Aprendizagem 2022/23

Homework IV – Group 009

I. Pen-and-paper

1) Denominando as observações da seguinte forma:

	y_1	y_2
x_1	1	2
x_2	-1	1
x_3	1	0

Expectation-Step:

Para x_1:

As joint probabilities são:

$$P(x_1,c=1)=P(x_1/c=1)P(c=1)=N\left(\begin{bmatrix}1\\1\end{bmatrix},\mu=\begin{bmatrix}2\\2\end{bmatrix},\Sigma_1=\begin{bmatrix}2&1\\1&2\end{bmatrix}\right)\pi_1=0.0658\,x\,0.5=0.0329$$

$$P(x_1,c=2)=P(x_1/c=2)P(c=2)=N\left(\begin{bmatrix}1\\1\end{bmatrix},\mu=\begin{bmatrix}0\\0\end{bmatrix},\Sigma_1=\begin{bmatrix}2&0\\0&2\end{bmatrix}\right)\pi_2=0.0228\,x\,0.5=0.0114$$

Portanto os normalized posteriors são:

$$P(c=1/x_1)=0.7428$$
$$P(c=2/x_1)=0.2572$$

Para x_2:

As joint probabilities são:

$$P(x_2,c=1)=P(x_2/c=1)P(c=1)=N\left(\begin{bmatrix}-1\\1\end{bmatrix},\mu=\begin{bmatrix}2\\2\end{bmatrix},\Sigma_1=\begin{bmatrix}2&1\\1&2\end{bmatrix}\right)\pi_1=0.0089\,x\,0.5=0.0045$$

$$P(x_2,c=2)=P(x_2/c=2)P(c=2)=N\left(\begin{bmatrix}1\\1\end{bmatrix},\mu=\begin{bmatrix}-1\\1\end{bmatrix},\Sigma_1=\begin{bmatrix}2&0\\0&2\end{bmatrix}\right)\pi_2=0.0483\,x\,0.5=0.0241$$

Portanto os normalized posteriors são:

$$P(c=1/x_2)=0.1558$$
$$P(c=2/x_2)=0.8442$$

Para x_3:

As joint probabilities são:

$$P(x_3,c=1)=P(x_3/c=1)P(c=1)=N\left(\begin{bmatrix}1\\1\end{bmatrix},\mu=\begin{bmatrix}1\\0\end{bmatrix},\Sigma_1=\begin{bmatrix}2&1\\1&2\end{bmatrix}\right)\pi_1=0.0338\,x0.5=0.0169$$

$$P(x_3,c=2)=P(x_3/c=2)P(c=2)=N\left(\begin{bmatrix}1\\1\end{bmatrix},\mu=\begin{bmatrix}1\\0\end{bmatrix},\Sigma_1=\begin{bmatrix}2&0\\0&2\end{bmatrix}\right)\pi_2=0.062\,x\,0.5=0.031$$

Portanto os normalized posteriors são:

$$P(c=1/x_3)=0.3529$$
$$P(c=2/x_3)=0.6471$$

Aprendizagem 2022/23

Homework IV – Group 009

Maximization-Step:

Atualização das médias:

$$\mu_c = \frac{\sum_{i=1}^{3} P(c/x_i)x_i}{\sum_{i=1}^{3} P(c/x_i)}$$

Para c=1:

$$\mu_1 = \frac{P(c=1/x_1)x_1 + P(c=1/x_2)x_2 + P(c=1/x_3)x_3}{P(c=1/x_1) + P(c=1/x_2) + P(c=1/x_3)}$$

$$\mu_1 = \frac{0.7428\begin{bmatrix}1\\2\end{bmatrix} + 0.1558\begin{bmatrix}-1\\1\end{bmatrix} + 0.3529\begin{bmatrix}1\\0\end{bmatrix}}{0.7428+0.1558+0.3529} = \begin{bmatrix}0.751\\1.3115\end{bmatrix}$$

Para c=2:

$$\mu_2 = \frac{P(c=2/x_1)x_1 + P(c=2/x_2)x_2 + P(c=2/x_3)x_3}{P(c=2/x_1) + P(c=2/x_2) + P(c=2/x_3)}$$

$$\mu_2 = \frac{0.2572\begin{bmatrix}1\\2\end{bmatrix} + 0.8442\begin{bmatrix}-1\\1\end{bmatrix} + 0.6471\begin{bmatrix}1\\0\end{bmatrix}}{0.2572+0.8442+0.6471} = \begin{bmatrix}0.0344\\0.777\end{bmatrix}$$

