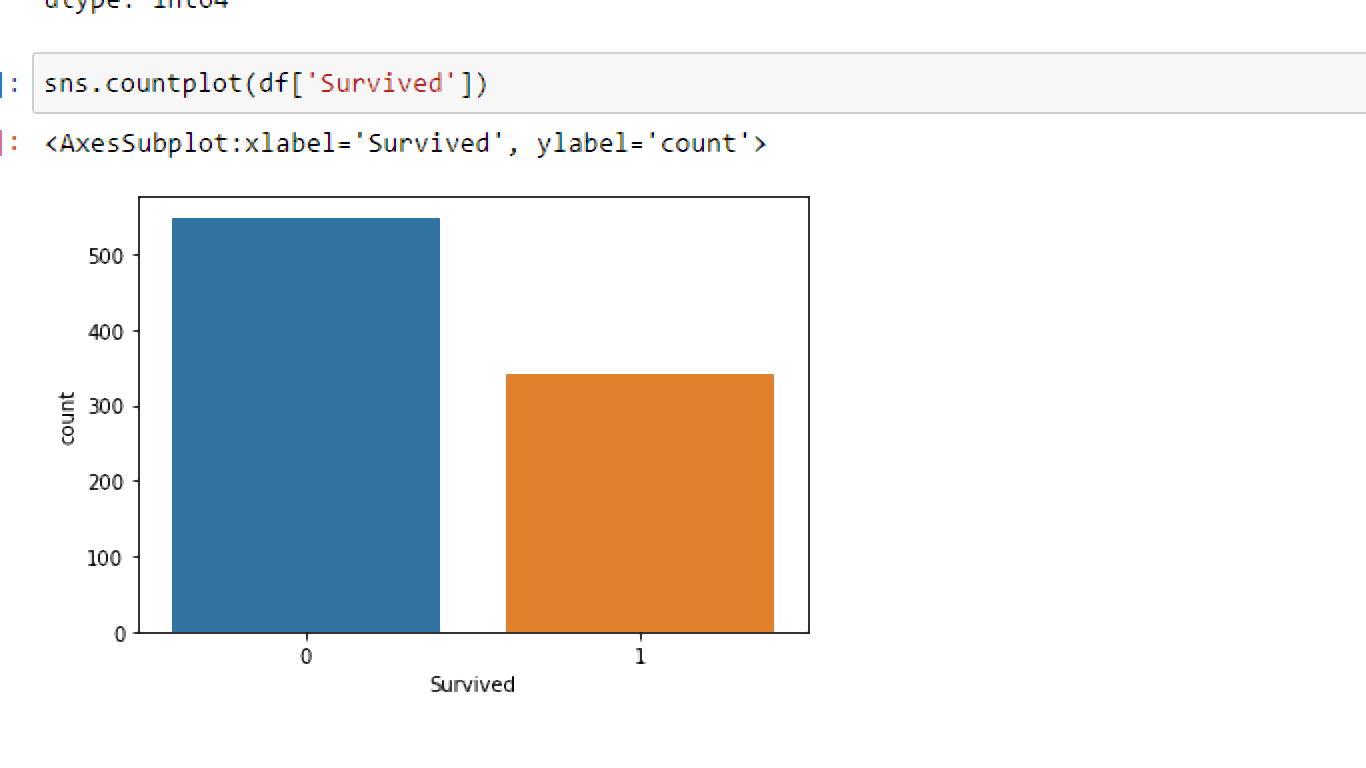
**The training-set has 891 examples and 11 features + the target variable (survived)**. 2 of the features are floats, 5 are integers and 5 are objects.



