Aprendizagem 2021/22

Homework IV – Group 039

I. Pen-and-paper

1)

$$p(x_1|c_1=1)=\mathcal{N}\left(\begin{bmatrix}2\\4\end{bmatrix}\mid\begin{bmatrix}2\\4\end{bmatrix},\begin{bmatrix}1&0\\0&1\end{bmatrix}\right)=0.159155$$

$$p(x_2|c_1=1)=\mathcal{N}\left(\begin{bmatrix}-1\\-4\end{bmatrix}\mid\begin{bmatrix}2\\4\end{bmatrix},\begin{bmatrix}1&0\\0&1\end{bmatrix}\right)=2.23909e-17$$

$$p(x_3|c_1=1)=\mathcal{N}\left(\begin{bmatrix}-1\\2\end{bmatrix}\mid\begin{bmatrix}2\\4\end{bmatrix},\begin{bmatrix}1&0\\0&1\end{bmatrix}\right)=0.000239$$

$$p(x_4|c_1=1)=\mathcal{N}\left(\begin{bmatrix}4\\0\end{bmatrix}\mid\begin{bmatrix}2\\4\end{bmatrix},\begin{bmatrix}1&0\\0&1\end{bmatrix}\right)=7.22562e-06$$

$$p(x_1|c_2=1)=\mathcal{N}\left(\begin{bmatrix}2\\4\end{bmatrix}\mid\begin{bmatrix}-1\\-4\end{bmatrix},\begin{bmatrix}2&0\\0&2\end{bmatrix}\right)=9.43878e-10$$

$$p(x_2|c_2=1)=\mathcal{N}\left(\begin{bmatrix}-1\\-4\end{bmatrix}\mid\begin{bmatrix}-1\\-4\end{bmatrix},\begin{bmatrix}2&0\\0&2\end{bmatrix}\right)=0.0795775$$

$$p(x_3|c_2=1)=\mathcal{N}\left(\begin{bmatrix}-1\\2\end{bmatrix}\mid\begin{bmatrix}-1\\-4\end{bmatrix},\begin{bmatrix}2&0\\0&2\end{bmatrix}\right)=9.82064e-06$$

$$p(x_4|c_2=1)=\mathcal{N}\left(\begin{bmatrix}4\\0\end{bmatrix}\mid\begin{bmatrix}-1\\-4\end{bmatrix},\begin{bmatrix}2&0\\0&2\end{bmatrix}\right)=2.91366e-06$$

$$p(x_1,c_1=1)=\pi_1\cdot p(x_1|c_1=1)=0.111408$$

$$p(x_2,c_1=1)=\pi_1\cdot p(x_2|c_1=1)=1.56736e-17$$

$$p(x_3,c_1=1)=\pi_1\cdot p(x_3|c_1=1)=0.000167$$

$$p(x_4,c_1=1)=\pi_1\cdot p(x_4|c_1=1)=5.05794e-06$$

$$p(x_1,c_2=1)=\pi_2\cdot p(x_1|c_2=1)=2.83163e-10$$

$$p(x_2,c_2=1)=\pi_2\cdot p(x_2|c_2=1)=0.0238732$$

$$p(x_3,c_2=1)=\pi_2\cdot p(x_3|c_2=1)=2.94619e-06$$

$$p(x_4,c_2=1)=\pi_2\cdot p(x_4|c_2=1)=8.44098e-07$$

$$p(x_1)=\sum_{k=1}^{2}p(c_k=1,x_1)=0.11408$$

$$p(x_2)=\sum_{k=1}^{2}p(c_k=1,x_2)=0.0238732$$

$$p(x_3)=\sum_{k=1}^{2}p(c_k=1,x_3)=0.000170442$$

$$p(x_4)=\sum_{k=1}^{2}p(c_k=1,x_4)=5.90203e-06$$

$$\gamma(c_{11})=p(c_1=1|x_1)=\frac{p(c_1=1,x_1)}{p(x_1)}=1.0$$

$$\gamma(c_{21})=p(c_1=1|x_2)=\frac{p(c_1=1,x_2)}{p(x_2)}=6.56535\mathrm{e}-16$$

$$\gamma(c_{31})=p(c_1=1|x_3)=\frac{p(c_1=1,x_3)}{p(x_3)}=0.982714$$

$$\gamma(c_{41})=p(c_1=1|x_4)=\frac{p(c_1=1,x_4)}{p(x_4)}=0.856982$$

Aprendizagem 2021/22

Homework IV – Group 039

$$\gamma(c_{12}) = p(c_2 = 1|x_1) = \frac{p(c_2=1,x_1)}{p(x_1)} = 2.54167e - 09$$

