**LOGISTIC REGRESSION (IRIS dataset)**

***(results have been compared with SVM)***

import pandas as pd

import matplotlib.pyplot as plt

import numpy as np

df = pd.read\_csv('C:/Users/tanis/Downloads/ML\_intern\_resources/3.logistic\_regression/Iris.csv')

df.head()

df['Species'].unique()

df['Species'] = df['Species'].replace({'Iris-setosa':1, 'Iris-versicolor':2, 'Iris-virginica':3})

df.head()

x = df.iloc[:, :-1].values

y = df.iloc[:, -1].values

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

x\_train, x\_test, y\_train, y\_test = train\_test\_split(x, y, test\_size = 0.1, random\_state=0)

from sklearn.linear\_model import LogisticRegression

regressor = LogisticRegression(max\_iter = 500)

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

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

print(np.concatenate((y\_test.reshape(len(y\_test),1),y\_pred.reshape(len(y\_pred),1)),1))

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

cm = confusion\_matrix(y\_test, y\_pred)

cm

array([[3, 0, 0],

[0, 8, 0],

[0, 0, 4]], dtype=int64)

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

1.0

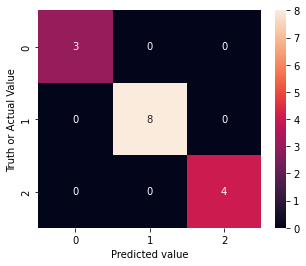
import seaborn as sn

plt.figure(figsize=(5,4))

sn.heatmap(cm, annot=True)

plt.xlabel('Predicted value')

plt.ylabel('Truth or Actual Value')



**USING SVM**

from sklearn.svm import SVC

classifier = SVC(kernel='rbf', random\_state=0)

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

y\_pred\_2 = classifier.predict(x\_test)

print(np.concatenate((y\_test.reshape(len(y\_test),1),y\_pred\_2.reshape(len(y\_pred\_2),1)),1))

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

cm\_2 = confusion\_matrix(y\_test, y\_pred\_2)

cm\_2

array([[3, 0, 0],

[0, 8, 0],

[0, 1, 3]], dtype=int64)

accuracy\_score(y\_test, y\_pred\_2)

0.9333333333333333

import seaborn as sn

plt.figure(figsize=(5,4))

sn.heatmap(cm\_2, annot=True)

plt.xlabel('Predicted value')

plt.ylabel('Truth or Actual Value')

