**MACHINE LOGISTIC PROJECT**

In this project, I have used Logistic Regression Machine Learning model for the iris dataset from Kaggle website

**PROBLEM STATEMENT:**

The Iris flower data set or Fisher's Iris data set is a multivariate data set introduced by the British statistician and biologist Ronald Fisher in his 1936 paper The use of multiple measurements in taxonomic problems as an example of linear discriminant analysis. It is sometimes called Anderson's Iris data set because Edgar Anderson collected the data to quantify the morphologic variation of Iris flowers of three related species. Two of the three species were collected in the Gaspé Peninsula "all from the same pasture, and picked on the same day and measured at the same time by the same person with the same apparatus".

**ML METHODOLOGY:**

Logistic regression is a statistical method for analyzing a dataset in which there are one or more independent variables that determine an outcome. The outcome is measured with a dichotomous variable (in which there are only two possible outcomes). It is another technique borrowed by machine learning from the field of statistics. It is the go-to method for binary classification problems (problems with two class values).Techniques used to learn the coefficients of a logistic regression model from data

**DATASET DESCRIPTION:**

Some relevant columns in the dataset

* Sepal length(cm)
* Sepal width(cm)
* petal length(cm)
* petal width(cm)
* Flower\_type

**PRE\_PROCESSING:**

Pre\_processing refers to the transformations applied to our data before feeding it to the algorithm.

%matplotlib inline

import matplotlib.pyplot as plt

import numpy as np

import pandas as pd

from sklearn.model\_selection import train\_test\_split

from sklearn import datasets

from sklearn.metrics import mean\_squared\_error,accuracy\_score,classification\_report

from sklearn.preprocessing import StandardScaler

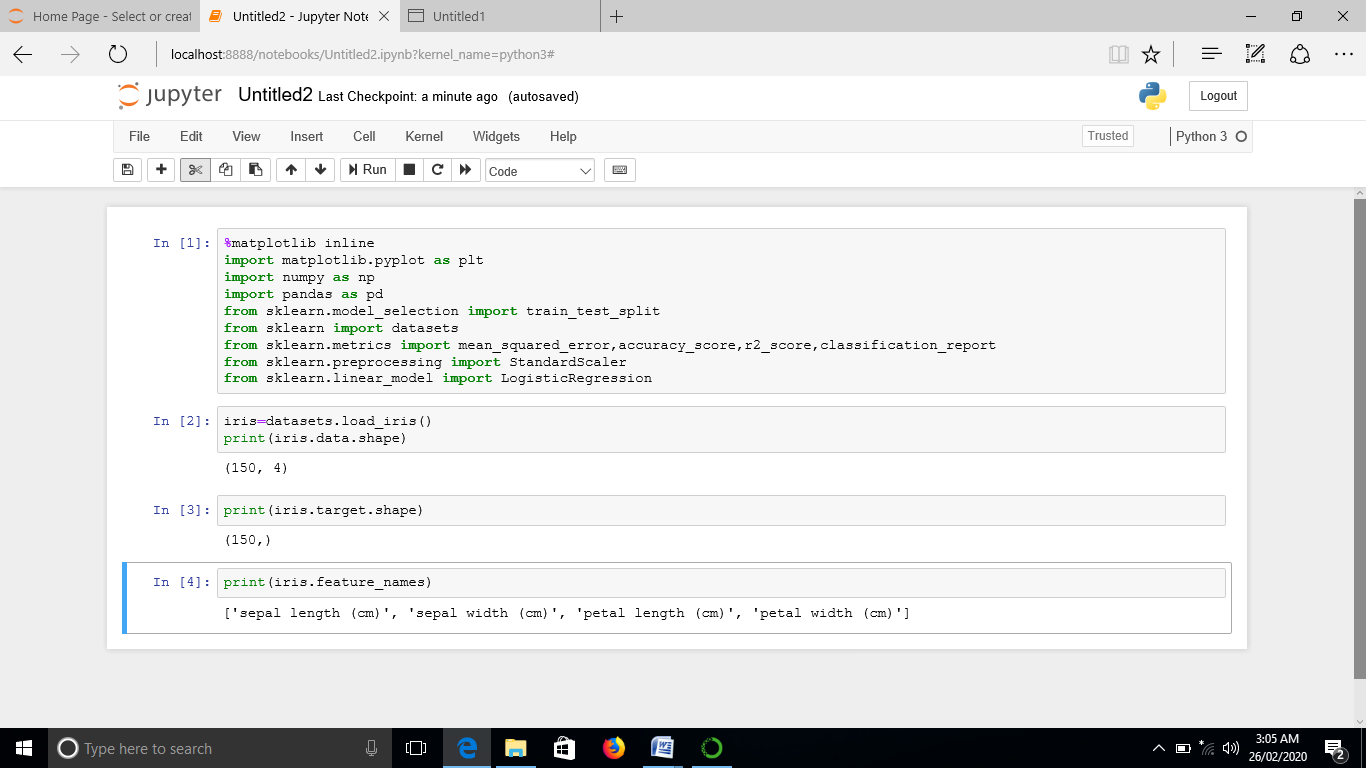
from sklearn.linear\_model import LogisticRegression