$$\gamma(c_{22}) = p(c_2 = 1|x_2) = \frac{p(c_2=1,x_2)}{p(x_2)} = 1.0$$

$$\gamma(c_{32}) = p(c_2 = 1|x_3) = \frac{p(c_2=1,x_3)}{p(x_3)} = 0.0172856$$

$$\gamma(c_{42}) = p(c_2 = 1|x_4) = \frac{p(c_2=1,x_4)}{p(x_4)} = 0.143018$$

$$N_1 = \sum_{n=1}^{4} \gamma(c_{n1}) = 2.839696$$

$$N_2 = \sum_{n=1}^{4} \gamma(c_{n2}) = 1.603036$$

$$\mu_1 = \frac{1}{N_1} \cdot \sum_{n=1}^{4} \gamma(c_{n1}) \cdot x_n = \begin{bmatrix} 1.565383 \\ 2.100728 \end{bmatrix}$$

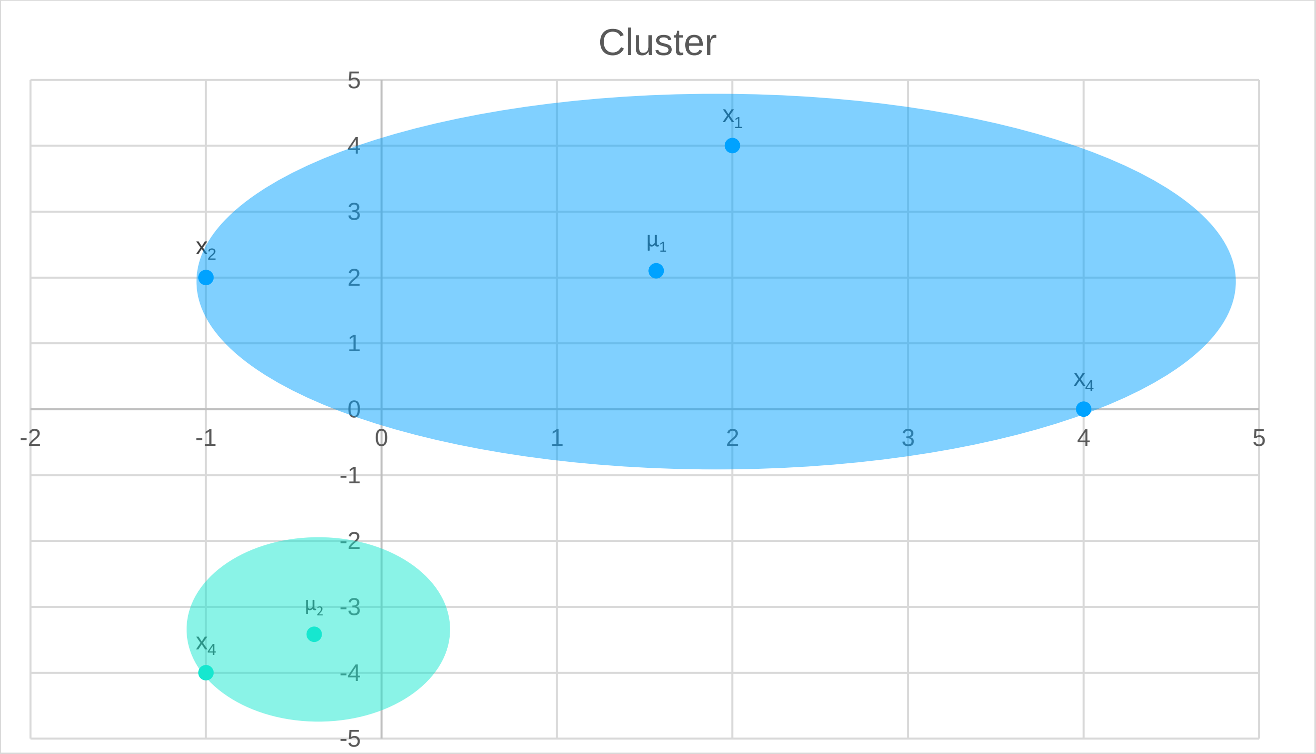
$$\mu_2 = \frac{1}{N_2} \cdot \sum_{n=1}^{4} \gamma(c_{n2}) \cdot x_n = \begin{bmatrix} -0.383704 \\ -3.417578 \end{bmatrix}$$