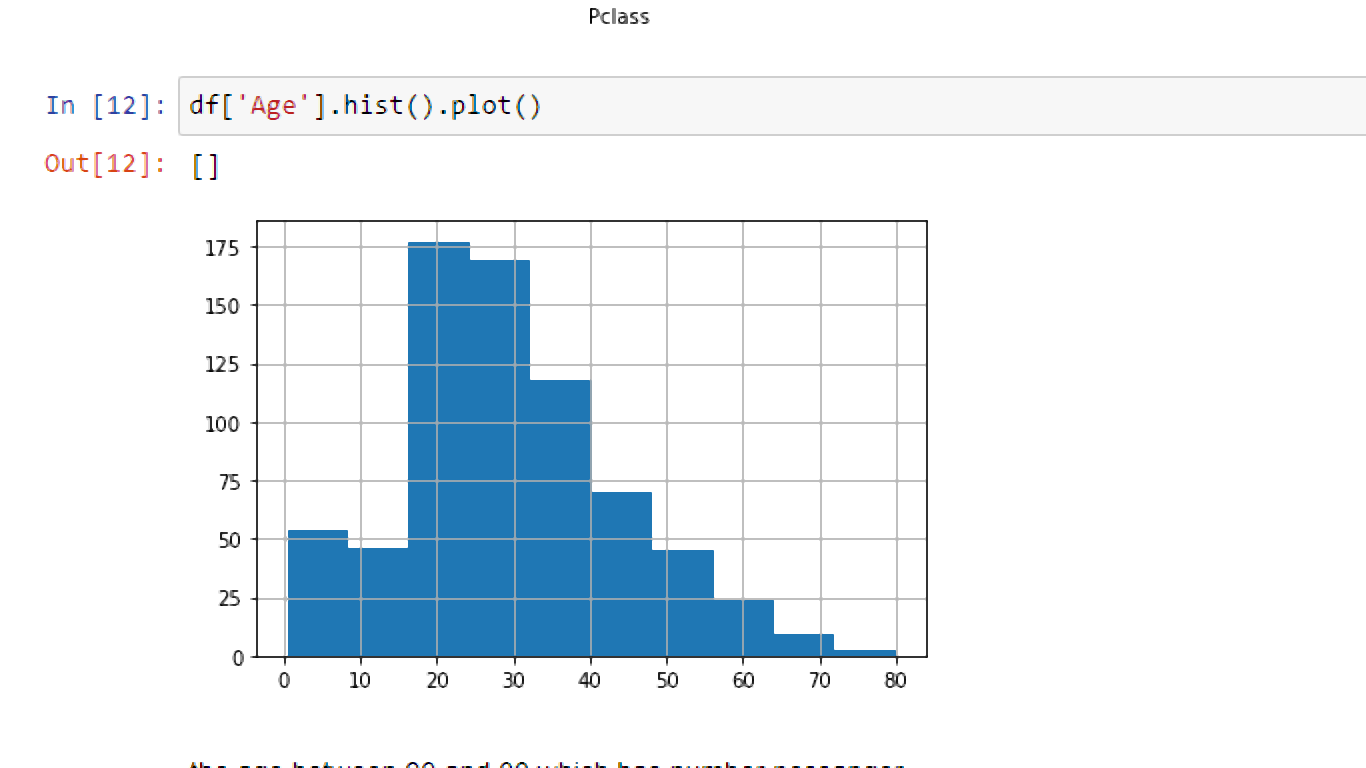
Above we can see that **38% out of the training-set survived the Titanic**. We can also see that the passenger ages range from 0.4 to 80. On top of that we can already detect some features, that contain missing values, like the ‘Age’ feature.



In the above ,we have used the isnull to find out the sum of null values , and also we have age which has the null value of 177,in cabin we have null value of 687.



o number of person has survived which has high value 1 number of person has survived which has low value.the pclass 3 which has the High value of near to 500.



the age between 20 and 30 which has number pessanger

sns.pairplot(df)

