Aprendizagem 2021/22

**Homework I – Group 047**

# I. Pen-and-paper

1) $P(c = 1) = 0.7\,,\ P(x_i \mid c = 1) = N\left(\begin{bmatrix} 2 \\ 4 \end{bmatrix}, \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}\right);\quad P(c = 2) = 0.3\,,\ P(x_i \mid c = 2) = N\left(\begin{bmatrix} -1 \\ -4 \end{bmatrix}, \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}\right)$

<u>E-Step</u>:

**1. Likelihoods**

| | $\boldsymbol{k = 1}$ | $\boldsymbol{k = 2}$ |
|---|---|---|
| $\boldsymbol{P(x_1 \mid c = k)}$ | 0.15915494309189535 | 9.43877951346626e-10 |
| $\boldsymbol{P(x_2 \mid c = k)}$ | 2.2390899578253236e-17 | 0.07957747154594767 |
| $\boldsymbol{P(x_3 \mid c = k)}$ | 0.00023927977920047084 | 9.82064017319871e-06 |
| $\boldsymbol{P(x_4 \mid c = k)}$ | 7.2256232377243294e-06 | 2.8136605178593184e-06 |

**2. Joint Probabilities**

$P(x_i\,, c = k) = P(c = k) \times P(x_i \mid c = k)$

| | $\boldsymbol{k = 1}$ | $\boldsymbol{k = 2}$ |
|---|---|---|
| $\boldsymbol{P(x_1\,, c = k)}$ | 0.11140846016432673 | 2.831633854039878e-10 |
| $\boldsymbol{P(x_2\,, c = k)}$ | 1.5673629704777265e-17 | 0.023873241463784303 |
| $\boldsymbol{P(x_3\,, c = k)}$ | 0.00016749584544032957 | 2.946192051959613e-06 |
| $\boldsymbol{P(x_4\,, c = k)}$ | 5.05793626640703e-06 | 8.440981553577955e-07 |

**3. Normalized Posteriors**

$$P(c = k\,, x_i) = \frac{P(x_i\,, c = k)}{\sum_k P(x_i\,, c = k)}$$

| | $\boldsymbol{k = 1}$ | $\boldsymbol{k = 2}$ |
|---|---|---|
| $\boldsymbol{P(c = k\,, x_1)}$ | 0.9999999974583315 | 2.541668597399302e-09 |
| $\boldsymbol{P(c = k\,, x_2)}$ | 6.565354658081997e-16 | 0.9999999999999992 |
| $\boldsymbol{P(c = k\,, x_3)}$ | 0.9827144048774182 | 0.01728559512258177 |
| $\boldsymbol{P(c = k\,, x_4)}$ | 0.8569818311724802 | 0.1430181688275199 |

<u>M-Step</u>:

**1. Estimate new priors**

$$P(c = k) = \frac{\sum_i P(c = k\,, x_i)}{\sum_k \sum_i P(c = k\,, x_i)}$$

| | $\boldsymbol{k = 1}$ | $\boldsymbol{k = 2}$ |
|---|---|---|
| $\boldsymbol{P(c = k)}$ | 0.7099240583770576 | 0.29007594162294237 |

**2. Estimate new means**

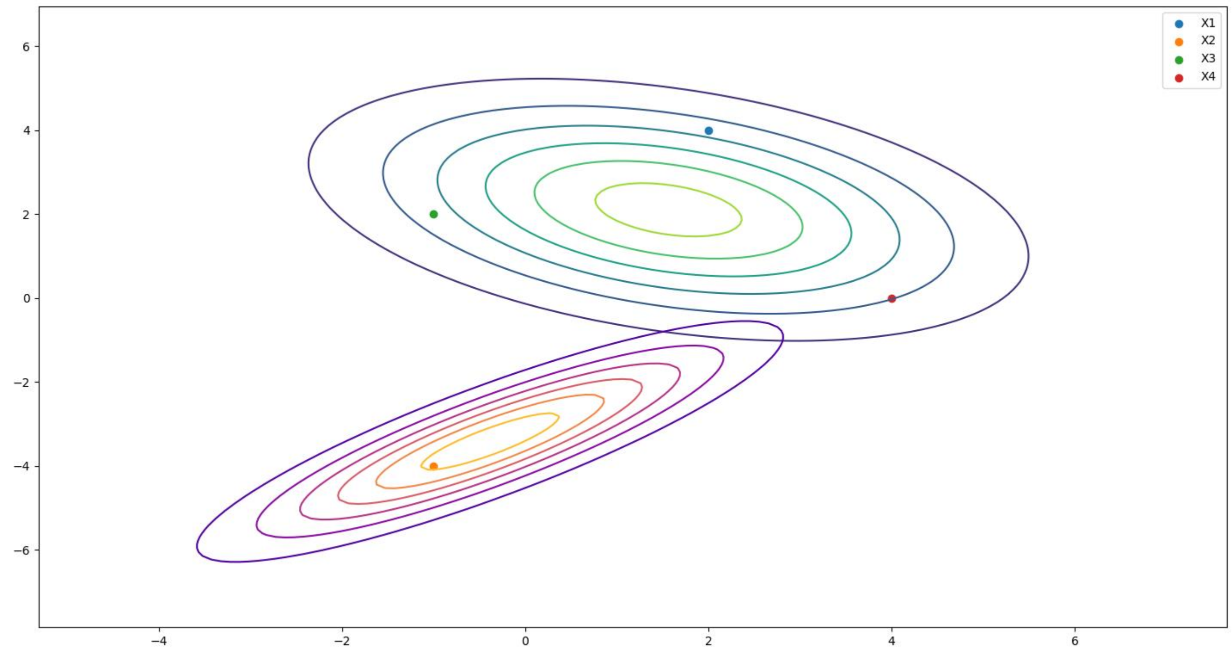
$$\mu_k = \frac{\sum_i P(c = k\,, x_i) \times\ x_i}{\sum_i P(c = k\,, x_i)}$$

