Name: Meith Navlakha
SAP: 60004190068
DWM Practical

EXPERIMENT 2

AIM:

Implementation of Classification algorithm using:

1. Decision Tree ID3
2. Naïve Bayes algorithm

THEORY:

Naïve Bayes:

Naive Bayes methods are a set of supervised learning algorithms based on applying Bayes' theorem with the "naive" assumption of conditional independence between every pair of features given the value of the class variable. Bayes' theorem states the following relationship, given class variable y and dependent feature vector x_1 through x_n, :

$$P(y \mid x_1, \ldots, x_n) = \frac{P(y)P(x_1, \ldots, x_n \mid y)}{P(x_1, \ldots, x_n)}$$

and we can use Maximum A Posteriori (MAP) estimation to estimate P(y) and P(xi|y); the former is then the relative frequency of class y in the training set.

$$\hat{y} = \arg\max_y P(y) \prod_{i=1}^{n} P(x_i \mid y),$$

The different naive Bayes classifiers differ mainly by the assumptions they make regarding the distribution of P(xi|y).

Decision Tree:

Decision Trees (DTs) are a non-parametric supervised learning method used for classification and regression. The goal is to create a model that predicts the value of a target variable by learning simple decision rules inferred from the data features. A tree can be seen as a piecewise constant approximation.

Name: Meith Navlakha
SAP: 60004190068
DWM Practical

ID3 (Iterative Dichotomiser 3) was developed in 1986 by Ross Quinlan. The algorithm creates a multiway tree, finding for each node (i.e. in a greedy manner) the categorical feature that will yield the largest information gain for categorical targets. Trees are grown to their maximum size and then a pruning step is usually applied to improve the ability of the tree to generalise to unseen data.

Advantages Decision Tree:

- Simple to understand and to interpret. Trees can be visualised.
- Requires little data preparation. Other techniques often require data normalisation, dummy variables need to be created and blank values to be removed. Note however that this module does not support missing values.
- Able to handle both numerical and categorical data.
- Able to handle multi-output problems.
- Possible to validate a model using statistical tests. That makes it possible to account for the reliability of the model.

Disadvantages Decision Trees:

- Decision-tree learners can create over-complex trees that do not generalise the data well. This is called overfitting.
- Decision trees can be unstable because small variations in the data might result in a completely different tree being generated.
- Decision tree learners create biased trees if some classes dominate. It is therefore recommended to balance the dataset prior to fitting with the decision tree.

CODE:

```
from sklearn.model_selection import train_test_split
from sklearn.metrics import confusion_matrix
from sklearn.metrics import r2_score
from sklearn.metrics import classification_report , auc ,accuracy_score ,
precision_score , recall_score, roc_auc_score , f1_score, roc_curve
# from sklearn.metrics import f1_sco
from sklearn.naive_bayes import GaussianNB
from sklearn import tree
```

```
def trainModel(X ,y , data_name):
  X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3 ,
random_state=0)

  # Naive Bayes
  nb = GaussianNB()
  nb = nb.fit(X_train, y_train)
  y_pred_nb = nb.predict(X_test)
  acc_nb = r2_score(y_test , y_pred_nb)
  cm_nb = confusion_matrix(y_test , y_pred_nb)
  # precision_nb = precision_score(y_test , y_pred_nb ,average='binary')
  report_nb = classification_report(y_test , y_pred_nb ,labels=[1,0])

  print('\n*****', data_name , '*****')
  print('\nNaives Bayes : ')
  print('Accuracy : ' ,acc_nb)
  # print('Precision : ' ,precision_nb)
  print('\nConfusion Matrix : \n' ,cm_nb)
  print('\nReport : \n' , report_nb)

  # Decision Tree
  dt = tree.DecisionTreeClassifier()
  dt = dt.fit(X_train, y_train)
  y_pred_dt = dt.predict(X_test)
  acc_dt = r2_score(y_test , y_pred_dt)
  cm_dt = confusion_matrix(y_test , y_pred_dt)
  # precision_dt = precision_score(y_test , y_pred_dt ,average='binary')
  report_dt = classification_report(y_test , y_pred_dt ,labels=[1,0])


  print('\nDecision Tree : ')
  print('Accuracy : ' ,acc_dt)
  # print('Precision : ' ,precision_dt)
  print('\nConfusion Matrix : \n' ,cm_dt)
  print('\nReport : \n' , report_dt)

  return (acc_nb , acc_dt)


// DATASET

models= []
names = []

from sklearn.datasets import load_digits
X, y = load_digits(return_X_y=True)
```

```
acc_nb_4 , acc_dt_4 = trainModel(X, y , 'LOAD DIGITS')
models.append((acc_nb_4 , acc_dt_4))
names.append('LOAD DIGITS')

from sklearn.datasets import load_ine
X, y = load_wine(return_X_y=True)
acc_nb_2 , acc_dt_2 = trainModel(X, y , 'DIABETES DATASET')
models.append((acc_nb_2 , acc_dt_2))
names.append('DIABETES')

from sklearn.datasets import load_breast_cancer
X, y = load_breast_cancer(return_X_y=True)
acc_nb_3 , acc_dt_3 = trainModel(X, y , 'BREAST CANCER')
models.append((acc_nb_3 , acc_dt_3))
names.append('BREAST CANCER')

from sklearn.datasets import load_iris
X, y = load_iris(return_X_y=True)
acc_nb_1 , acc_dt_1 = trainModel(X, y , 'IRIS DATASET')
models.append(( acc_nb_1 , acc_dt_1))
names.append('IRIS')

from sklearn.datasets import fetch_olivetti_faces
X, y = fetch_olivetti_faces(return_X_y=True)
acc_nb_5 , acc_dt_5 = trainModel(X, y , 'OLIVETTI FACES')
models.append((acc_nb_5 , acc_dt_5))
names.append('OLIVETTI FACES')
```

