ML_Lab_NBC

June 24, 2021

0.1 24 June 2021

0.2 ML Lab 4

0.3 Naive Bayes Classifier

0.3.1 Dr Neeraj Gupta

[1]:

```
#Importing the libraries
import numpy as nm
import matplotlib.pyplot as mtp
import pandas as pd
```

[5]:

```
# Importing the dataset
dataset = pd.read_csv('image/user_data.csv')

X = dataset.iloc[:, [2,3]].values
#print(X)

y = dataset.iloc[:, 4].values
print(y)
```

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

[10]:

```
# Splitting the dataset into the Training set and Test set
from sklearn.model_selection import train_test_split
x_train, x_test, y_train, y_test = train_test_split(X, y, test_size = 0.
 ↪25,random_state = 0)
```

```
#x_train
```

[13]:

```
# Feature Scaling
from sklearn.preprocessing import StandardScaler
sc = StandardScaler()
x_train = sc.fit_transform(x_train)
x_test = sc.fit_transform(x_test)
#x_train
```

[16]:

```
#Fitting Naive Bayes to the training data
from sklearn.naive_bayes import GaussianNB

clf = GaussianNB()
clf.fit(x_train, y_train)
```

[16]:

```
GaussianNB(priors=None, var_smoothing=1e-09)
```

[17]:

```
#Predicting the test data results

y_pred = clf.predict(x_test)
y_pred
```

[17]:

```
array([0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0, 1, 0, 0, 1, 0, 0, 1,
       0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0,
       1, 0, 0, 1, 0, 1, 1, 0, 0, 1, 1, 0, 0, 0, 1, 0, 0, 1, 0, 0, 0, 1,
       0, 0, 0, 0, 1, 0, 0, 1, 0, 0, 0, 0, 1, 1, 1, 1, 0, 0, 1, 0, 0, 1,
       1, 0, 0, 1, 0, 0, 0, 0, 0, 1, 1, 1], dtype=int64)
```

[18]:

```
#Making the confusion matrix
from sklearn.metrics import confusion_matrix
cm = confusion_matrix(y_test, y_pred)
cm
```

[18]:

```
array([[64,  4],
       [ 5, 27]], dtype=int64)
```

[21]:

```
from sklearn.metrics import accuracy_score
print(accuracy_score(y_test, y_pred))
```

```
0.91
```

0.4 Support Vector Machine

[23]:

```
import pandas as pd
data = pd.read_csv("image/apples_and_oranges.csv")
```

[24]:

```
data.head()
```

[24]:

```
   Weight  Size   Class
0      69  4.39  orange
1      69  4.21  orange
2      65  4.09  orange
3      72  5.85   apple
4      67  4.70  orange
```

[25]:

```
from sklearn.model_selection import train_test_split
training_set, test_set = train_test_split(data, test_size = 0.2, random_state = 1)
```

[26]:

```
X_train = training_set.iloc[:,0:2].values
Y_train = training_set.iloc[:,2].values
X_test = test_set.iloc[:,0:2].values
Y_test = test_set.iloc[:,2].values
```

[36]:

```
#Initial SVM and fitting the training data
from sklearn.svm import SVC

clf = SVC(kernel='linear', random_state=1) # poly, rbf

clf.fit(X_train, Y_train)

Y_pred = clf.predict(X_test)

print(Y_pred)
print(Y_test)
```

```
['orange' 'orange' 'apple' 'apple' 'orange' 'apple' 'orange' 'apple']
['orange' 'orange' 'apple' 'orange' 'orange' 'apple' 'orange' 'apple']
```

[37]:

```
from sklearn.metrics import confusion_matrix
cm = confusion_matrix(Y_test, Y_pred)
cm
```

[37]:

```
array([[3, 0],
       [1, 4]], dtype=int64)
```

[38]:

```
from sklearn.metrics import accuracy_score
print(accuracy_score(Y_test, Y_pred))
```

```
0.875
```

[ ]: