Aprendizagem 2021/22

**Homework I – Group 117**

# I. Pen-and-paper

## Answer 1


Bernardo Castiço ist196845 HOMEWORK 4

Hugo Rita ist196870


1- $P(X|C=1) = N\left(\mu_1 = \begin{bmatrix}2\\4\end{bmatrix}, \Sigma_1 = \begin{bmatrix}1&0\\0&1\end{bmatrix}\right)$ $P(X|C=2) = N\left(\mu_2 = \begin{bmatrix}-1\\-4\end{bmatrix}, \Sigma_2 = \begin{bmatrix}2&0\\0&2\end{bmatrix}\right)$

$\pi_1 = P(C_1=1) = 0,7$ $\pi_2 = P(C_2=1) = 0,3$

· E-step

$x_1 = \begin{bmatrix}2\\4\end{bmatrix}$ $\boxed{\text{Para } c=1}$

→ prior: $p(C=1) = 0,7$

→ likelihood: $P(x|C=1) = \frac{1}{2\pi}\frac{1}{\det\Sigma^1}\exp\left(-\frac{1}{2}(x_1-\mu_1)^T(\Sigma^1)^{-1}(x_1-\mu_1)\right) =$

$= \frac{1}{2\pi}\frac{1}{\det\Sigma^1}\exp\left(-\frac{1}{2}\left(\begin{bmatrix}2\\4\end{bmatrix}-\begin{bmatrix}2\\4\end{bmatrix}\right)^T\cdot\begin{bmatrix}1&0\\0&1\end{bmatrix}^{-1}\cdot\left(\begin{bmatrix}2\\4\end{bmatrix}-\begin{bmatrix}2\\4\end{bmatrix}\right)\right) = 0,1592$

→ joint probability

$P(C=1,x_1) = P(C=1).P(x_1|C=1) = 0,7 \times 0,1592 = 0,1114$

$\boxed{\text{Para } c=2}$

→ prior: $P(C=2) = 0,3$

→ likelihood: $P(x|C=2) = \frac{1}{2\pi}\frac{1}{\det\Sigma^2}\exp\left(-\frac{1}{2}(x_1-\mu_2)^T(\Sigma^2)^{-1}(x_1-\mu_2)\right)$

$= \frac{1}{2\pi}\frac{1}{\det\Sigma^2}\exp\left(-\frac{1}{2}\left(\begin{bmatrix}2\\4\end{bmatrix}-\begin{bmatrix}-1\\-4\end{bmatrix}\right)^T\left(\begin{bmatrix}2&0\\0&2\end{bmatrix}\right)^{-1}\left(\begin{bmatrix}2\\4\end{bmatrix}-\begin{bmatrix}-1\\-4\end{bmatrix}\right)\right) = 9,439\times10^{-10}$

→ joint probability

$P(C=2,x_1) = P(C=2).P(x_1|C=2) = 0,3\times9,439\times10^{-10} =$
$= 2,832\times10^{-10}$

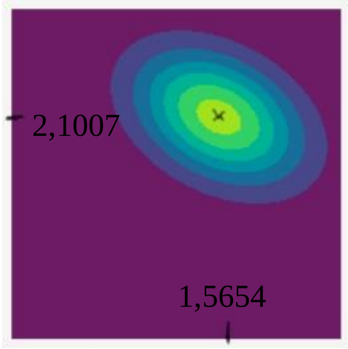
NORMALIZAR

$C=1$: $P(C=1|x_1) = \frac{P(C=1,x_1)}{P(C=1,x_1)+P(C=2,x_1)} = \frac{0,1114}{0,1114+2,832\times10^{-10}} \approx 1$

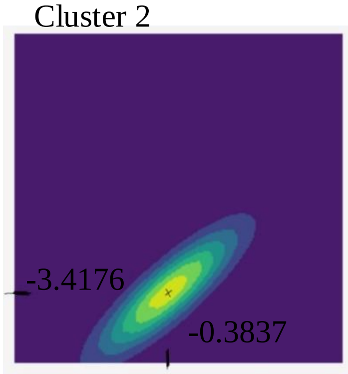
$C=2$: $P(C=2|x_1) = \frac{P(C=2,x_1)}{P(C=2,x_1)+P(C=1,x_1)} = \frac{2,832\times10^{-10}}{2,832\times10^{-10}+0,1114} \approx 0$