Name: Meith Navlakha
SAP: 60004190068
DWM Practical

OUTPUT:

PART A

1) Diabetes Dataset

```
***** DIABETES DATASET *****

Naives Bayes :
Accuracy :  0.9042553191489362

Confusion Matrix :
 [[19  0  0]
 [ 2 19  1]
 [ 0  0 13]]

Report :
              precision    recall  f1-score   support

           1       1.00      0.86      0.93        22
           0       0.90      1.00      0.95        19

   micro avg       0.95      0.93      0.94        41
   macro avg       0.95      0.93      0.94        41
weighted avg       0.96      0.93      0.94        41



Decision Tree :
Accuracy :  0.9361702127659575

Confusion Matrix :
 [[18  1  0]
 [ 0 21  1]
 [ 0  0 13]]

Report :
              precision    recall  f1-score   support

           1       0.95      0.95      0.95        22
           0       1.00      0.95      0.97        19

   micro avg       0.97      0.95      0.96        41
   macro avg       0.98      0.95      0.96        41
weighted avg       0.98      0.95      0.96        41
```

Name: Meith Navlakha
SAP: 60004190068
DWM Practical

2) Breast Cancer

```
***** BREAST CANCER *****

Naives Bayes :
Accuracy :  0.6732804232804233

Confusion Matrix :
 [[ 57   6]
 [  7 101]]

Report :
              precision    recall  f1-score   support

           1       0.94      0.94      0.94       108
           0       0.89      0.90      0.90        63

    accuracy                           0.92       171
   macro avg       0.92      0.92      0.92       171
weighted avg       0.92      0.92      0.92       171
```

```
Decision Tree :
Accuracy :  0.6984126984126984

Confusion Matrix :
 [[60  3]
 [ 9 99]]

Report :
              precision    recall  f1-score   support

           1       0.97      0.92      0.94       108
           0       0.87      0.95      0.91        63

    accuracy                           0.93       171
   macro avg       0.92      0.93      0.93       171
weighted avg       0.93      0.93      0.93       171
```