iris=datasets.load\_iris()

print(iris.data.shape)

print(iris.target.shape)

print(iris.feature\_names)



print(iris.target)

print(‘Target: 1:{},2:{},3:{}'.format(np.sum(iris.target==0),np.sum(iris.target==1),

np.sum(iris.target==2)))

df=pd.DataFrame(iris.data)

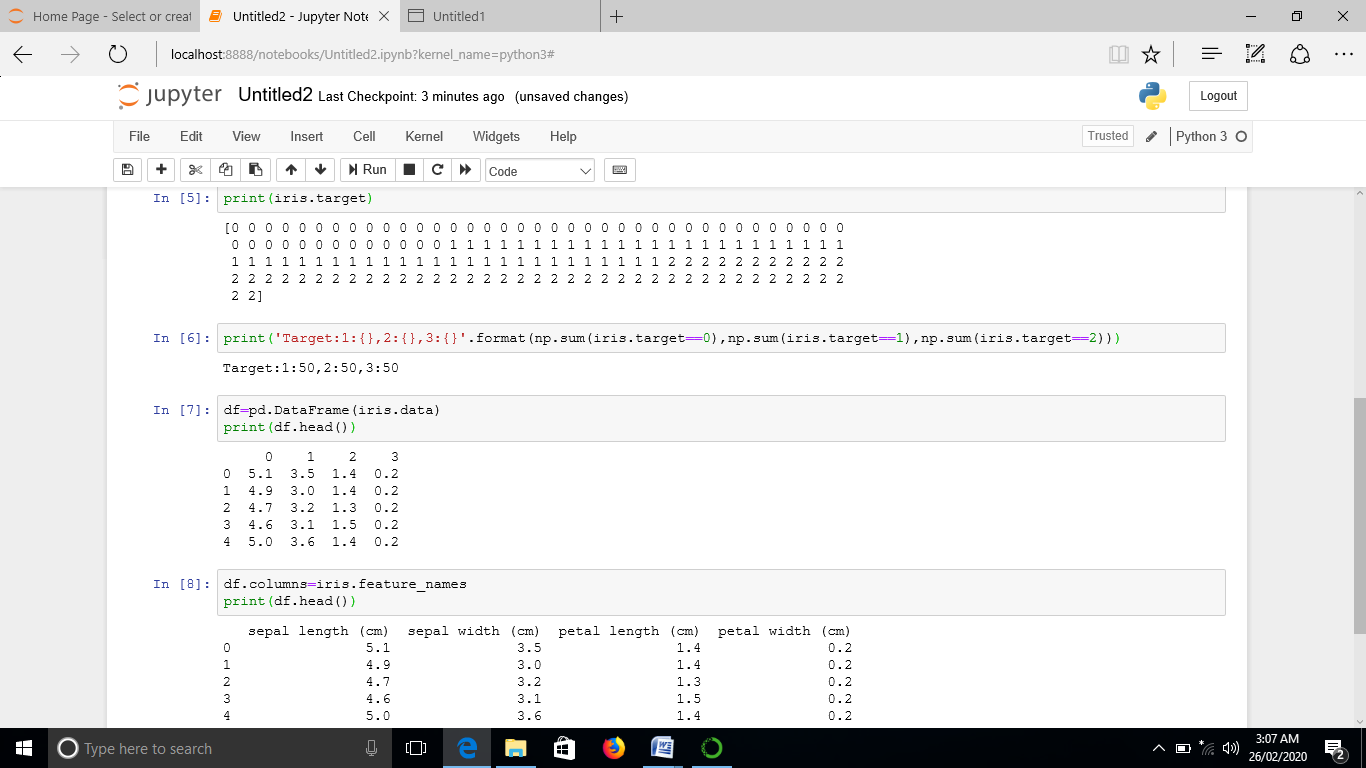
print(df.head())