1

Cluster 1




Cluster 2




$x_2 = \begin{bmatrix}-1\\-4\end{bmatrix}$ $\boxed{\text{Para } c=1}$

→ prior: $P(C=1) = 0,7$

→ likelihood: $P(x|C=1) = 2,239\times10^{-17}$

→ joint probability $= 1,567\times10^{-17}$

$\boxed{\text{Para } c=2}$

→ prior: $P(C=2) = 0,3$

→ likelihood: $P(x|C=2) = 0,0796$

→ joint probability $= 0,0239$

NORMALIZAR

$C=1$: $P(C=1|x_2) \approx 0$

$C=2$: $P(C=2|x_2) \approx 1$

$x_3 = \begin{bmatrix}-1\\2\end{bmatrix}$ $\boxed{\text{Para } c=1}$

→ prior: $P(C=1) = 0,7$

→ likelihood: $P(x|C=1) = 0,00024$

→ joint probability $= 0,00017$

$\boxed{\text{Para } c=2}$

→ prior: $P(C=2) = 0,3$

→ likelihood: $P(x|C=2) = 9,8206\times10^{-6}$

→ joint probability $= 2,9462\times10^{-6}$

NORMALIZAR

$C=1$: $P(C=1|x_3) = 0,9827$

$C=2$: $P(C=2|x_3) = 0,0173$

$x_4 = \begin{bmatrix}4\\0\end{bmatrix}$ $\boxed{\text{Para } c=1}$

→ prior: $P(C=1) = 0,7$

→ likelihood: $P(x|C=1) = 7,226\times10^{-6}$

→ joint probability $= 5,058\times10^{-6}$

$\boxed{\text{Para } c=2}$

→ prior $P(C=2) = 0,3$

→ likelihood: $P(x|C=2) = 2,814\times10^{-6}$

→ joint probability $= 8,441\times10^{-7}$

NORMALIZAR

$C=1$. $P(C=1|x_4) = 0,8570$

$C=2$: $P(C=2|x_4) = 0,143$

2

$\mu_1 = \frac{1\times\begin{bmatrix}2\\4\end{bmatrix}+0,9827\begin{bmatrix}-1\\2\end{bmatrix}+0,857\begin{bmatrix}4\\0\end{bmatrix}}{1+0,9827+0,857} = \begin{bmatrix}1,5654\\2,1007\end{bmatrix}$

$\mu_2 = \frac{1\times\begin{bmatrix}-1\\-4\end{bmatrix}+0,0173\begin{bmatrix}-1\\2\end{bmatrix}+0,143\begin{bmatrix}4\\0\end{bmatrix}}{1+0,0173+0,143} = \begin{bmatrix}-0,3838\\-3,4176\end{bmatrix}$

$\Sigma_{11} = \frac{1(2-1,5654)^2+0(-1-1,5654)^2+0,9827(-1-1,5654)^2+0,857(4-1,5654)^2}{1+0+0,9827+0,857}$

$= 4,1328$

$\Sigma_{12}=\Sigma_{21} = \frac{1\times(2-1,5654)(4-2,1007)+0\times(-1-1,5654)(-4-2,1007)+0,9827(-1-1,5654)(2-2,1007)+0,857(4-1,5654)(0-2,1007)}{1+0+0,9827+0,857}$

$= -1,1634$

$\Sigma_{22} = \frac{1\times(4-2,1007)^2+0(-4-2,1007)^2+0,9827(2-2,1007)^2+0,857(0-2,1007)^2}{1+0+0,9827+0,857} =$

$= 2,6056$

$P(X|C=1) = N\left(\mu_1 = \begin{bmatrix}1,5654\\2,1007\end{bmatrix}, \Sigma_1 = \begin{bmatrix}4,1328 & -1,1634\\-1,1634 & 2,6056\end{bmatrix}\right)$