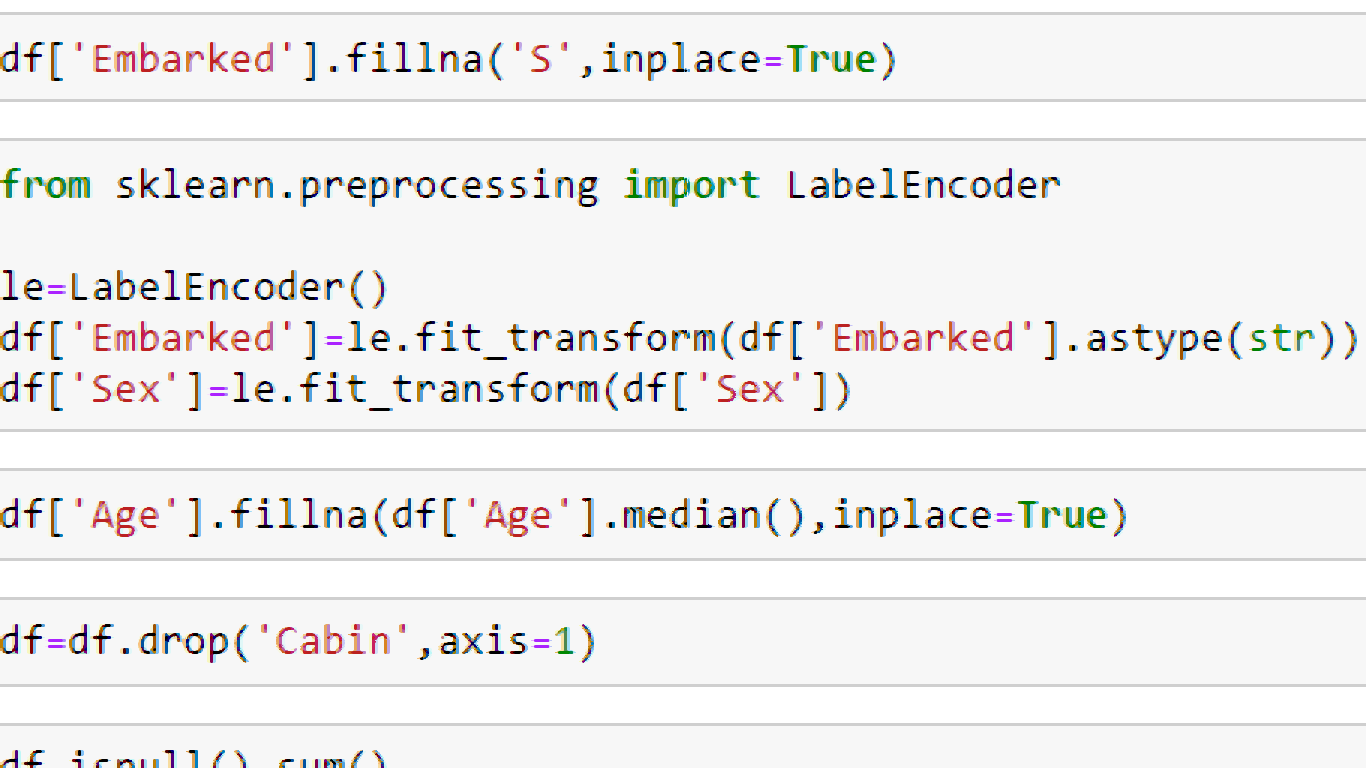
plt.show()

Passenger which is skew

Age which is left skewed

Parch which is left skewed

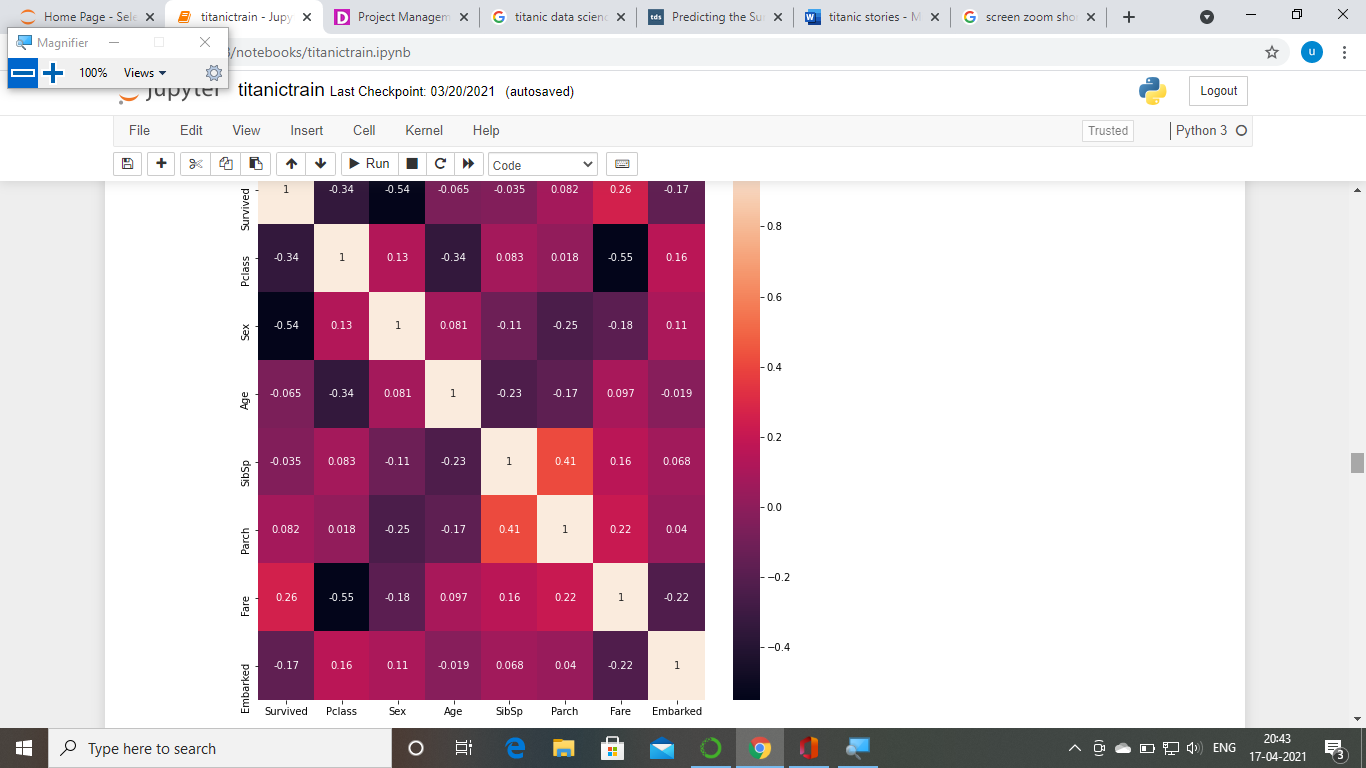
Fare which is left skewed



In the columns Embarked we have filled s using fillna,for understanding the machine learning we need to all value into numeric value so we have used label encoder to convert the Embarked,sex.