$$\Sigma_1 = \frac{1}{N_1} \cdot \sum_{n=1}^{4} \gamma(c_{n1}) \cdot (x_n - \mu_1) \cdot (x_n - \mu_1)^T = \begin{bmatrix} 4.132823 & -1.163368 \\ -1.163368 & 2.605601 \end{bmatrix}$$

$$\Sigma_2 = \frac{1}{N_2} \cdot \sum_{n=1}^{4} \gamma(c_{n2}) \cdot (x_n - \mu_2) \cdot (x_n - \mu_2)^T = \begin{bmatrix} 2.701660 & 2.106241 \\ 2.106241 & 2.169242 \end{bmatrix}$$

$$\pi_1 = p(c_1 = 1) = \frac{N_1}{4} = 0.709924$$

$$\pi_2 = p(c_2 = 1) = \frac{N_2}{4} = 0.290075$$



2)

$$S(x_1) = 1 - \frac{a(x_1)}{b(x_1)} = 1 - \frac{\frac{1}{2}(\|x_1 - x_3\|_2 + \|x_1 - x_4\|_2)}{\|x_1 - x_2\|_2} = 0.527289$$

$$S(x_2) = 1 - \frac{a(x_2)}{b(x_2)} = 1 - \frac{0}{\frac{1}{3}(\|x_2 - x_1\|_2 + \|x_2 - x_3\|_2 + \|x_2 - x_4\|_2)} = 1.000000$$

$$S(x_3) = 1 - \frac{a(x_3)}{b(x_3)} = 1 - \frac{\frac{1}{2}(\|x_3 - x_1\|_2 + \|x_3 - x_4\|_2)}{\|x_3 - x_2\|_2} = 0.250774$$

$$S(x_4) = 1 - \frac{a(x_4)}{b(x_4)} = 1 - \frac{\frac{1}{2}(\|x_4 - x_1\|_2 + \|x_4 - x_3\|_2)}{\|x_4 - x_2\|_2} = 0.230274$$

$$S(c_1) = \frac{S(x_1) + S(x_3) + S(x_4)}{3} = 0.336112$$

$$S(c_2) = S(x_2) = 1.000000$$

$$S(C) = \frac{S(c_1) + S(c_2)}{2} = 0.668056$$

Silhouette level is high (S(C) very close to 1), thus there is good evidence for clusters to be cohesive and well-separated.

3)

The VC dimension for a:

I. MLP with three hidden layers with as much nodes as the number of input variables is $N \times N + N \times 1 + 3(N \times N + N \times 1) + N \times N + N \times 1 = 5N^2 + 5N$

II. Decision tree (DT) assuming input variables are discretized using three bins is 3^N;

III. Bayesian (B) classifier with a multivariate Gaussian likelihood is $1 + 2 \times \left(N + N + \frac{N^2 - N}{2}\right) = 1 + 3N + N^2$