| | $\boldsymbol{k = 1}$ | $\boldsymbol{k = 2}$ |
|---|---|---|
| $\boldsymbol{\mu_k}$ | $\begin{bmatrix} 1.56538325 \\ 2.10072779 \end{bmatrix}$ | $\begin{bmatrix} -0.38370376 \\ -3.41757815 \end{bmatrix}$ |

**3. Estimate new covariance matrixes**

$$\Sigma_k^{(n,m)} = \frac{\sum_i P(c = k\,, x_i) \times (x_{in} - \mu_{kn}) \times (x_{im} - \mu_{km})}{\sum_i P(c = k\,, x_i)}$$

| | $\boldsymbol{k = 1}$ | $\boldsymbol{k = 2}$ |
|---|---|---|
| $\boldsymbol{\Sigma_k}$ | $\begin{bmatrix} 4.13282298 & -1.16336779 \\ -1.16336779 & 2.60560106 \end{bmatrix}$ | $\begin{bmatrix} 2.70166014 & 2.1062406 \\ 2.1062406 & 2.16924195 \end{bmatrix}$ |



**2)** From the previous exercise we know that $x_1, x_3, x_4 \in c_1$and $x_2 \in c_2$.

$$S(x_i) = 1 - \frac{a(x_i)}{b(x_i)},$$

$a(x_i) = Average\ euclidean\ distance\ betweeen\ x_i\ and\ the\ points\ of\ the\ same\ cluster\ (Cohesion)$

$b(x_i) = Average\ euclidean\ distance\ between\ x_i\ and\ the\ points\ from\ the\ other\ clusters\ (Separation)$

**Silhouette ($c_1$):**

$$S(x_1) = 1 - \frac{a(x_1)}{b(x_1)} = 1 - \frac{4.038843615231785}{8.54400374531753} = 0.5272891099275046$$

$$S(x_3) = 1 - \frac{a(x_3)}{b(x_3)} = 1 - \frac{4.495358041299246}{6.0} = 0.250773659783459$$

$$S(x_4) = 1 - \frac{a(x_4)}{b(x_4)} = 1 - \frac{4.928650381067042}{6.4031242374328485} = 0.23027412895504817$$

$$S(c_1) = \frac{S(x_1) + S(x_3) + S(x_4)}{3} = 0.3361122995553372$$

**Silhouette ($c_2$):**

$$S(c_2) = S(x_2) = 1 - \frac{a(x_2)}{b(x_2)} = 1 - \frac{0}{6.982375994250127} = 1$$

**Silhouette of the solution:**

$$S = \frac{S(c_1) + S(c_2)}{2} = 0.6680561497776686$$

**3)**

**a)** Knowing that the VC Dimension is a measure of the degrees of freedom of classifier, a good proxy of this is the number of parameters of the classifier. With **n** being the data dimensionality, we estimate the following VC Dimensions:

Aprendizagem 2021/22

**Homework I – Group 047**

**i.** MLP with a configuration of (n,n,n,n,2):

**Weights parameters**:

$W^{[1]} = W^{[2]} = W^{[3]} = n \times n \xrightarrow{total} 3n^2\ ,\ W^{[4]} = 2n$

**Bias parameters:**

$B^{[1]} = B^{[2]} = B^{[3]} = n \xrightarrow{total} 3n\ ,\ B^{[4]} = 2$

**Total parameters (VC Dimension):**

$3n^2 + 2n + 3n + 2 = 3n^2 + 5n + 2 \xrightarrow{n=5} 102$

**ii.** Decision tree with input variables where discretized using three bins:

Each level of the tree is going to have $3^n$ outcomes, therefore the VC Dimension is $3^n \xrightarrow{n=5} 245$

**iii.** Bayesian classifier with a multivariate Gaussian likelihood:

**Priors:**

There's 2 classes, therefore 2 priors, however by knowing 1 we can know the other, so we only need to estimate **1** value.