df.columns=iris.feature\_names

print(df.head())

df['flowe\_type']=iris.target

print(df.head(3))



df['flower\_type']=iris.target

print(df.head(3))

X\_train,X\_test,y\_train,y\_test=train\_test\_split(

iris.data,iris.target,test\_size=0.25,random\_state=42)

scaler=StandardScaler()

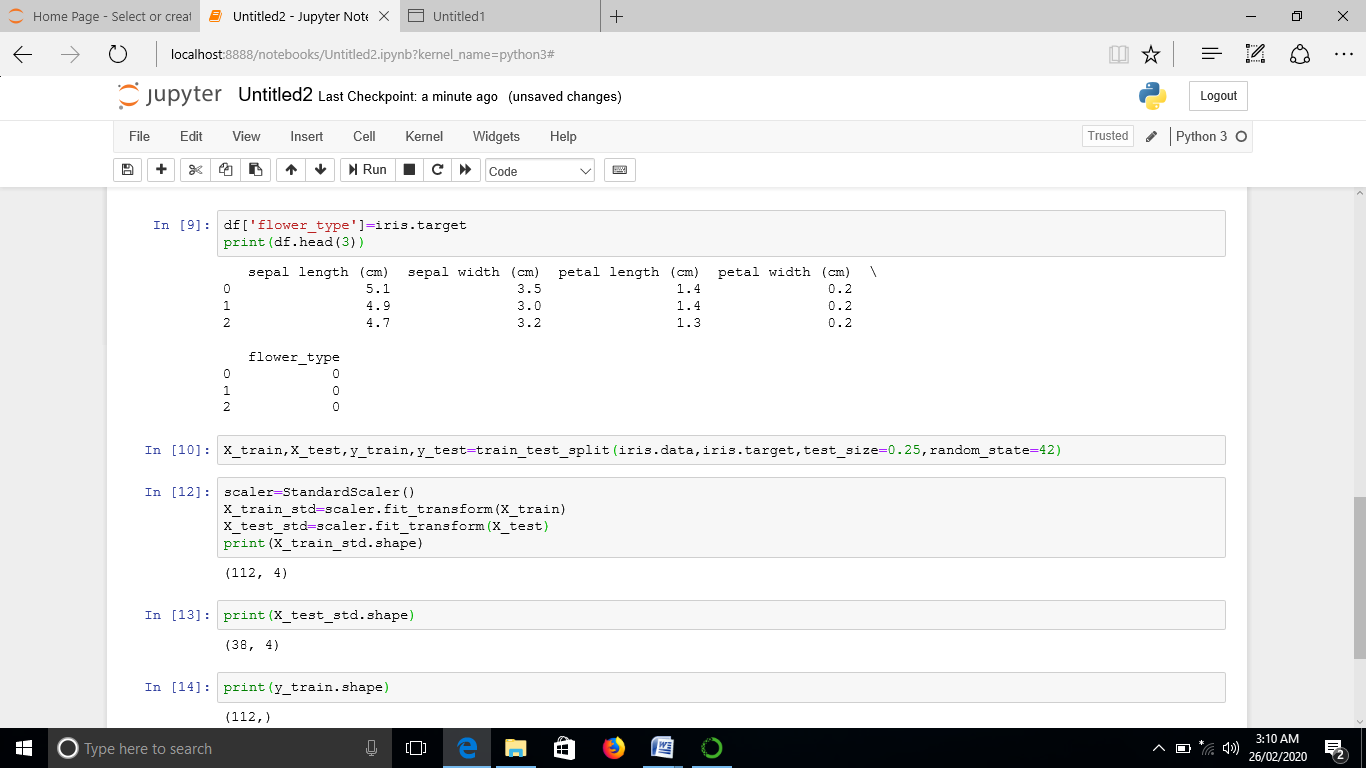
X\_train\_std=scaler.fit\_transform(X\_train)