And also for we have used the median option to fullfill the Ahe column

We have droped the cabin column

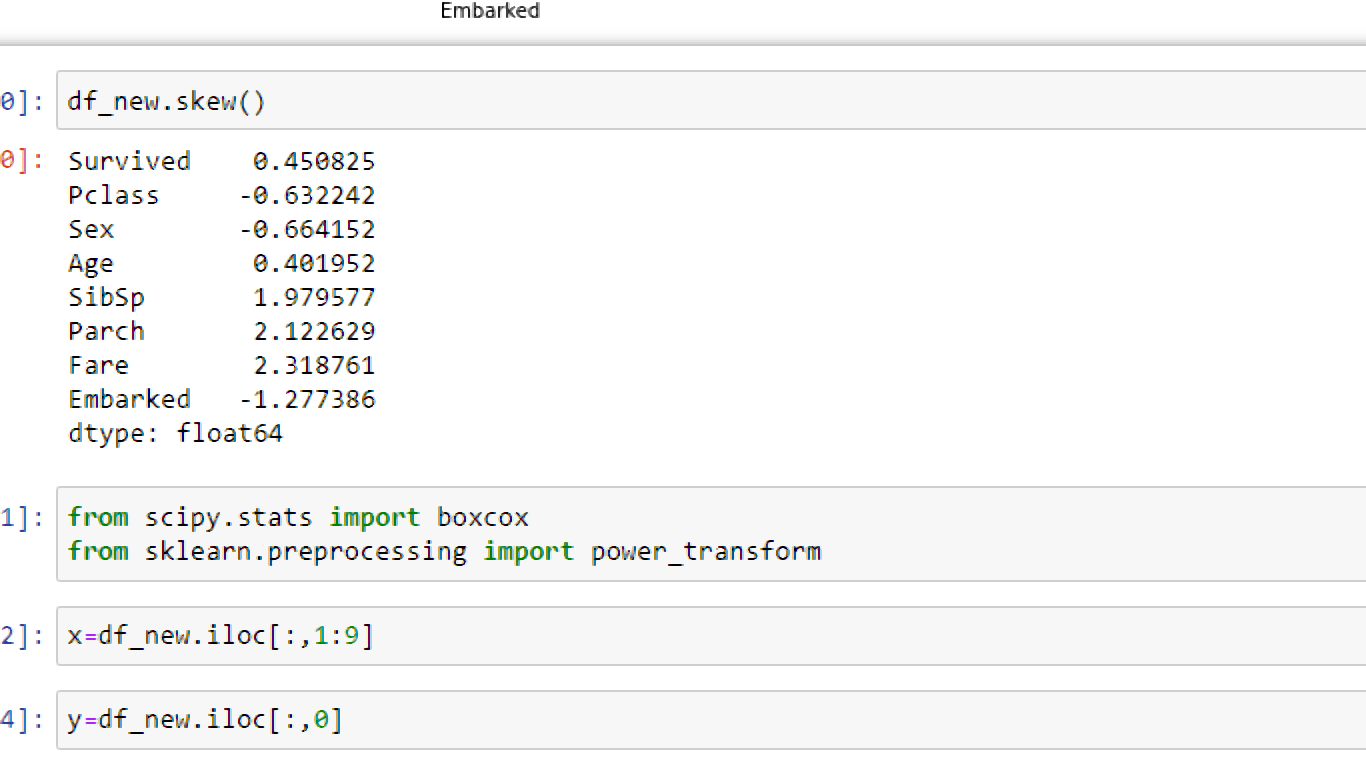


Fare which has the high correlated,sex which has the highly correlated,

df.plot(kind='box',subplots=True,layout=(5,15),figsize=(20,40))

By using the plot we have identified that the age and fare which has the outliers.remaining all columns which has no outliers

By using zscore we removed the outliers



We have used the skew to find out which one has highly skewned

Pclass,sex,Emarked which high value.

# Building Machine Learning Models

Now we will train several Machine Learning models and compare their results. Note that because the dataset does not provide labels for their testing-set, we need to use the predictions on the training set to compare the algorithms with each other. Later on, we will use cross validation.

**SVC** (Support Vector Classifier)

The objective of a Linear **SVC** (Support Vector Classifier) is to fit to the data you provide, returning a "best fit" hyperplane that divides, or categorizes, your data. From there, after getting the hyperplane, you can then feed some features to your classifier to see what the "predicted" class is.

accuracy score0.9939024390243902

**kNeighbourClassifier**

By default, the **KNeighborsClassifier** looks for the 5 nearest neighbors. We must explicitly tell the classifier to use Euclidean distance for determining the proximity between neighboring points

accuracy score0.9817073170731707

Gaussian Naive Bayes

A Gaussian Naive Bayes algorithm is a special type of NB algorithm. It's specifically used when the features have continuous values. It's also assumed that all the features are following a gaussian distribution i.e, normal distribution

accuracy score1.0

A Decision Tree

A Decision Tree is a simple representation for classifying examples. It is a Supervised Machine Learning where the data is continuously split according to a certain parameter.

accuracy score 1.0

**Random forest**

**Random forest** is a flexible, easy to use machine learning algorithm that produces, even without hyper-parameter tuning, a great result most of the time. It is also one of the most used algorithms, because of its simplicity and diversity (it can be used for both classification and regression tasks).

