lab6_svm

November 15, 2022

```
[ ]: import pandas as pd
     import numpy as np
     import matplotlib.pyplot as plt
     from sklearn.model_selection import train_test_split
     from sklearn.linear_model import LogisticRegression
     from sklearn import preprocessing,svm
     from sklearn.metrics import accuracy_score,confusion_matrix
     import graphviz
     from sklearn import tree
```

```
[ ]: data=pd.read_csv("C:\Coding\ML_python\machine-learning-lab-main\datasets\iris.
       ↪csv")
     data.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
```

```
[ ]: X=data.iloc[:,0:4]
     y=data.species
     print(X)
     print(y)
```

```
     sepal_length  sepal_width  petal_length  petal_width
0             5.1          3.5           1.4          0.2
1             4.9          3.0           1.4          0.2
2             4.7          3.2           1.3          0.2
3             4.6          3.1           1.5          0.2
4             5.0          3.6           1.4          0.2
..            …          …            …          …
145           6.7          3.0           5.2          2.3
146           6.3          2.5           5.0          1.9
147           6.5          3.0           5.2          2.0
148           6.2          3.4           5.4          2.3
149           5.9          3.0           5.1          1.8
```

```
[150 rows x 4 columns]
0         setosa
1         setosa
2         setosa
3         setosa
4         setosa
         …
145    virginica
146    virginica
147    virginica
148    virginica
149    virginica
Name: species, Length: 150, dtype: object
```

[ ]:

```
label_encoder = preprocessing.LabelEncoder()
y= label_encoder.fit_transform(y)
y
```

[ ]:

```
array([0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
       0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
       0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
       1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
       1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
       2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
       2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2])
```

[ ]:

```
model=svm.SVC()
xtrain,xtest,ytrain,ytest=train_test_split(X,y,test_size=0.3)
print(xtrain,ytrain)
```

```
     sepal_length  sepal_width  petal_length  petal_width
14            5.8          4.0           1.2          0.2
26            5.0          3.4           1.6          0.4
2             4.7          3.2           1.3          0.2
81            5.5          2.4           3.7          1.0
3             4.6          3.1           1.5          0.2
..            …            …             …            …
24            4.8          3.4           1.9          0.2
27            5.2          3.5           1.5          0.2
104           6.5          3.0           5.8          2.2
11            4.8          3.4           1.6          0.2
96            5.7          2.9           4.2          1.3

[105 rows x 4 columns] [0 0 0 1 0 1 0 0 1 1 0 2 2 2 0 1 1 0 2 2 2 0 0 1 1 2 2 1
2 1 1 1 2 0 2 0 0
 1 0 0 2 1 1 1 0 2 2 1 2 1 0 0 1 2 1 2 1 1 0 1 1 0 2 2 1 2 0 0 2 0 2 2 2 2
 1 2 2 0 1 0 0 1 1 2 1 0 0 1 1 1 0 0 2 1 2 1 0 0 2 1 0 0 2 0 1]
```

[ ]:

```
model.fit(xtrain,ytrain)
ypred=model.predict(xtest)
```

[ ]:

```
print(ypred)
print(ytest)
```

```
[2 0 2 0 2 2 1 2 0 1 0 1 1 0 2 2 2 0 2 0 2 0 0 1 2 1 2 2 0 2 2 0 0 1 1 1 0
 1 1 2 0 2 2 2 2]
[2 0 2 0 2 2 1 1 0 1 0 1 1 0 2 2 2 0 2 0 2 0 0 1 2 1 2 2 0 1 2 0 0 1 1 1 0
 1 1 2 0 2 2 2 2]
```

[ ]:

```
print(model.score(xtest,ytest))
```

```
0.9555555555555556
```