Atualização das Matrizes de Covariância:

$$\Sigma_c^{(i,j)} = \frac{\sum_{k=1}^{3} P(c/x_k)(a_{ki}-\mu_{ci})(a_{kj}-\mu_{cj})}{\sum_{k=1}^{3} P(c/x_k)}$$

Para c=1:

$$\Sigma_1^{(0,0)} = \frac{P(c=1/x_1)(a_{10}-\mu_{10})^2 + P(c=1/x_2)(a_{20}-\mu_{10})^2 + P(c=1/x_3)(a_{30}-\mu_{10})^2}{P(c=1/x_1) + P(c=1/x_2) + P(c=1/x_3)}$$

$$\Sigma_1^{(0,0)} = \frac{0.7428(1-0.751)^2 + 0.1558(-1-0.751)^2 + 0.3529(1-0.751)^2}{0.7428+0.1558+0.3529} = 0.4361$$

$$\Sigma_1^{(1,0)} = \Sigma_1^{(0,1)} = \frac{P(c=1/x_1)(a_{10}-\mu_{10})(a_{11}-\mu_{11}) + P(c=1/x_2)(a_{20}-\mu_{10})(a_{21}-\mu_{11}) + P(c=1/x_3)(a_{30}-\mu_{10})(a_{31}-\mu_{11})}{P(c=1/x_1) + P(c=1/x_2) + P(c=1/x_3)}$$

$$\Sigma_1^{(1,0)} = \Sigma_1^{(0,1)} = \frac{0.7428(1-0.751)(2-1.3115) + 0.1558(-1-0.751)(1-1.3115) + 0.3529(1-0.751)(0-1.3115)}{0.7428+0.1558+0.3529} = 0.0776$$

$$\Sigma_1^{(1,1)}=\frac{P(c=1/x_1)(a_{11}-\mu_{11})^2+P(c=1/x_2)(a_{21}-\mu_{11})^2+P(c=1/x_3)(a_{31}-\mu_{11})^2}{P(c=1/x_1)+P(c=1/x_2)+P(c=1/x_3)}$$

$$\Sigma_1^{(1,1)}=\frac{0.7428(2-1.3115)^2+0.1558(1-1.3115)^2+0.3529(0-1.3115)^2}{0.7428+0.1558+0.3529}=0.7785$$

Portanto temos:

$$\Sigma_1=\begin{bmatrix}0.4361 & 0.0776\\ 0.0776 & 0.7785\end{bmatrix}$$

<u>Para c=2:</u>

$$\Sigma_2^{(0,0)}=\frac{P(c=2/x_1)(a_{10}-\mu_{20})^2+P(c=2/x_2)(a_{20}-\mu_{20})^2+P(c=2/x_3)(a_{30}-\mu_{20})^2}{P(c=2/x_1)+P(c=2/x_2)+P(c=2/x_3)}$$

$$\Sigma_2^{(0,0)}=\frac{0.2572(1-0.0344)^2+0.8442(-1-0.0344)^2+0.6471(1-0.0344)^2}{0.2572+0.8442+0.6471}=0.9988$$

$$\Sigma_2^{(1,0)}=\Sigma_2^{(0,1)}=\frac{P(c=2/x_1)(a_{10}-\mu_{20})(a_{11}-\mu_{21})+P(c=2/x_2)(a_{20}-\mu_{20})(a_{21}-\mu_{21})+P(c=2/x_3)(a_{30}-\mu_{20})(a_{31}-\mu_{21})}{P(c=2/x_1)+P(c=2/x_2)+P(c=2/x_3)}$$

$$\Sigma_2^{(1,0)}=\Sigma_2^{(0,1)}=\frac{0.2572(1-0.0344)(2-0.777)+0.8442(-1-0.0344)(1-0.777)+0.6471(1-0.0344)(0-0.777)}{0.2572+0.8442+0.6471}=-0.2153$$

$$\Sigma_2^{(1,1)}=\frac{P(c=2/x_1)(a_{11}-\mu_{21})^2+P(c=2/x_2)(a_{21}-\mu_{21})^2+P(c=2/x_3)(a_{31}-\mu_{21})^2}{P(c=2/x_1)+P(c=2/x_2)+P(c=2/x_3)}$$

$$\Sigma_2^{(1,1)}=\frac{0.2572(2-0.777)^2+0.8442(1-0.777)^2+0.6471(0-0.777)^2}{0.2572+0.8442+0.6471}=0.4675$$

