lab 5

#classification using logistic regression

import numpy as np

import pandas as pd

import matplotlib.pyplot as plt

import seaborn as sns

from sklearn.datasets import load\_iris

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

from sklearn.linear\_model import LogisticRegression

from sklearn.metrics import confusion\_matrix,accuracy\_score,classification\_report

from sklearn.preprocessing import StandardScaler #for standarization

#load dataset

dataset=load\_iris()

x = dataset.data

y = dataset.target

# standarzation dataset

sc = StandardScaler()

x = sc.fit\_transform(x)

# train test split

x\_train,x\_test,y\_train,y\_test = train\_test\_split(x,y,test\_size=0.2,random\_state=0)

#create logistic model

model = LogisticRegression(max\_iter=1000)

model.fit(x\_train,y\_train)

# predict the value

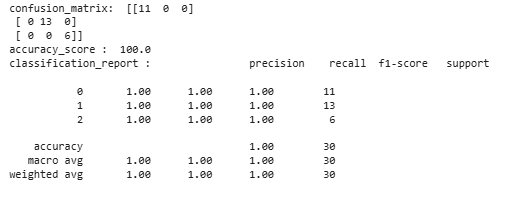
y\_pred = model.predict(x\_test)

# evaluation metrics

print("confusion\_matrix: ",confusion\_matrix(y\_test,y\_pred))

print("accuracy\_score : ",accuracy\_score(y\_test,y\_pred)\*100) # ACCURACY OUT OF 100 %

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



# Visualize OF confusion MATRIX

sns.heatmap(confusion\_matrix(y\_test,y\_pred),annot=True,fmt='d',xticklabels=dataset.target\_names,yticklabels=dataset.target\_names )

plt.title("Confusion Matrix")

plt.xlabel("Predicted")

plt.ylabel("Actual")

plt.show()

