Name: Md Toufique Hasan

Student Number: 151129267

Exercise date: 08/12/2022

Data Mining Exercise: 5

Answer: 1

```
% Load Data
iris_data = readmatrix("D:\TUNI\Courses\Period-2\DATA.ML.340 [Data Mining]\" + ...
    "Weekly exercises 4\Iris.txt");

% Columns of COEFF contains the direction vectors for new vector space
% PCA centeres the data
[COEFF, SCORE] = pca(iris_data);
m = mean(iris_data)
```

```
m = 1×6
   75.5000    5.8433    3.0540    3.7587    1.1987    2.0000
```

```
m = repmat(m,150,1);
cdata = iris_data-m;
r1 = COEFF'*cdata(1,:)'
```

```
r1 = 6×1
  -74.5535
    0.4971
    0.1348
   -0.1535
    0.1892
   -0.0957
```

```
b = inv(COEFF')*r1
```

```
b = 6×1
  -74.5000
   -0.7433
    0.4460
   -2.3587
   -0.9987
   -1.0000
```

```
% Latent contains the principal component variances
[coeff,score,latent] = pca(iris_data);
s = sum(latent)
```

```
s = 1.8927e+03
```

```
l = latent*100
```

```
l = 6×1
10^5 ×
    1.8914
    0.0010
    0.0002
    0.0001
    0.0000
    0.0000
```

```
l = l/s
```

```
l = 6×1
   99.9280
    0.0535
    0.0120
    0.0040
    0.0015
    0.0011
```

```
e2 = sum(l(1:2))
```

```
e2 = 99.9815
```

Answer: 2

```
K = 3
```

```
K = 3
```

```
ResponVariable = iris_data(:,6);
PredVariables = iris_data(:,2:5);

[idx,weights] = relieff(PredVariables,ResponVariable,3)
```

```
idx = 1×4
     4     2     1     3
weights = 1×4
    0.0106    0.0154    0.0082    0.0207
```

```
K = 4
```

```
K = 4
```

```
[idx,weights] = relieff(PredVariables,ResponVariable,4)
```

```
idx = 1×4
     4     3     1     2
weights = 1×4
    0.0083    0.0047    0.0099    0.0218
```

```
K = 5
```

```
K = 5
```

```
[idx,weights] = relieff(PredVariables,ResponVariable,5)
```

```
idx = 1×4
     4     1     3     2
weights = 1×4
    0.0117    0.0029    0.0111    0.0219
```

```
K = 6
```

```
K = 6
```

```
[idx,weights] = relieff(PredVariables,ResponVariable,6)
```

```
idx = 1×4
     4     3     1     2
weights = 1×4
    0.0110    0.0043    0.0156    0.0234
```

```
K = 7
```

```
K = 7
```

```
[idx,weights] = relieff(PredVariables,ResponVariable,7)
```

```
idx = 1×4
     4     3     1     2
weights = 1×4
    0.0095    0.0064    0.0159    0.0224
```

```
K = 8
```

```
K = 8
```

```
[idx,weights] = relieff(PredVariables,ResponVariable,8)
```

```
idx = 1×4
     4     3     1     2
weights = 1×4
    0.0065    0.0056    0.0156    0.0259
```

```
K = 9
```

```
K = 9
```

```
[idx,weights] = relieff(PredVariables,ResponVariable,9)
```

```
idx = 1×4
     4     3     2     1
weights = 1×4
    0.0008    0.0022    0.0198    0.0312
```

```
K = 10
```

```
K = 10
```

```
[idx,weights] = relieff(PredVariables,ResponVariable,10)
```

```
idx = 1×4
     4     3     2     1
weights = 1×4
    0.0005    0.0037    0.0172    0.0317
```