**Likelihood (Mean):**

There's n features, therefore exists **n** means for each class, in total **2n** means.

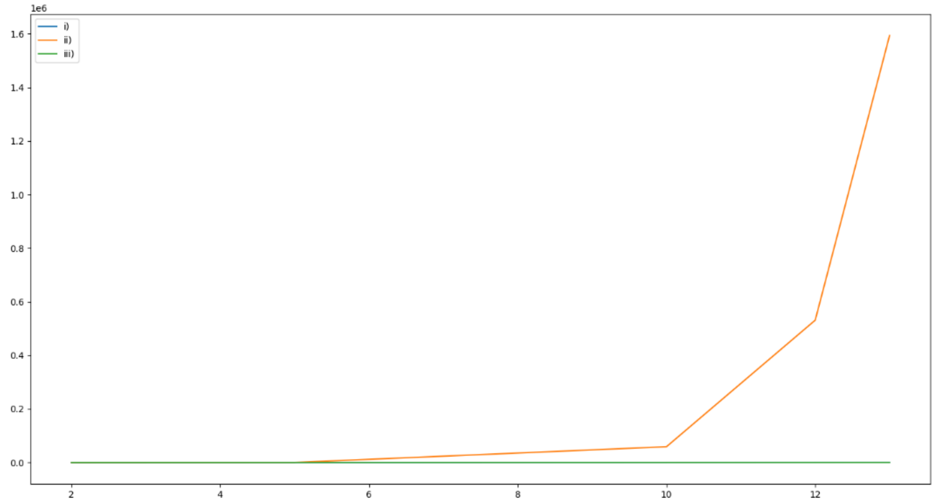
**Likelihood (Covariance Matrix):**

It's a n × n symmetrical matrix, one for each class, therefore it has $\mathbf{2}\left(\mathbf{n} + \frac{\boldsymbol{n^2 - n}}{\mathbf{2}}\right) = \boldsymbol{n^2} + \boldsymbol{n}$ unique values

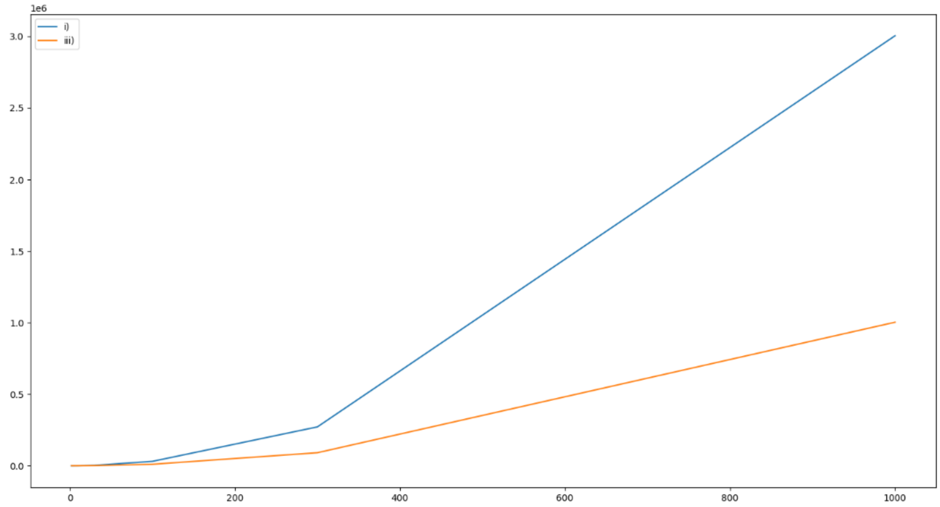
**Total parameters (VC Dimension):**

$1 + 2n + n^2 + n = n^2 + 3n + 1 \xrightarrow{n=5} 41$

**b)** As expected by expressions calculated before, the VC Dimension of the decision tree has a much more noticeable growth than the MLP and the Bayesian Classifier, making it more likely to overfitting.



**c)** Both classifiers have a similar growth rate, which can be explain by the similarity in their expression, being both type $O(n) = n^2$.