$\Sigma_{11} = \frac{0(2+0,3838)^2+1(-1+0,3838)^2+0,0173(-1+0,3838)^2+0,143(4+0,3838)^2}{0+1+0,0173+0,143}$

$= 2,7014$

$\Sigma_{12}=\Sigma_{21} = \frac{0\times(2+0,3838)(4+3,4176)+1(-1+0,3838)(-4+3,4176)+0,0173(-1+0,3838)(2+3,1476)+0,143(4+0,3838)(0+3,1476)}{0+1+0,0173+0,143}$

$= 2,1060$

$\Sigma_{22} = \frac{0(4+3,4176)^2+1\times(-4-3,4176)^2+0,0173(2+3,4176)^2+0,143(0+3,4176)^2}{0+1+0,0173+0,143} =$

$= 2,1694$

$P(X|C=2) = N\ \mu_2 \begin{bmatrix}-0,3838\\-3,4176\end{bmatrix}, \Sigma_2 = \begin{bmatrix}2,7014 & 2,1060\\2,1060 & 2,1694\end{bmatrix}$

$P(C=1) = \frac{P(C=1|x_1)+P(C=1|x_2)+P(C=1|x_3)+P(C=1|x_4)}{P(C=1|x_1)+P(C=1|x_2)+P(C=1|x_3)+P(C=1|x_4)+P(C=2|x_1)+P(C=2|x_2)+P(C=2|x_3)+P(C=2|x_4)} =$

$= 0,710$

$P(C=0) = 1-P(C=1) = 0,290$

3

Aprendizagem 2021/22

**Homework I – Group 117**

## Answer 2

2 - $x_1, x_3$ e $x_4 \in$ Cluster 1

$x_2 \in$ Cluster 2

Silhauette para $x_1$

$a(x_1) = \frac{1}{2}(\|x_1 - x_3\|_2 + \|x_1 - x_4\|_2) = \frac{1}{2}\left(\sqrt{(2+1)^2+(4-2)^2} + \sqrt{(2-4)^2+(4-0)^2}\right) = 4{,}039$

$b(x_1) = \|x_1 - x_2\|_2 = \sqrt{(2+1)^2+(4+4)^2} = 8{,}544$

$s(x_1) = 1 - \frac{a(x_1)}{b(x_1)} = 1 - \frac{4{,}039}{8{,}544} = 0{,}527$

Silhauette para $x_2$

$a(x_2) = 0$

$b(x_2) = \frac{1}{3}(\|x_2 - x_1\|_2 + \|x_2 - x_3\|_2 + \|x_2 - x_4\|_2) = 6{,}982$

$s(x_2) = 1 - \frac{a(x_2)}{b(x_2)} = 1 - \frac{0}{6{,}982} = 1$

Silhauette para $x_3$

$a(x_3) = \frac{1}{2}(\|x_3 - x_1\|_2 + \|x_3 - x_4\|_2) = 4{,}495$

$b(x_3) = \|x_3 - x_2\|_2 = 6$

$s(x_3) = 1 - \frac{a(x_3)}{b(x_3)} = 1 - \frac{4{,}495}{6} = 0{,}251$

Silhauette para $x_4$

$a(x_4) = \frac{1}{2}(\|x_4 - x_1\|_2 + \|x_4 - x_3\|_2) = 4{,}929$

$b(x_4) = \|x_4 - x_2\|_2 = 6{,}403$

$s(x_4) = 1 - \frac{a(x_4)}{b(x_4)} = 0{,}230$

Silhauette de $C_1$

$s(C_1) = \frac{s(x_1) + s(x_3) + s(x_4)}{3} = \frac{0{,}527 + 0{,}251 + 0{,}230}{3} = 0{,}336$