accuracy score 1.0

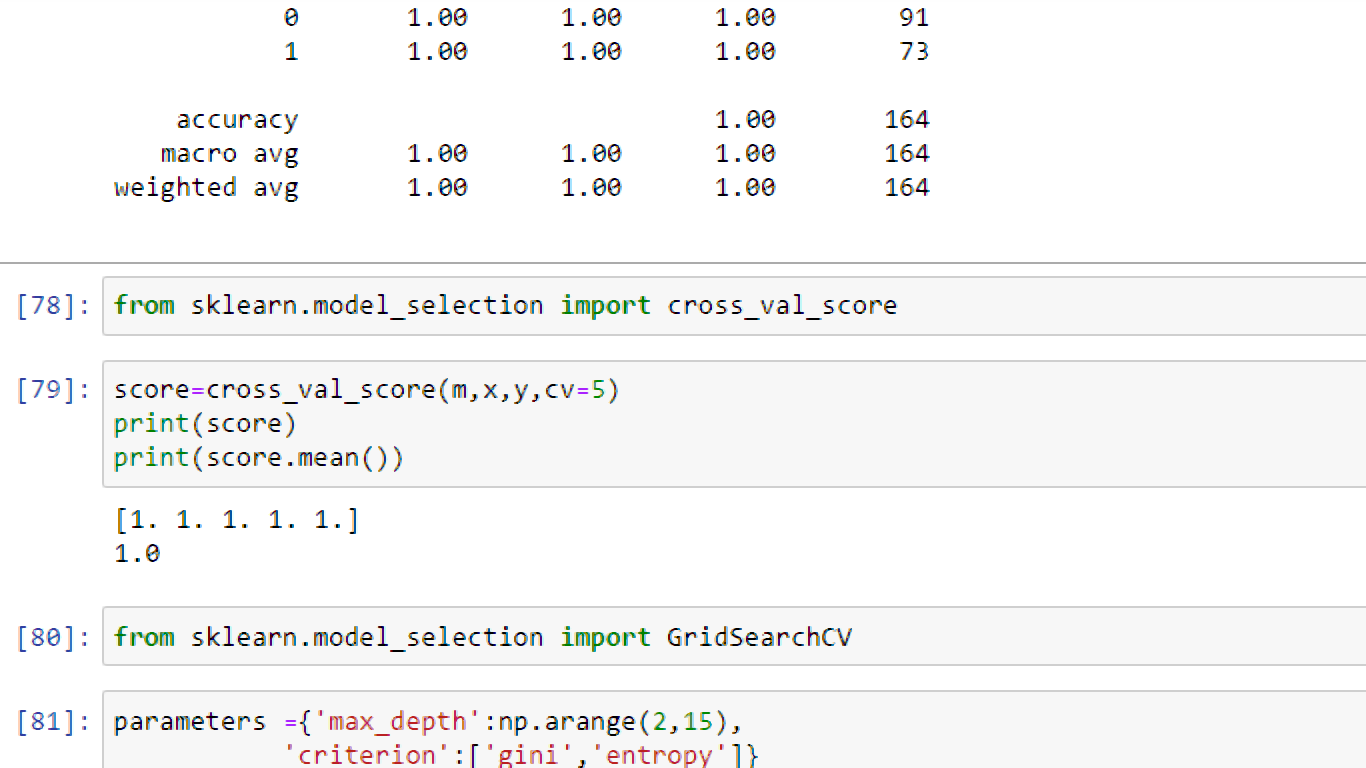
**AdaBoost**

**AdaBoost** is best used to boost the performance of decision trees on binary classification problems. Adaboost was originally called Adaboost. M1 by the authors of the technique Freund and Schapire. More recently it may be referred to as discrete Adaboost because it is used for classification rather than regression.

accuracy score 1.0

cross\_val\_score

cross\_val\_score returns **score** of test fold where cross\_val\_predict returns predicted y **values** for the test fold. For the cross\_val\_score() , you are using the average of the output, which will be affected by the number of folds because then it may have some folds which may have high error (not fit correctly).



svc=SVC()

knn= KNeighborsClassifier()

gnb = GaussianNB()

dtc = DecisionTreeClassifier()

rfc = RandomForestClassifier()

abc = AdaBoostClassifier()

The above all the classifier which has involved the cross value score

**GridSearchCV** is a function that comes in Scikit-learn's(or SK-learn) model\_selection package.So an important point here to note is that we need to have Scikit-learn library installed on the computer. This function helps to loop through predefined hyperparameters and fit your estimator (model) on your training set.

GCV.best\_params\_

{'criterion': 'gini', 'max\_depth': 2}

GCV\_pred**=**GCV.best\_estimator\_.predict(x\_test)

accuracy\_score(y\_test,GCV\_pred)

1.0