Jaiwin Shah
2019130058
Te Comps
Batch C

Experiment 4

Aim: To train and test a machine learning model using naïve bayes algorithm.

Theory: Naive Bayes classifiers are a collection of classification algorithms based on **Bayes' Theorem**. It is not a single algorithm but a family of algorithms where all of them share a common principle, i.e. every pair of features being classified is independent of each other.

The fundamental Naive Bayes assumption is that each feature makes an independent and equal contribution to the outcome.

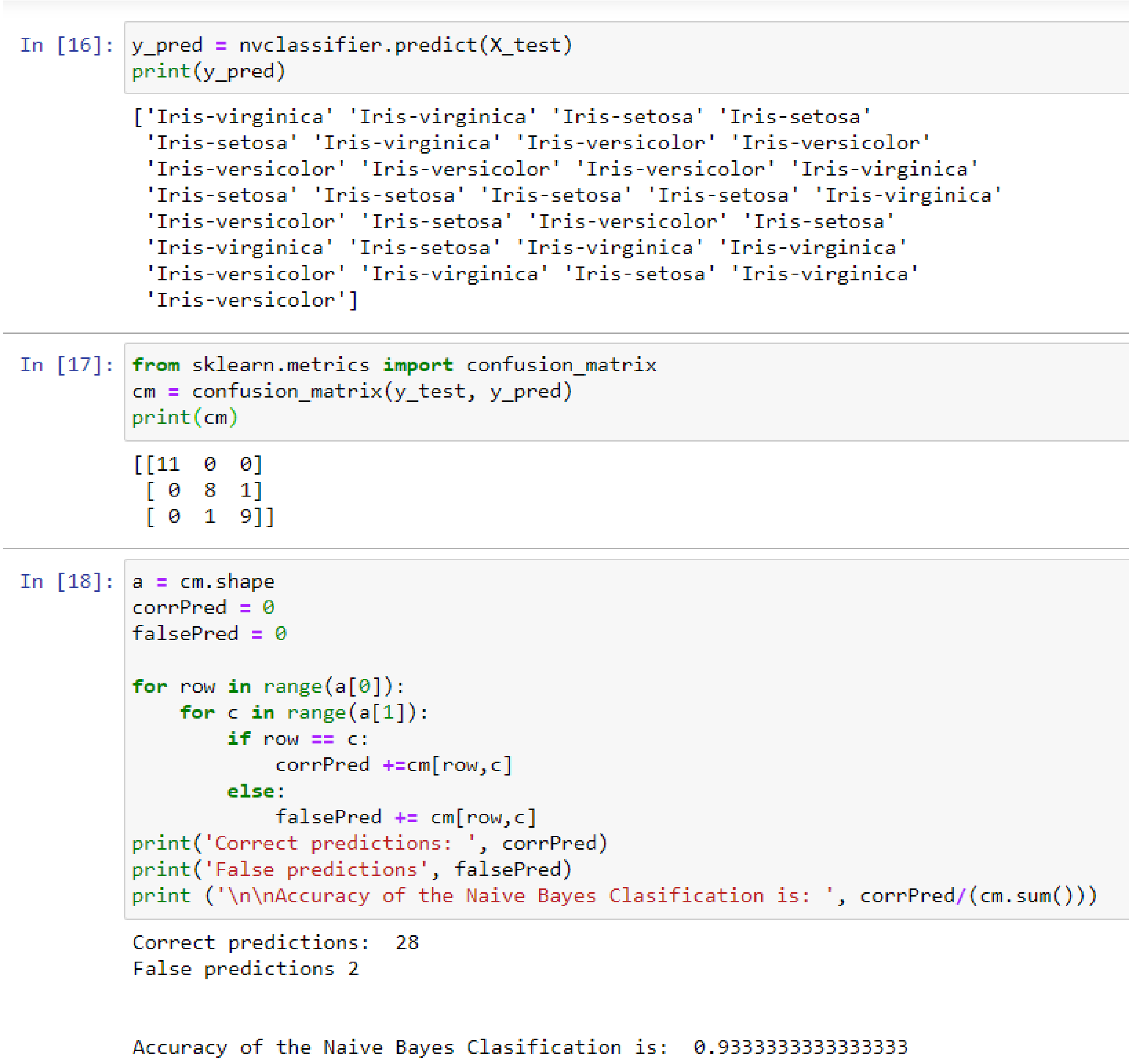
Bayes' Theorem finds the probability of an event occurring given the probability of another event that has already occurred.