Name: Meith Navlakha
SAP: 60004190068
DWM Practical

Name: Meith Navlakha
SAP: 60004190068
DWM Practical

3) Load Digit

```
***** LOAD DIGITS *****

Naives Bayes :
Accuracy :  0.5021533751442248

Confusion Matrix :
 [[45  0  0  0  0  0  0  0  0  0]
 [ 0 46  0  0  0  0  0  0  6  0]
 [ 0  6 26  5  0  0  0  0 16  0]
 [ 0  0  0 46  0  0  0  1  6  1]
 [ 0  3  0  0 36  0  2  7  0  0]
 [ 0  1  0  2  0 51  1  2  0  0]
 [ 0  0  1  0  0  0 59  0  0  0]
 [ 0  0  0  0  1  0  0 52  0  0]
 [ 0  5  0  3  0  1  0  1 51  0]
 [ 0  1  0 14  1  0  0  3  5 33]]

Report :
              precision    recall  f1-score   support

           1       0.74      0.88      0.81        52
           0       1.00      1.00      1.00        45

   micro avg       0.85      0.94      0.89        97
   macro avg       0.87      0.94      0.90        97
weighted avg       0.86      0.94      0.90        97


Decision Tree :
Accuracy :  0.6543176028422267

Confusion Matrix :
 [[42  0  0  2  1  0  0  0  0  0]
 [ 0 44  1  1  5  0  1  0  0  0]
 [ 1  1 41  2  1  0  1  2  3  1]
 [ 0  0  3 41  0  2  1  0  4  3]
 [ 1  0  0  1 44  0  0  0  2  0]
 [ 0  0  1  2  0 47  0  2  2  3]
 [ 1  0  0  0  4  0 54  0  0  1]
 [ 0  1  0  0  0  0  0 48  2  2]
 [ 0  5  2  4  1  0  0  3 42  4]
 [ 0  1  1  3  0  2  0  0  2 48]]

Report :
              precision    recall  f1-score   support

           1       0.85      0.85      0.85        52
           0       0.93      0.93      0.93        45

   micro avg       0.89      0.89      0.89        97
   macro avg       0.89      0.89      0.89        97
weighted avg       0.89      0.89      0.89        97
```

Name: Meith Navlakha
SAP: 60004190068
DWM Practical

4) Iris Dataset

```
***** IRIS DATASET *****

Naives Bayes :
Accuracy :  1.0

Confusion Matrix :
 [[16  0  0]
 [ 0 18  0]
 [ 0  0 11]]

Report :
              precision    recall  f1-score   support

           1       1.00      1.00      1.00        18
           0       1.00      1.00      1.00        16

   micro avg       1.00      1.00      1.00        34
   macro avg       1.00      1.00      1.00        34
weighted avg       1.00      1.00      1.00        34


   Decision Tree :
   Accuracy :  0.9621848739495799

   Confusion Matrix :
    [[16  0  0]
    [ 0 17  1]
    [ 0  0 11]]

   Report :
                  precision    recall  f1-score   support

              1       1.00      0.94      0.97        18
              0       1.00      1.00      1.00        16

      micro avg       1.00      0.97      0.99        34
      macro avg       1.00      0.97      0.99        34
   weighted avg       1.00      0.97      0.98        34
```

Name: Meith Navlakha
SAP: 60004190068
DWM Practical

5) Olivetti Faces

```
***** OLIVETTI FACES *****

Naives Bayes :
Accuracy :  0.554752833331136

Confusion Matrix :
 [[3 0 0 ... 0 0 0]
 [0 2 0 ... 0 0 0]
 [0 0 2 ... 0 0 1]
 ...
 [0 0 0 ... 3 0 0]
 [0 0 0 ... 0 1 0]
 [0 0 0 ... 0 0 3]]

Report :
              precision    recall  f1-score   support

           1       1.00      0.50      0.67         4
           0       1.00      0.50      0.67         6

   micro avg       1.00      0.50      0.67        10
   macro avg       1.00      0.50      0.67        10
weighted avg       1.00      0.50      0.67        10



  Decision Tree :
  Accuracy :  -0.1146120346811157

  Confusion Matrix :
   [[0 0 0 ... 1 0 0]
   [0 1 0 ... 1 0 0]
   [0 0 0 ... 0 0 0]
   ...
   [0 0 0 ... 2 0 0]
   [0 0 0 ... 0 1 0]
   [0 0 0 ... 0 0 1]]

  Report :
                precision    recall  f1-score   support

             1       0.50      0.25      0.33         4
             0       0.00      0.00      0.00         6

     micro avg       0.50      0.10      0.17        10
     macro avg       0.25      0.12      0.17        10
  weighted avg       0.20      0.10      0.13        10
```

Comparison

	Naive Bayes	Decision Tree
LOAD DIGITS	0.502153	0.625987
DIABETES	0.904255	0.904255
BREAST CANCER	0.673280	0.648148
IRIS	1.000000	0.962185
OLIVETTI FACES	0.554753	0.131537

Conclusion:

In this experiment I implemented Naïve Bayes and Decision Tree on 5 datasets viz, Iris, Diabetes, Load Digits, Breast Cancer and Olivetti Faces and displayed the accuracy, precision, recall, confusion matrix. Both Decision Tree and Naïve Bayes did well on Daibetes and Iris dataset. Decision Tree had very low accuracy in Olivetti Faces.