Silhauette de $C_2$

$s(C_2) = \frac{s(x_2)}{1} = \frac{1}{1} = 1$

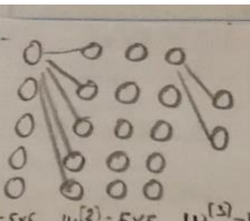
Silhauette de C

$s(C) = \frac{s(C_1) + s(C_2)}{2} = \frac{0{,}336 + 1}{2} = 0{,}668$

## Answer 3-a

i-

3 - a) i-



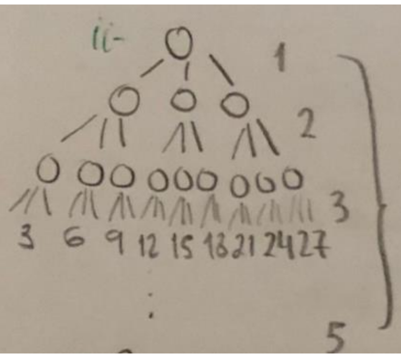
Total parâmetros = $3 \times (5 \times 5 + 5 \times 1) + 2 \times 5 + 2 = 102$

$w^{[1]} = 5 \times 5$ $w^{[2]} = 5 \times 5$ $w^{[3]} = 5 \times 5$ $w^{[4]} = 2 \times 5$

$b^{[1]} = 5 \times 1$ $b^{[2]} = 5 \times 1$ $b^{[3]} = 5 \times 1$ $b^{[4]} = 2 \times 1$

ii-

ii-




$3^5 = 243$ total parâmetros

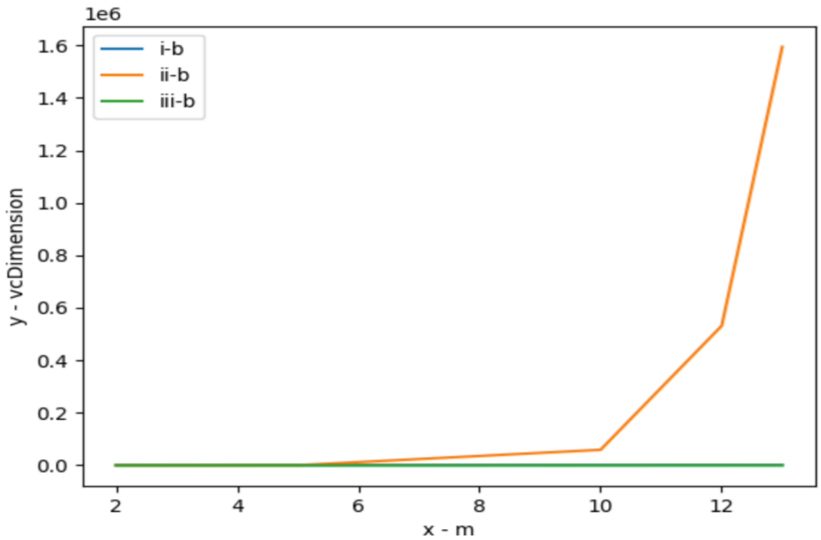
**iii-**

iii- $P(C=0) = 1 - P(C=1)$ — 1 parametro

$N(\mu, \Sigma)$ $\quad \mu = \begin{bmatrix} \mu_1 \\ \mu_2 \\ \vdots \\ \mu_5 \end{bmatrix}$ = 5 parametros $\quad \Sigma = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{array} \begin{array}{c} \begin{array}{ccccc} 1 & 2 & 3 & 4 & 5 \end{array} \\ \left[\begin{array}{ccccc} \cdot & \cdot & \cdot & \cdot & \cdot \\ & \cdot & \cdot & \cdot & \cdot \\ & & \cdot & \cdot & \cdot \\ & & & \cdot & \cdot \\ & & & & \cdot \end{array}\right] \end{array}$ 15 param.

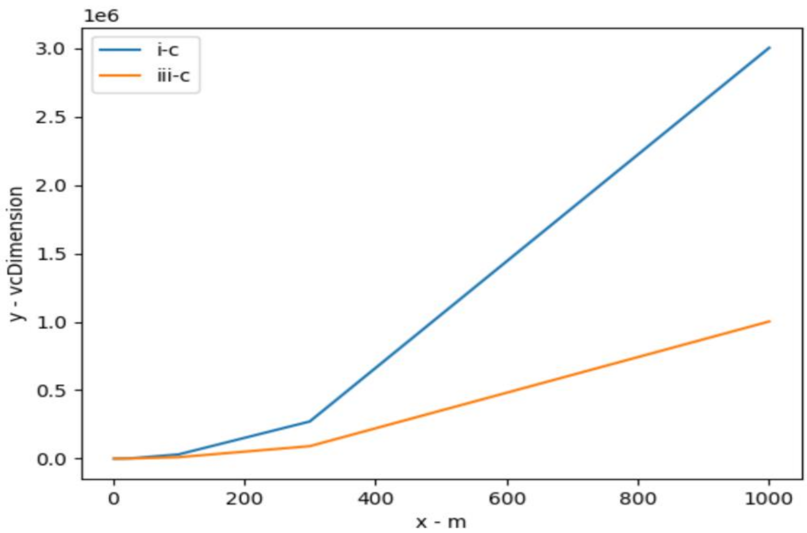
$1 + (15+5) \times 2 = 41$ parâmetros

## Answer 3-b




Pela observação do gráfico, verifica-se que a variação da vcDimension da decision tree (ii), aumenta exponencialmente, e verifica-se um aumento abrupto a partir de data dimensionality = 10 face à MLP com 3 hidden layers e ao Bayesian Classifier com uma multivariate Gaussian likelihood.

## Answer 3-c




Pela observação do gráfico, verifica-se que um maior aumento da vcDimension no MLP Classifier com 3 hidden layers do que no Bayesian Classifier com uma multivariate Gaussian likelihood a partir da data dimensionality = 100.

Aprendizagem 2021/22

**Homework I – Group 117**

## II. Programming and critical analysis

### Answer 4

```
ECR K = 2
13.5
ECR k = 3
6.666666666666666
```

```
Silhouette K = 2
0.5967981179111456
Silhouette K = 3
0.5245427800706391
```

**a)**

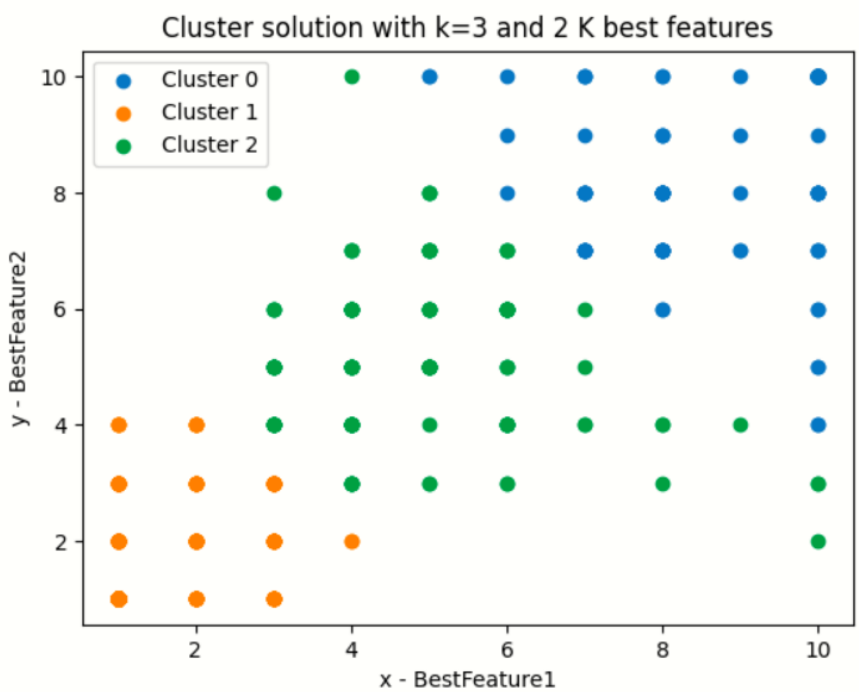
Pelos resultados acima apresentados verifica-se que no algoritmo kMeans, k = 3 apresenta um melhor ECR value que k = 2 para a nossa data.