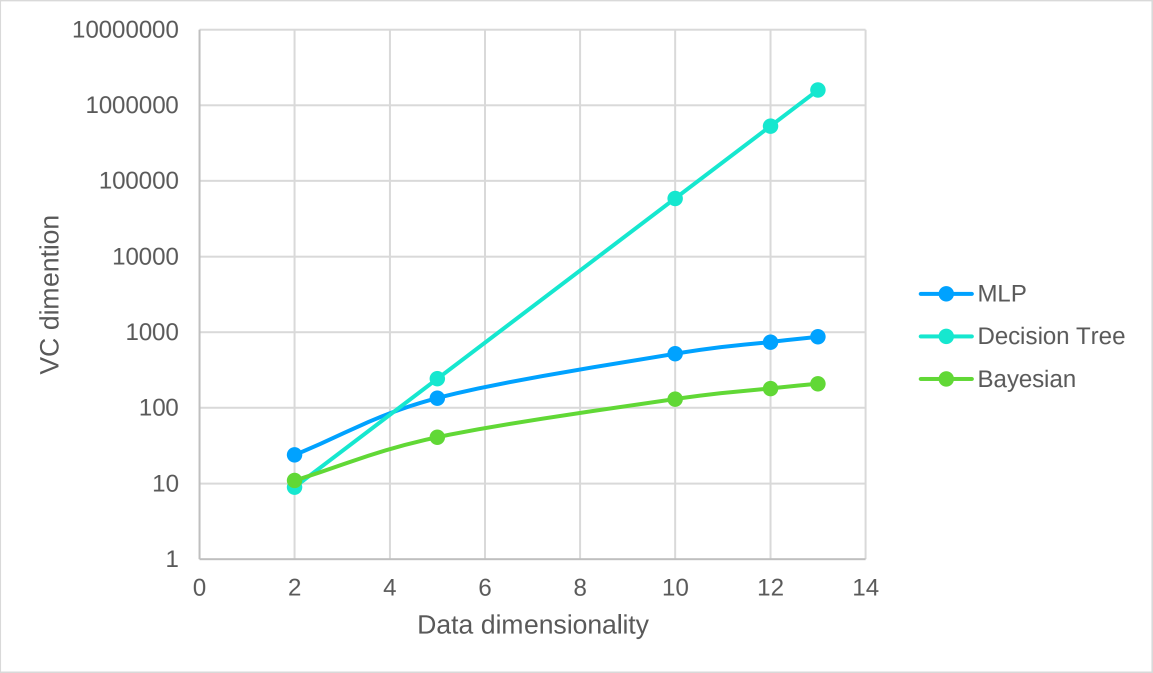
a.