X\_test\_std=scaler.fit\_transform(X\_test)

print(x\_train\_std.shape)

print(x\_test\_std.shape)

print(y\_train.shape)



print(y\_test.shape)

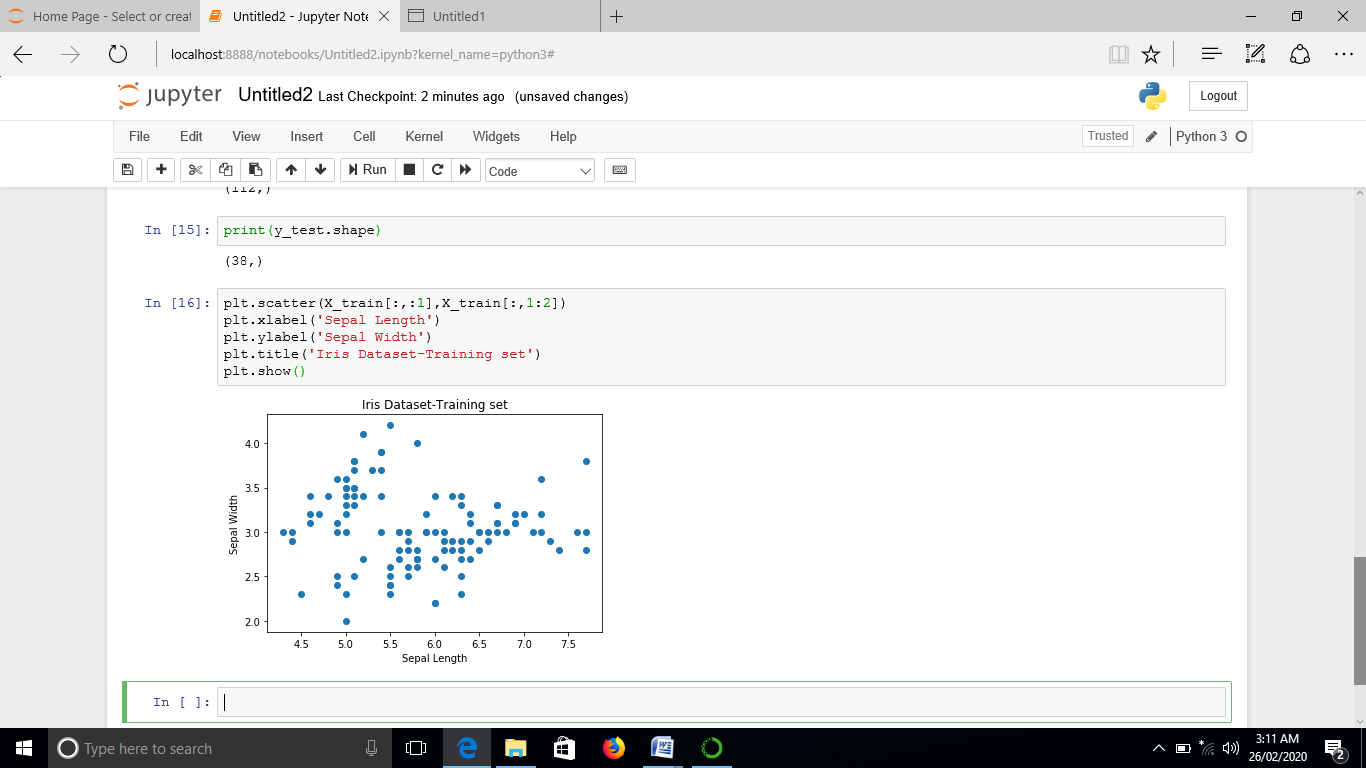
plt.scatter(X\_train[:,:1],X\_train[:,1:2])

plt.xlabel('Sepal Length')

plt.ylabel('Sepal width')

plt.title('Iris Dataset-Training set')

plt.show()