Code:

```
import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

import seaborn as sns


dataset=pd.read_csv(r"C:\Users\jaiwi\Desktop\Iris.csv")

dataset= dataset.drop(columns=['Id'], axis= 1)

dataset

X = dataset.iloc[:,:4].values

X

y = dataset['Species'].values

from sklearn.model_selection import train_test_split


X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.20,
random_state = 82)
```

```
# Feature Scaling to bring the variable in a single scale
from sklearn.preprocessing import StandardScaler
sc = StandardScaler()
X_train = sc.fit_transform(X_train)
X_test = sc.transform(X_test)

from sklearn.naive_bayes import GaussianNB
nvclassifier = GaussianNB()
nvclassifier.fit(X_train, y_train)

y_pred = nvclassifier.predict(X_test)
print(y_pred)

from sklearn.metrics import confusion_matrix
cm = confusion_matrix(y_test, y_pred)
print(cm)

a = cm.shape
corrPred = 0
falsePred = 0

for row in range(a[0]):
    for c in range(a[1]):
        if row == c:
            corrPred +=cm[row,c]
        else:
            falsePred += cm[row,c]
print('Correct predictions: ', corrPred)
print('False predictions', falsePred)
print ('\n\nAccuracy of the Naive Bayes Clasification is: ', corrPred/(cm.sum()))
```

Output:



```
In [16]: y_pred = nvclassifier.predict(X_test)
         print(y_pred)

         ['Iris-virginica' 'Iris-virginica' 'Iris-setosa' 'Iris-setosa'
          'Iris-setosa' 'Iris-virginica' 'Iris-versicolor' 'Iris-versicolor'
          'Iris-versicolor' 'Iris-versicolor' 'Iris-versicolor' 'Iris-virginica'
          'Iris-setosa' 'Iris-setosa' 'Iris-setosa' 'Iris-setosa' 'Iris-virginica'
          'Iris-versicolor' 'Iris-setosa' 'Iris-versicolor' 'Iris-setosa'
          'Iris-virginica' 'Iris-setosa' 'Iris-virginica' 'Iris-virginica'
          'Iris-versicolor' 'Iris-virginica' 'Iris-setosa' 'Iris-virginica'
          'Iris-versicolor']
```

```
In [17]: from sklearn.metrics import confusion_matrix
         cm = confusion_matrix(y_test, y_pred)
         print(cm)

         [[11  0  0]
          [ 0  8  1]
          [ 0  1  9]]
```

```
In [18]: a = cm.shape
         corrPred = 0
         falsePred = 0

         for row in range(a[0]):
             for c in range(a[1]):
                 if row == c:
                     corrPred +=cm[row,c]
                 else:
                     falsePred += cm[row,c]
         print('Correct predictions: ', corrPred)
         print('False predictions', falsePred)
         print ('\n\nAccuracy of the Naive Bayes Clasification is: ', corrPred/(cm.sum()))

         Correct predictions:  28
         False predictions 2


         Accuracy of the Naive Bayes Clasification is:  0.9333333333333333
```

Conclusion:

From above experiment of naïve bayes, I learned about the basic bayes theorem. **Bayes' theorem** describes the probability of occurrence of an event related to any condition. In naïve bayes algorithm, we find out the probability of the category of the output and select the category with highest probability.

Naïve bayes algorithm works on two assumptions that every data point in the dataset contributes independently and equally to the dataset.

Using naïve bayes algorithm, we can predict the category with fair accuracy of probably around than 90-95%.