Sendo o ECR a média dos pontos mal classificados concluimos que ao adicionarmos um novo cluster vai existir uma maior margem para classificação de pontos, e portanto menos pontos mal classificados, assim é natural que o ECR seja mais pequeno para k = 3.

**b)**

Pelos resultados acima apresentados verifica-se que no algoritmo kMeans, k = 2 apresenta uma melhor silhouette que k = 3 para a nossa data. Isto deve-se ao facto de para k = 2 os clusters serem mais compactos e estarem mais separados entre si.

### Answer 5



## Answer 6

```
ECR Ex5
11.666666666666666
Silhouette Ex5
0.7074226869204926
```

No exercício 5 verifica-se que a silhouette do algoritmo kMeans com k = 2 e apenas selecionando as 2 melhores features da nossa data segundo a mutual information é bastante boa, isto significa que os cluster são compactos e estão afastados entre si, algo que se pode verificar pelo gráfico apresentado na resposta 5.

Quanto ao ECR value verifica-se que este é melhor do que para k = 2 usando toda a data, mas pior do que para k = 3 usando toda a data.

Sendo o ECR a média dos pontos mal classificados concluimos que ao adicionarmos um novo cluster vai existir uma maior margem para classificação de pontos, e portanto menos pontos mal classificados, assim é natural que o ECR seja mais pequeno do que para k = 2. Dado que no exercício 5 apenas se selecionam as duas melhores features, a nossa data torna-se mais imprecisa o que leva a um maior grupo de pontos mal classificados, portanto é normal que o ECR value do exercício 5 seja maior que o de k = 3.

Aprendizagem 2021/22
**Homework I – Group 117**

# III. APPENDIX

Paste your programming code here using Consolas 9pt or 10pt.