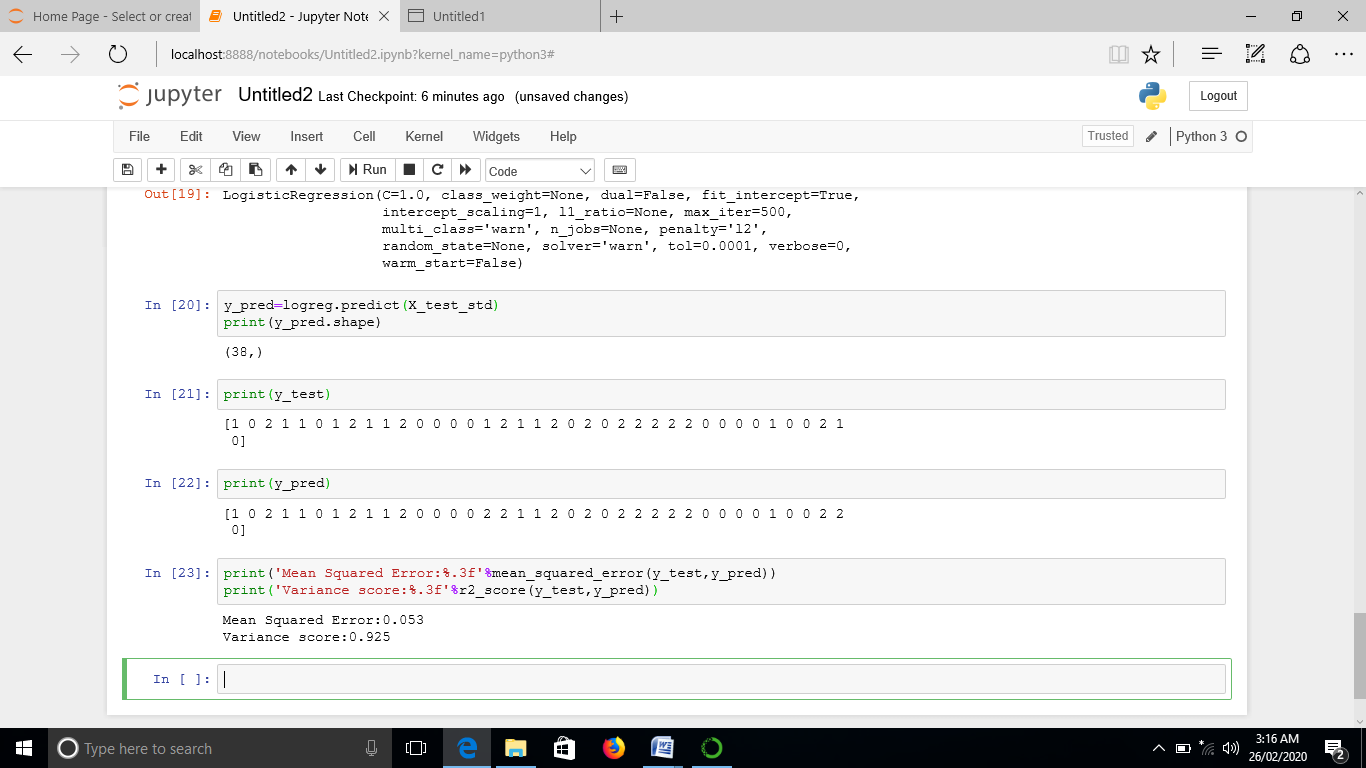


logreg=LogisticRegression(max\_iter=500)

logreg.fit(X\_train\_std,y\_train)