$$d(MLP) = 5 \times 5^2 + 5 \times 5 = 150$$

$$d(DT) = 3^5 = 243$$

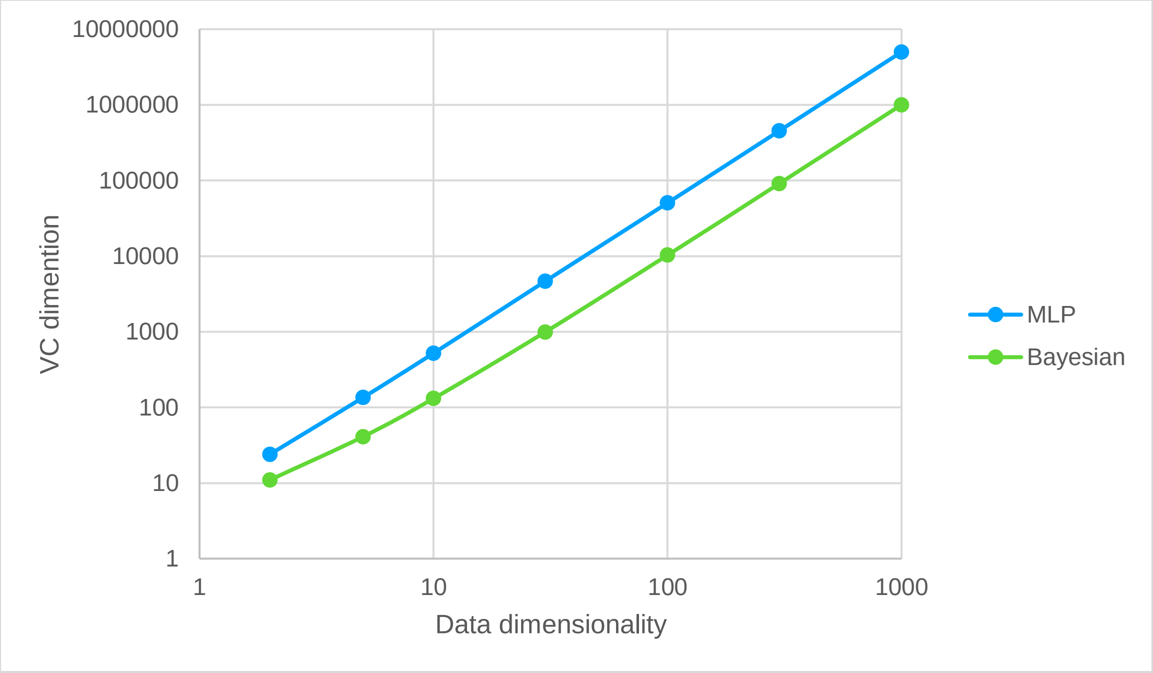
$$d(B) = 1 + 3 \times 5 + 5^2 = 41$$

b.



Given the decision tree's lower VC dimension for data dimensionalities lower than 5 its more likely to have higher in sample error and lower overfitting susceptibility, for higher values it is the most likely to fit the training data (lower in-sample-error) and has increased overfitting susceptibility, followed by the MLP and the Bayesian Classifier.

c.



The MLP, regardless of the data dimensionality always has higher VC dimension that the Bayesian Classifier, meaning that it is overall more likely to fit the training data (having lower in-sample-error) although having increased overfitting susceptibility.

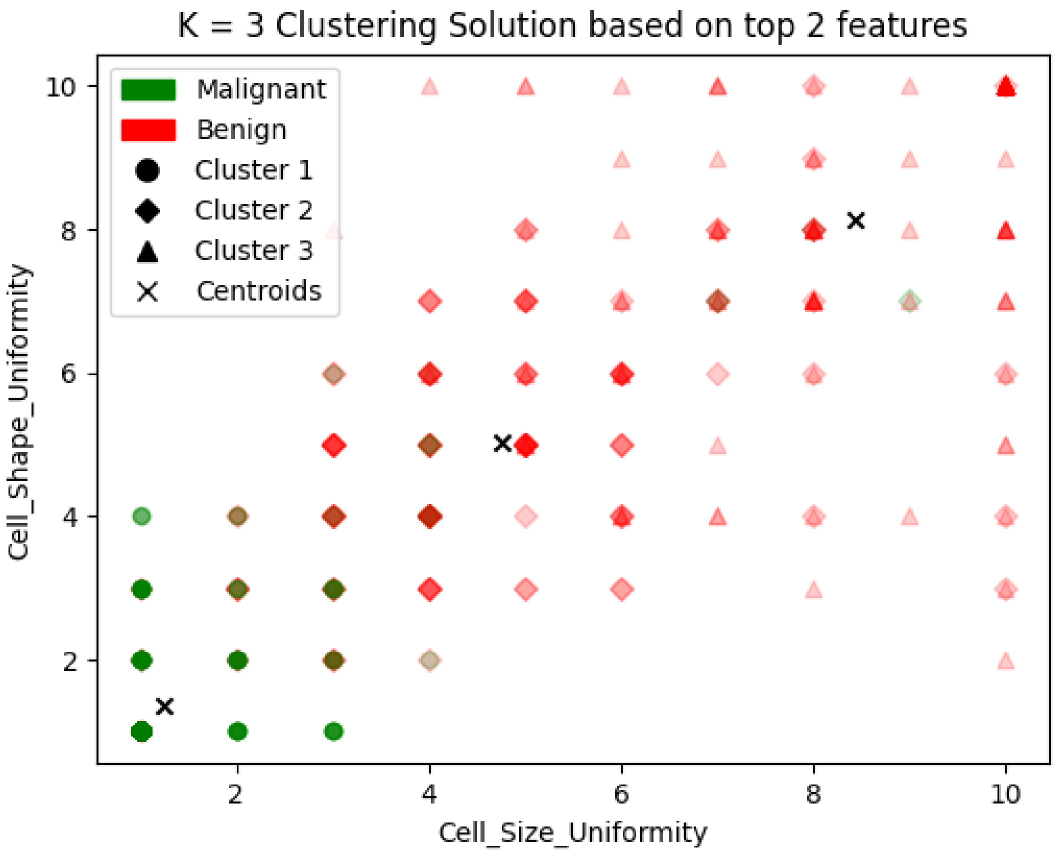
II. Programming and critical analysis

4)

a. Based on the external measure of each solution we conclude that in the solution with k=2 the clusters, overall, resemble less of their class than in the k=3 solution, meaning, each cluster in the k=3 solution is more adapted to a specific class.

b. Based on the silhouette values of each solution, we conclude that in the solution with k=2 the cluster are overall further apart from each other than in the k=3 solution.