Use **highlighting** or colored text to facilitate the analysis by your faculty hosts.

```python
# Grupo 117 Aprendizagem HomeWork 4
# Bernardo Castico ist196845
# Hugo Rita ist196870

import pandas as pd
import sklearn
from scipy.io import arff
from sklearn.cluster import KMeans
import numpy as np
from sklearn.metrics import silhouette_score
from sklearn.feature_selection import SelectKBest
from sklearn.feature_selection import mutual_info_classif
import matplotlib.pyplot as plt

def getDataToMatrix(lines):
    realLines = []
    data = []
    toDelete = []
    for i in range(len(lines)):
        if i > 11:
            realLines += [lines[i]]
    for i in range(len(realLines)):
        for j in range(len(realLines[i])):
            if realLines[i][j] == "benign\n":
                realLines[i][j] = "benign"
            elif realLines[i][j] == "malignant\n":
                realLines[i][j] = "malignant"
            elif realLines[i][j] == '?':
                toDelete += [i]
            else:
                realLines[i][j] = int(realLines[i][j])
    for i in range(len(realLines)):
        if i not in toDelete:
            data += [realLines[i]]
    return data

def splitData(list):
    a = []
    b = []
    for i in list:
        a.append(i[:-1])
```

```python
        b.append(i[-1])
    return [a,b]

def main():
    data, res2 = [],[]
    cluster02, cluster12, cluster03, cluster13, cluster23, cluster05, cluster15, cluster25 =
0,0,0,0,0,0,0,0
    Benign02, malignant02, Benign12, malignant12, Benign03, malignant03, Benign13, malignant13,
Benign23, malignant23 = 0,0,0,0,0,0,0,0,0,0
    Benign05, malignant05, Benign15, malignant15, Benign25, malignant25 = 0,0,0,0,0,0
    xCluster0, yCluster0, xCluster1, yCluster1, xCluster2, yCluster2 = [],[],[],[],[],[]
    with open("HW3-breast.txt") as f:
        lines = f.readlines()
    for line in lines:
        tmp = line.split(',')
        res2.append(tmp)
    data = getDataToMatrix(res2)

    trainDataSplit = splitData(data)

    kMeans2 = KMeans(n_clusters=2, random_state=0).fit(trainDataSplit[0])
    kMeans3 = KMeans(n_clusters=3, random_state=0).fit(trainDataSplit[0])

    kLabels2 = kMeans2.labels_
    kLabels3 = kMeans3.labels_

    for i in range(len(kLabels2)):
        if kLabels2[i] == 0:
            cluster02 += 1
            if trainDataSplit[1][i] == 'malignant':
                malignant02 += 1
            else:
                Benign02 += 1
        elif kLabels2[i] == 1:
            cluster12 += 1
            if trainDataSplit[1][i] == 'malignant':
                malignant12 += 1
            else:
                Benign12 += 1
        if kLabels3[i] == 0:
            cluster03 += 1
            if trainDataSplit[1][i] == 'malignant':
                malignant03 += 1
            else:
                Benign03 += 1
        elif kLabels3[i] == 1:
            cluster13 += 1
```

```
            if trainDataSplit[1][i] == 'malignant':
                malignant13 += 1
            else:
                Benign13 += 1
        elif kLabels3[i] == 2:
            cluster23 += 1
            if trainDataSplit[1][i] == 'malignant':
                malignant23 += 1
            else:
                Benign23 += 1

    ECR2 = 0.5*((cluster02-max(Benign02,malignant02)) + (cluster12-max(Benign12, malignant12)))
    ECR3 = (1/3)*((cluster03-max(Benign03,malignant03)) + (cluster13-max(Benign13, malignant13))+
(cluster23-max(Benign23, malignant23)))

    print("ECR K = 2")
    print(ECR2)
    print("ECR k = 3")
    print(ECR3)
    print("Silhouette K = 2")
    print(silhouette_score(trainDataSplit[0], kLabels2))
    print("Silhouette K = 3")
    print(silhouette_score(trainDataSplit[0], kLabels3))

    #EX5

    decision = SelectKBest(mutual_info_classif, k=2).fit(trainDataSplit[0], trainDataSplit[1])
    decisionTrainData = decision.transform(trainDataSplit[0])

    kMeans3Ex5 = KMeans(n_clusters=3, random_state=0).fit(decisionTrainData)
    kLabelsEx5 = kMeans3Ex5.labels_

    for i in range(len(kLabelsEx5)):
        if kLabelsEx5[i] == 0:
            cluster05 += 1
            if trainDataSplit[1][i] == 'malignant':
                malignant05 += 1
            else:
                Benign05 += 1
        elif kLabelsEx5[i] == 1:
            cluster15 += 1
            if trainDataSplit[1][i] == 'malignant':
                malignant15 += 1
            else:
                Benign15 += 1
        elif kLabelsEx5[i] == 2:
            cluster25 += 1
```

```
            if trainDataSplit[1][i] == 'malignant':
                malignant25 += 1
            else:
                Benign25 += 1

    ECR5 = (1/3)*((cluster05-max(Benign05,malignant05)) + (cluster15-max(Benign15, malignant15))+
(cluster25-max(Benign25, malignant25)))
    print("ECR Ex5")
    print(ECR5)
    print("Silhouette Ex5")
    print(silhouette_score(decisionTrainData, kLabelsEx5))

    for i in range(len(kLabelsEx5)):
        if kLabelsEx5[i] == 0:
            xCluster0 += [decisionTrainData[i][0]]
            yCluster0 += [decisionTrainData[i][1]]
        elif kLabelsEx5[i] == 1:
            xCluster1 += [decisionTrainData[i][0]]
            yCluster1 += [decisionTrainData[i][1]]
        else:
            xCluster2 += [decisionTrainData[i][0]]
            yCluster2 += [decisionTrainData[i][1]]

    plt.scatter(xCluster0, yCluster0, label="Cluster 0")
    plt.scatter(xCluster1, yCluster1, label="Cluster 1")
    plt.scatter(xCluster2, yCluster2, label="Cluster 2")

    plt.xlabel('x - BestFeature1')
    plt.ylabel('y - BestFeature2')
    plt.title('Cluster solution with k=3 and 2 K best features')

    # show a legend on the plot
    plt.legend()
    plt.show()

main()
```

**END**