Y\_pred=logreg.predict(X\_test\_std)

Print(y\_pred.shape)

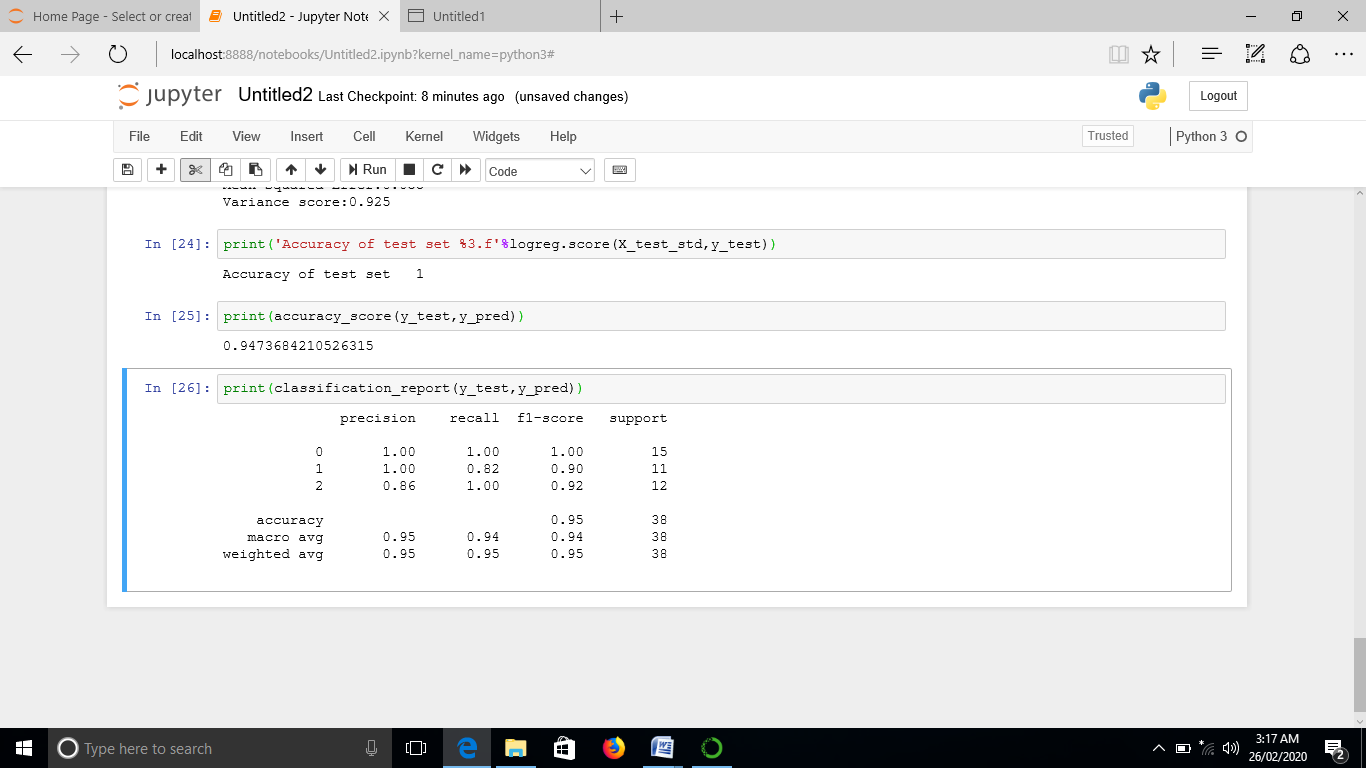


print(y\_pred)

print(‘Mean squared Error:%.3f’%mean\_squared\_error(y\_test,y\_pred))

print(‘Variance score:%.3f’%r2\_score(y\_test,y\_pred))

print(classification\_report(y\_test,y\_pred))



**CONCLUSION:**

Model weights of datasets are calculated by using logistic regression and calculate the training and testing dataset.