Answer: 3

```
A1 = iris_data(1:40,2:end);
A2 = iris_data(51:90,2:end);
A3 = iris_data(101:140,2:end);

fprintf('40 first cases from each class')
```

```
40 first cases from each class
```

```
iris_clean = [A1;A2;A3]
```

```
iris_clean = 120×5
    5.1000    3.5000    1.4000    0.2000    1.0000
    4.9000    3.0000    1.4000    0.2000    1.0000
    4.7000    3.2000    1.3000    0.2000    1.0000
    4.6000    3.1000    1.5000    0.2000    1.0000
    5.0000    3.6000    1.4000    0.2000    1.0000
    5.4000    3.9000    1.7000    0.4000    1.0000
    4.6000    3.4000    1.4000    0.3000    1.0000
    5.0000    3.4000    1.5000    0.2000    1.0000
    4.4000    2.9000    1.4000    0.2000    1.0000
    4.9000    3.1000    1.5000    0.1000    1.0000
      ⋮
```

```
[idx, Centriod] = kmeans(iris_data,3);
Centriod
```

```
Centriod = 3×6
  125.0000    6.5706    2.9706    5.5235    2.0118    2.9804
   25.0000    5.0061    3.4204    1.4653    0.2449    1.0000
   74.5000    5.9220    2.7800    4.2060    1.3040    1.9800
```

```
B1 = iris_data(41:50,2:end);
B2 = iris_data(91:100,2:end);
B3 = iris_data(141:150,2:end);

fprintf('Remaining 10 cases from each class')
```

```
Remaining 10 cases from each class
```

```
Rem10 = [B1;B2;B3]
```

```
Rem10 = 30×5
    5.0000    3.5000    1.3000    0.3000    1.0000
    4.5000    2.3000    1.3000    0.3000    1.0000
    4.4000    3.2000    1.3000    0.2000    1.0000
    5.0000    3.5000    1.6000    0.6000    1.0000
    5.1000    3.8000    1.9000    0.4000    1.0000
    4.8000    3.0000    1.4000    0.3000    1.0000
    5.1000    3.8000    1.6000    0.2000    1.0000
    4.6000    3.2000    1.4000    0.2000    1.0000
    5.3000    3.7000    1.5000    0.2000    1.0000
    5.0000    3.3000    1.4000    0.2000    1.0000
      ⋮
```

```
Rem10NeedPrediction = Rem10(:,1:4);
OriginalClasses = Rem10(:,5)
```

```
OriginalClasses = 30×1
     1
     1
     1
     1
     1
     1
     1
     1
     1
     1
     ⋮
```

```
X = iris_clean(:,1:4)
```

```
X = 120×4
    5.1000    3.5000    1.4000    0.2000
    4.9000    3.0000    1.4000    0.2000
    4.7000    3.2000    1.3000    0.2000
    4.6000    3.1000    1.5000    0.2000
    5.0000    3.6000    1.4000    0.2000
    5.4000    3.9000    1.7000    0.4000
    4.6000    3.4000    1.4000    0.3000
    5.0000    3.4000    1.5000    0.2000
    4.4000    2.9000    1.4000    0.2000
    4.9000    3.1000    1.5000    0.1000
     ⋮
```

```
Y = iris_clean(:,5)
```

```
Y = 120×1
     1
     1
     1
     1
     1
     1
     1
     1
     1
     1
     ⋮
```

```
Mdl = fitcnb(X,Y)
```

```
Mdl =
  ClassificationNaiveBayes
             ResponseName: 'Y'
    CategoricalPredictors: []
               ClassNames: [1 2 3]
           ScoreTransform: 'none'
          NumObservations: 120
         DistributionNames: {'normal'  'normal'  'normal'  'normal'}
    DistributionParameters: {3×4 cell}
```

Properties, Methods

```
Mdl.ClassNames
```

```
ans = 3×1
     1
     2
     3
```

```
Mdl.Prior
```

```
ans = 1×3
    0.3333    0.3333    0.3333
```

```
Mdl.Prior = [0.5 0.2 0.3];
Mdl.Prior
```

```
ans = 1×3
    0.5000    0.2000    0.3000
```

```
PredictedClasses = Mdl.predict(Rem10NeedPrediction)
```

```
PredictedClasses = 30×1
     1
     1
     1
     1
     1
     1
     1
     1
     1
     1
     ⋮
```

```
Conf = confusionmat(PredictedClasses, OriginalClasses)
```

```
Conf = 3×3
    10     0     0
     0    10     0
     0     0    10
```

Answer: 4

```
Var6 = iris_data(:,6);
IrisDataRed = iris_data(:,2:5)
```

```
IrisDataRed = 150×4
    5.1000    3.5000    1.4000    0.2000
    4.9000    3.0000    1.4000    0.2000
    4.7000    3.2000    1.3000    0.2000
    4.6000    3.1000    1.5000    0.2000
    5.0000    3.6000    1.4000    0.2000
    5.4000    3.9000    1.7000    0.4000
    4.6000    3.4000    1.4000    0.3000
```

```
    5.0000    3.4000    1.5000    0.2000
    4.4000    2.9000    1.4000    0.2000
    4.9000    3.1000    1.5000    0.1000
      ⋮
```

```
Reducedata = pca(IrisDataRed)
```

```
Reducedata = 4×4
    0.3616    0.6565   -0.5810    0.3173
   -0.0823    0.7297    0.5964   -0.3241
    0.8566   -0.1758    0.0725   -0.4797
    0.3588   -0.0747    0.5491    0.7511
```

```
PredictedClasses = Mdl.predict(Reducedata)
```

```
PredictedClasses = 4×1
     3
     3
     3
     3
```