Name : Akshit Kothari
Section : BCA A1
Roll no : 07 (2221118)

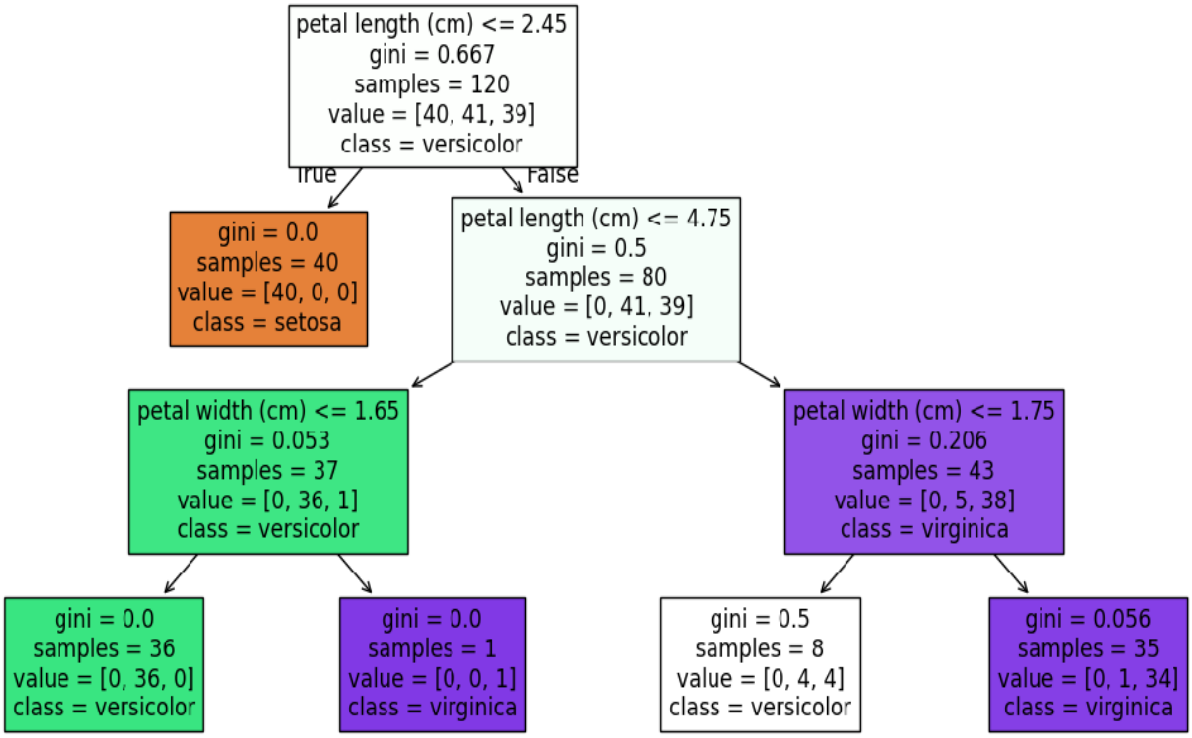
PROBLEM STATEMENT (16):- Write a program to implement decision tree.

PROGRAM:-

```
from sklearn.datasets import load_iris

import numpy as np

iris=load_iris()

x=iris.data

y=iris.target


from sklearn.model_selection import train_test_split

x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.2,random_state=42)


from sklearn.tree import DecisionTreeClassifier

clf=DecisionTreeClassifier(criterion='gini',max_depth=3,random_state=42)

clf.fit(x_train,y_train)

y_pred=clf.predict(x_test)

from sklearn.metrics import accuracy_score

accu=accuracy_score(y_test,y_pred)

print("accuracy",accu*100)


from sklearn import tree

import matplotlib.pyplot as plt

plt.figure(figsize=(12,8))

tree.plot_tree(clf,filled=True,feature_names=iris.feature_names,class_names=iris.target_names)

plt.show()
```

Name : Akshit Kothari
Section : BCA A1
Roll no : 07 (2221118)

Output:



Name : Akshit Kothari
Section : BCA A1
Roll no : 07 (2221118)

PROBLEM STATEMENT (15):- Write a program to implement naïve bayes.

PROGRAM:-

```
from sklearn.datasets import load_iris

import numpy as np

iris=load_iris()

x=iris.data

y=iris.target


from sklearn.model_selection import train_test_split

x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.2,random_state=42)


from sklearn.naive_bayes import GaussianNB

gb=GaussianNB()

gb.fit(x_train,y_train)


y_pred=gb.predict(x_test)


from sklearn.metrics import accuracy_score

accu=accuracy_score(y_test,y_pred)

print("prediction",y_pred)

print("accuracy",accu*100)
```

Name : Akshit Kothari
Section : BCA A1
Roll no : 07 (2221118)

Output:

prediction [1 0 2 1 1 0 1 2 1 1 2 0 0 0 0 1 2 1 1 2 0 2 0 2 2 2 2 2 0 0]

accuracy 100.0

Name : Akshit Kothari
Section : BCA A1
Roll no : 07 (2221118)

PROBLEM STATEMENT (14):- Write a program to implement SVM.

PROGRAM:-

```
from sklearn.datasets import load_iris

iris=load_iris()

x=iris.data[:,:2]

y=iris.target


from sklearn.model_selection import train_test_split

x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.3)


from sklearn.svm import SVC

svm=SVC(kernel='linear')

svm.fit(x_train,y_train)


y_pred=svm.predict(x_test)


from sklearn.metrics import accuracy_score

acu=accuracy_score(y_test,y_pred)

print(acu)

print(y_pred)


import numpy as np

x_min,x_max=x[:,0].min()-1,x[:,0].max()+1

y_min,y_max=x[:,1].min()-1,x[:,1].max()+1
```

Name : Akshit Kothari
Section : BCA A1
Roll no : 07 (2221118)

```
xx,yy=np.meshgrid(np.arange(x_min,x_max,0.02),np.arange(y_min,y_max,0.02))

z=svm.predict(np.c_[xx.ravel(),yy.ravel()])



import matplotlib.pyplot as plt

plt.contourf(xx,yy,z.reshape(xx.shape),alpha=0.5)

plt.scatter(x[:,0],x[:,1],c=y,cmap='viridis')

plt.show()
```

Name : Akshit Kothari
Section : BCA A1
Roll no : 07 (2221118)

Output: