

## **EXPERIMENT 7**

**ADITHIYA V**

**241501010**

### **A PYTHON PROGRAM TO IMPLEMENT DECISION TREE**

#### **AIM:**

*TO IMPLEMENT A PYTHON PROGRAM TO DO WITH DECISION TREE*

#### **CODE:**

```
import numpy as np
import pandas as pd
from sklearn import svm
import matplotlib.pyplot as plt
import seaborn as sns; sns.set(font_scale=1.2)
recipes=pd.read_csv('/content/recipes_muffins_cupcakes.csv')
recipes.head()
recipes.shape

sns.lmplot(x='Sugar',y='Flour',data=recipes,hue='Type',palette='Set1',fit_reg
=False,scatter_kws={"s":70})

import matplotlib.pyplot as plt
import seaborn as sns; sns.set(font_scale=1.2)
import pandas as pd
import numpy as np
from sklearn import svm

recipes=pd.read_csv('/content/recipes_muffins_cupcakes.csv')

sugar_butter=recipes[['Sugar','Flour']].values
type_label=np.where(recipes['Type']=='Muffin',0,1)
model=svm.SVC(kernel='linear')
```

```

model.fit(sugar_butter,type_label)
w=model.coef_[0] #seperating the hyperplane
a = -w[0]/w[1] # calculate a
xx=np.linspace(5,30)
yy=a*xx-(model.intercept_[0]/w[1])
b=model.support_vectors_[0] #plot to sepearte hyperplane that pass
yy_down=a*xx+(b[1]-a*b[0])
b=model.support_vectors_[-1]
yy_up=a*xx+(b[1]-a*b[0])
sns.lmplot(x='Sugar',y='Flour',data=recipes,hue='Type',palette='Set1',fit_reg
=False,scatter_kws={"s":70})
plt.plot(xx,yy,linewidth=2,color='black')
plt.plot(xx, yy_down, 'k--')
plt.plot(xx, yy_up, 'k--')
plt.scatter(model.support_vectors_[:, 0], model.support_vectors_[:, 1],
s=100, facecolors='none', edgecolors='k')
plt.show()

```

```

scatter_kws={"s":70}
plt.plot(xx,yy,linewidth=2,color='black')
sns.lmplot(x='Sugar',y='Flour',data=recipes,hue='Type',palette='Set1',fit_reg
=False,scatter_kws={"s":70})
plt.plot(xx,yy,linewidth=2,color='black')
plt.plot(xx,yy_down,'k--')
plt.plot(xx,yy_up,'k--')
plt.scatter(model.support_vectors_[:,0],model.support_vectors_[:,-
1],s=80,facecolor='none')

```

```

from sklearn.metrics import confusion_matrix
from sklearn.model_selection import train_test_split
from sklearn.metrics import classification_report

x_train,x_test,y_train,y_test =
train_test_split(sugar_butter,type_label,test_size=0.2)
model1=svm.SVC(kernel='linear')

```

```

model1.fit(x_train,y_train)
pred = model1.predict(x_test)
print(pred)

print(confusion_matrix(y_test,pred))

print(classification_report(y_test,pred))

```

## OUTPUT:

```

IDLE Shell 3.12.3
File Edit Shell Debug Options Window Help
Python 3.12.3 (tags/v3.12.3:f6650f9, Apr 9 2024, 14:05:25) [MSC v.1938 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:/Users/itzdi/OneDrive/Documents/ML_Codes/Exp_6.py
[[0 0 0 1]
 [1 0]
 [0 1]]
      precision    recall   f1-score   support
      0       1.00     1.00     1.00      3
      1       1.00     1.00     1.00      1

      accuracy          1.00      4
      macro avg       1.00     1.00     1.00      4
weighted avg       1.00     1.00     1.00      4
>>> |

```

Figure 2

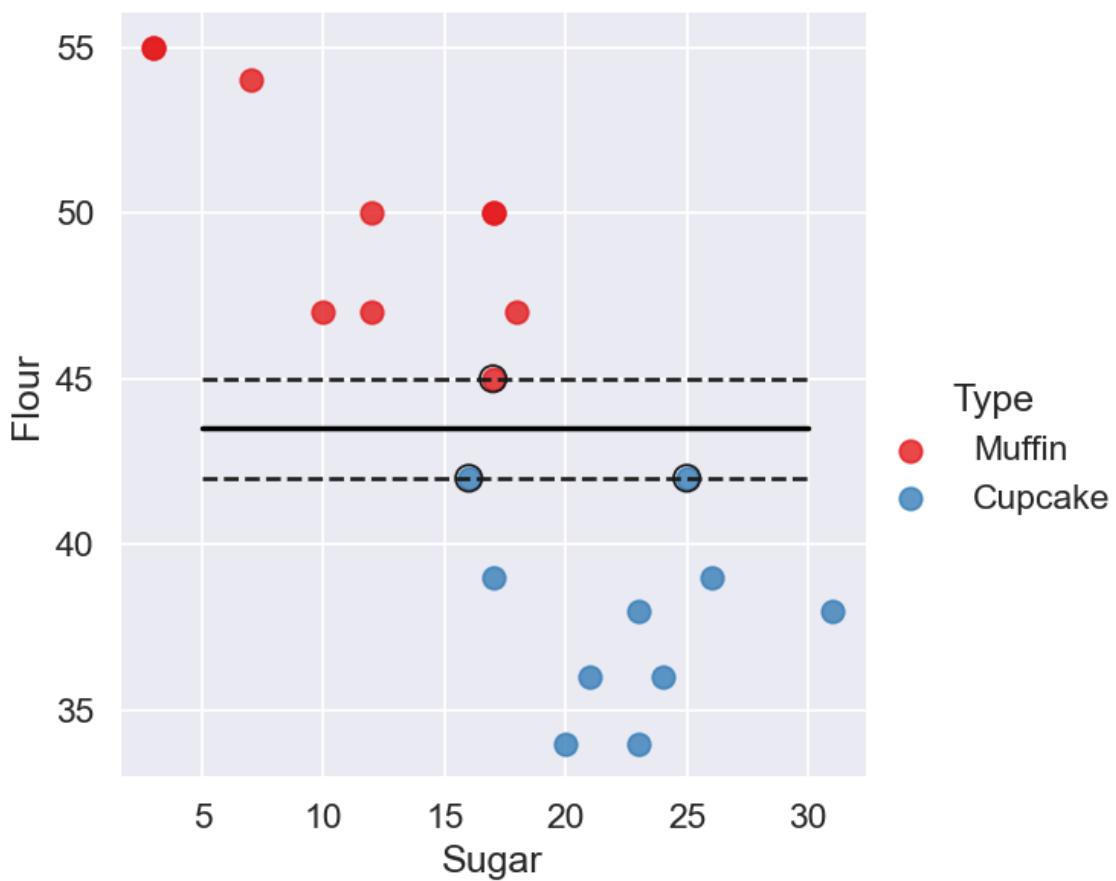
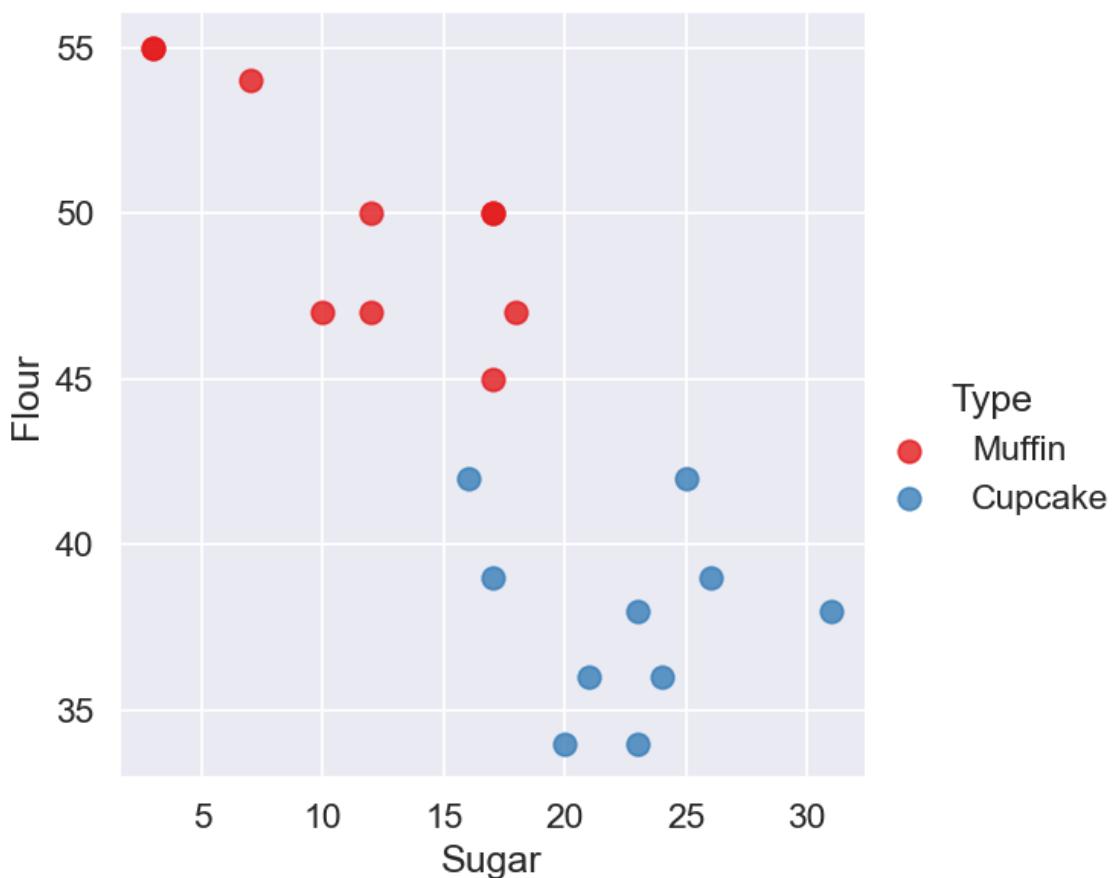


Figure 1



▲ ← → | + Q = | 🔍

## RESULT:

A PYTHON PROGRAM TO DO WITH DECISION TREE AS BEEN ANALYSED AND VERIFEID