5)



6) Analyzing the graph obtained in the previous question based on the two top features we can see that: in terms of classification, they classify reasonably well the data as benign/malignant; in terms of clustering, they don't describe a very good solution, with two of the cluster basically blended, working almost as a single cluster and the last one, even if clear, slightly overlapping the other group. Indicating that probably a better k (number of clusters) would be k=2.

III. APPENDIX

```python
import pandas as pd
import matplotlib.pyplot as plt
import matplotlib.patches as mpatches
import matplotlib.lines as mlines
import numpy as np
from scipy.io import arff
from sklearn.cluster import KMeans
from sklearn.metrics import silhouette_score
from sklearn.feature_selection import mutual_info_classif
from sklearn.feature_selection import SelectKBest

SEED = 39

# Extract Data
D_breast = pd.DataFrame( arff.loadarff( "breast.w.arff" )[0] )
# Elements array
X = D_breast.drop(columns=D_breast.columns[-1]).to_numpy().astype(int)
# Results array binarized
Y = D_breast[D_breast.columns[-1]].replace(b'benign', 0).replace(b'malignant', 1)

def quest4():

    def Cluster(k, Y_pred):
        cluster = np.where(Y_pred == k) [0]
        return cluster

    def ECR(k, Y, Y_pred):
        soma = 0

        for i in range(k):
            cluster = Cluster(i, Y_pred)
            total = [0]*k

            for ind in cluster: total[Y[ind]] += 1
            phi = max(total)
            soma += (len(cluster) - phi)

        return (1/k) * soma

    for k in [2, 3]:
        # Train model
        kmeans = KMeans(n_clusters=k, random_state=SEED).fit(X)
        Y_pred = kmeans.predict(X)

        print(f"\n------------ K = {k} ------------")
        # Error classification rate (external measure)
        ecr = ECR(k, Y, Y_pred)
        print(f"ECR = {ecr}")
        # Silhouette coefficient (internal measure)
        sil_score = silhouette_score(X, Y_pred, random_state=SEED)
        print(f"Silhouette = {sil_score}")
```

```python
def quest5():
    kmeans = KMeans(n_clusters=3, random_state=SEED)
    Y_pred = kmeans.fit_predict(X)

    Kbest = SelectKBest(mutual_info_classif, k=2)
    Kbest = Kbest.fit_transform(X, Y)

    def marker(i):
        if Y_pred[i] == 0: return "o"
        elif Y_pred[i] == 1: return "^"
        elif Y_pred[i] == 2: return "D"

    def color(i):
        if Y[i] == 0: return "green"
        elif Y[i] == 1: return "red"

    for i, point in enumerate(Kbest):
        plt.scatter(point[0], point[1], alpha=0.2, marker=marker(i), color=color(i))

    cluster_centers = kmeans.cluster_centers_
    for c in cluster_centers: plt.scatter(c[1], c[2], marker="x", color="black")

    plt.title("K = 3 Clustering Solution based on top 2 features")
    plt.xlabel("Cell_Size_Uniformity")
    plt.ylabel("Cell_Shape_Uniformity")

    green = mpatches.Patch(color='green', label='Malignant')
    red = mpatches.Patch(color='red', label='Benign')
    cl_1 = mlines.Line2D([], [], marker='o', color="black", linestyle='None',
markersize=8, label='Cluster 1')
    cl_2 = mlines.Line2D([], [], marker='D', color="black", linestyle='None',
markersize=6, label='Cluster 2')
    cl_3 = mlines.Line2D([], [], marker='^', color="black", linestyle='None',
markersize=7, label='Cluster 3')
    cl_c = mlines.Line2D([], [], marker='x', color="black", linestyle='None',
markersize=7, label='Centroids')

    plt.legend(handles=[green, red, cl_1, cl_2, cl_3, cl_c])

    plt.savefig("graph5")

quest4()
quest5()
```

END