NAIVE_BAYES

July 11, 2023

1 Import libraries

[ ]:
```
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
```

[ ]:
```
df=sns.load_dataset("iris")
df.head()
```

[ ]:
```
   sepal_length  sepal_width  petal_length  petal_width species
0           5.1          3.5           1.4          0.2  setosa
1           4.9          3.0           1.4          0.2  setosa
2           4.7          3.2           1.3          0.2  setosa
3           4.6          3.1           1.5          0.2  setosa
4           5.0          3.6           1.4          0.2  setosa
```

[ ]:
```
# selecting input and output
X=df.iloc[:,:-1]
y=df.iloc[:,-1:]
```

[ ]:
```
from sklearn.naive_bayes import GaussianNB
model=GaussianNB().fit(X,y)
model
```

```
c:\Users\shera\AppData\Local\Programs\Python\Python311\Lib\site-
packages\sklearn\utils\validation.py:1143: DataConversionWarning: A column-
vector y was passed when a 1d array was expected. Please change the shape of y
to (n_samples, ), for example using ravel().
  y = column_or_1d(y, warn=True)
```

[ ]:
```
GaussianNB()
```

[ ]:
```
# train test split and checking accuracy
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=0)
```

```
[ ]: #training the model on training data
     from sklearn.naive_bayes import GaussianNB
     model=GaussianNB().fit(X_train,y_train)
     model
```

```
c:\Users\shera\AppData\Local\Programs\Python\Python311\Lib\site-
packages\sklearn\utils\validation.py:1143: DataConversionWarning: A column-
vector y was passed when a 1d array was expected. Please change the shape of y
to (n_samples, ), for example using ravel().
  y = column_or_1d(y, warn=True)
```

```
[ ]: GaussianNB()
```

```
[ ]: # making predictions on testing data
     y_pred=model.predict(X_test)
     y_pred
```

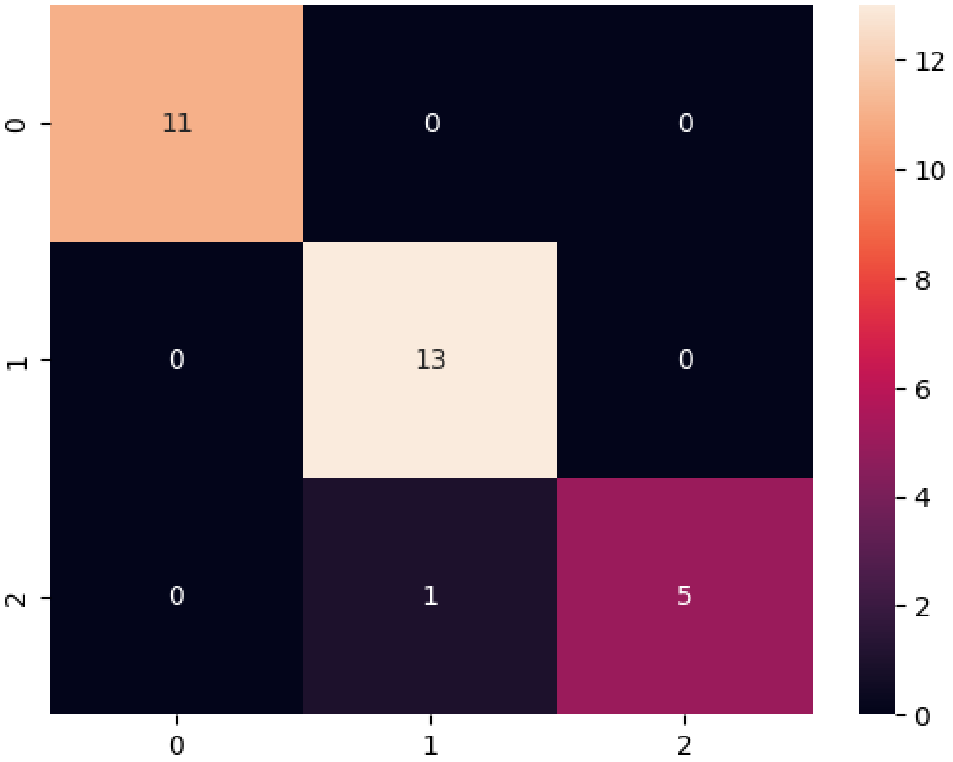
```
[ ]: array(['virginica', 'versicolor', 'setosa', 'virginica', 'setosa',
            'virginica', 'setosa', 'versicolor', 'versicolor', 'versicolor',
            'versicolor', 'versicolor', 'versicolor', 'versicolor',
            'versicolor', 'setosa', 'versicolor', 'versicolor', 'setosa',
            'setosa', 'virginica', 'versicolor', 'setosa', 'setosa',
            'virginica', 'setosa', 'setosa', 'versicolor', 'versicolor',
            'setosa'], dtype='<U10')
```

```
[ ]: from sklearn.metrics import accuracy_score
     score=accuracy_score(y_test, y_pred)
     print("Naive Bayes moddel accuracy =", score*100)
```

```
Naive Bayes moddel accuracy = 96.66666666666667
```

```
[ ]: from sklearn.metrics import confusion_matrix
     cm=confusion_matrix(y_test,y_pred)
     sns.heatmap(cm,annot=True)
```

```
[ ]: <Axes: >
```


11
0
0
0
13
0
0
1
5
0
1
2
0
1
2
12
10
8
6
4
2
0