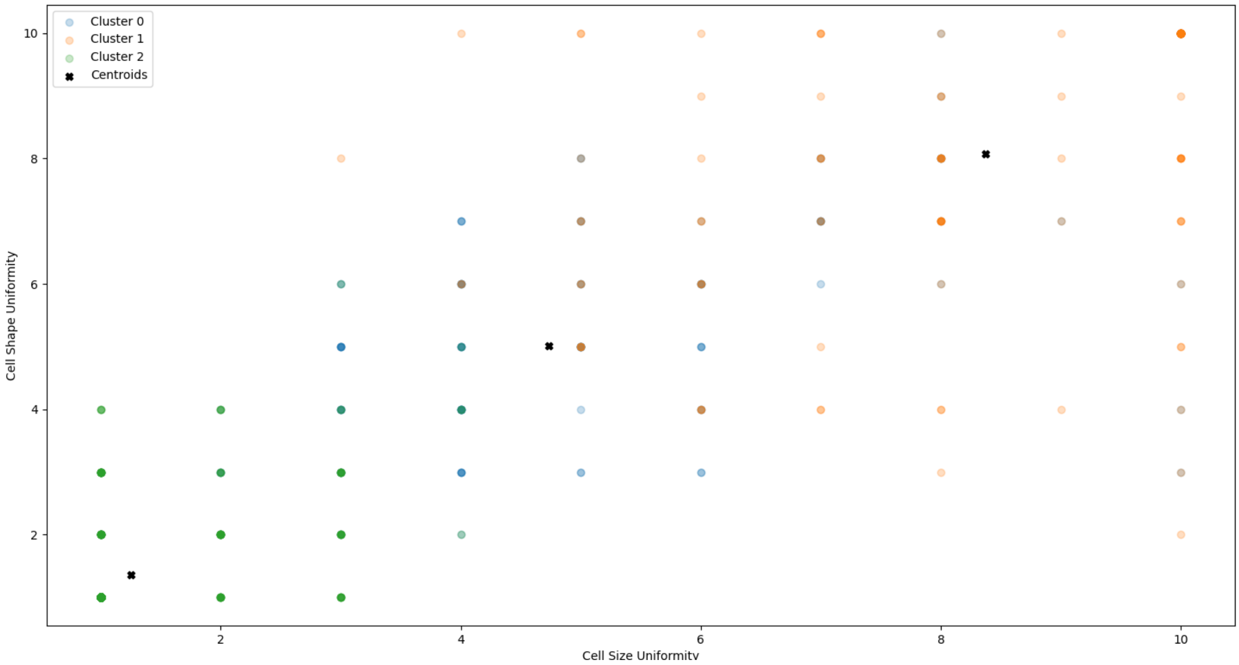
## II. Programming and critical analysis

**4)**

**a.** The produced solutions gave an ECR value of **13.5** for 2-means and **6.667** for 3-means.

**b.** The produced solutions gave a Silhouette value of **0.597** for 2-means and **0.525** for 3-means.

**5)**



**6)** The produced solution has poor cohesion, i.e., the points of the same cluster are very separated, and has an even worse separation, i.e., the distance between clusters is inexistent, seen by the overlapping points of different clusters.

# III. APPENDIX

```
from sklearn.feature_selection import mutual_info_classif, SelectKBest
from sklearn.metrics import silhouette_score
from sklearn.cluster import KMeans
import matplotlib.pyplot as plt
from scipy.io import arff

file = open("breast.w.arff", "r")
data, meta = arff.loadarff(file)

input = data[meta.names()[:-1]].tolist()
output = data["Class"].tolist()

#*4)
kMeans2 = KMeans(n_clusters=2, random_state=0).fit(input)
kMeans3 = KMeans(n_clusters=3, random_state=0).fit(input)
labels2 = kMeans2.labels_
labels3 = kMeans3.labels_

def ECR(labels, target, nLabels):
    clusters = []
    for i in range(nLabels):
        clusters += [[0,0]]

    for i in range(len(labels)):
        if target[i]==b'benign':
            clusters[labels[i]][0] += 1
        elif target[i]==b'malignant':
            clusters[labels[i]][1] += 1

    sum = 0
    for i in clusters:
        sum += (i[0] + i[1]) - max(i[0],i[1])

    return sum/nLabels

print("ECR k=2:", ECR(labels2, output, 2))
print("ECR k=3:", ECR(labels3, output, 3))

print("Silhouette k=2:", silhouette_score(input, labels2))
print("Silhouette k=3:", silhouette_score(input, labels3))

#*5)
n=3
selector = SelectKBest(mutual_info_classif, k=2)
kBest = selector.fit_transform(input,output)
features = selector.get_support(indices=True)
```

```
labels = kMeans3.labels_
centroids = kMeans3.cluster_centers_
print(centroids)

for c in range(n):
    cluster = [kBest[i] for i in range(len(labels)) if labels[i] == c]
    label = "Cluster {}".format(c)
    plt.scatter([x[0] for x in cluster],[y[1] for y in cluster], alpha=0.25, label=label)

plt.scatter([x[features[0]] for x in centroids],[y[features[1]] for y in centroids],marker='X',
c='black', label='Centroids')
plt.xlabel(meta.names()[features[0]].replace("_"," "))
plt.ylabel(meta.names()[features[1]].replace("_"," "))
plt.legend()
plt.show()
```

**END**