Portanto temos:

$$\Sigma_2=\begin{bmatrix}0.9988 & -0.2153\\ -0.2153 & 0.4675\end{bmatrix}$$

<u>Atualização dos Priors:</u>

$$\pi_k=P(c=k)=\frac{\sum_{i=1}^{3}P(c=k/x_i)}{\sum_{j=1}^{2}\sum_{i=1}^{3}P(c=j/x_i)}$$

<u>Para c=1:</u>

$$\pi_1=P(c=1)=\frac{P(c=1/x_1)+P(c=1/x_2)+P(c=1/x_3)}{P(c=1/x_1)+P(c=1/x_2)+P(c=1/x_3)+P(c=2/x_1)+P(c=2/x_2)+P(c=2/x_3)}$$

$$\pi_1=P(c=1)=\frac{0.7428+0.1558+0.3529}{0.7428+0.1558+0.3529+0.2572+0.8442+0.6471}=0.4172$$

Para c=2:

$$\pi_2 = P(c=2) = \frac{P(c=2/x_1)+P(c=2/x_2)+P(c=2/x_3)}{P(c=1/x_1)+P(c=1/x_2)+P(c=1/x_3)+P(c=2/x_1)+P(c=2/x_2)+P(c=2/x_3)}$$

$$\pi_2 = P(c=2) = \frac{0.2572+0.8442+0.6471}{0.7428+0.1558+0.3529+0.2572+0.8442+0.6471} = 0.5828$$

2) a)

As novas joint probabilities são:

Para x_1:

$$P(x_1,c=1) = P(x_1/c=1)P(c=1) = N\left(\begin{bmatrix}1\\2\end{bmatrix}, \mu=\begin{bmatrix}0.751\\1.3115\end{bmatrix}, \Sigma=\begin{bmatrix}0.4361 & 0.0776\\0.0776 & 0.7785\end{bmatrix}\right)\pi_1 = 0.19557\,x\,0.4172 = 0.0816$$

$$P(x_1,c=2) = P(x_1/c=2)P(c=2) = N\left(\begin{bmatrix}1\\2\end{bmatrix}, \mu=\begin{bmatrix}0.0344\\0.777\end{bmatrix}, \Sigma=\begin{bmatrix}0.9988 & -0.2153\\-0.2153 & 0.4675\end{bmatrix}\right)\pi_2 = 0.0135\,x\,0.5828 = 0.0079$$

Tendo em conta a suposição MAP temos:

$$P(x_1,c=1) > P(x_1,c=2) \Rightarrow x_1 \in c=1$$

Para x_2:

$$P(x_2,c=1) = P(x_2/c=1)P(c=1) = N\left(\begin{bmatrix}-1\\1\end{bmatrix}, \mu=\begin{bmatrix}0.751\\1.3115\end{bmatrix}, \Sigma=\begin{bmatrix}0.4361 & 0.0776\\0.0776 & 0.7785\end{bmatrix}\right)\pi_1 = 0.0082\,x\,0.4172 = 0.0034$$

$$P(x_2,c=2) = P(x_2/c=2)P(c=2) = N\left(\begin{bmatrix}-1\\1\end{bmatrix}, \mu=\begin{bmatrix}0.0344\\0.777\end{bmatrix}, \Sigma=\begin{bmatrix}0.9988 & -0.2153\\-0.2153 & 0.4675\end{bmatrix}\right)\pi_2 = 0.1436\,x\,0.5828 = 0.0837$$

Tendo em conta a suposição MAP temos:

$$P(x_2,c=1) < P(x_2,c=2) \Rightarrow x_2 \in c=2$$

Para x_3:

$$P(x_3,c=1) = P(x_3/c=1)P(c=1) = N\left(\begin{bmatrix}1\\0\end{bmatrix}, \mu=\begin{bmatrix}0.751\\1.3115\end{bmatrix}, \Sigma=\begin{bmatrix}0.4361 & 0.0776\\0.0776 & 0.7785\end{bmatrix}\right)\pi_1 = 0.0772\,x0.4172 = 0.0322$$

$$P(x_3,c=2) = P(x_3/c=2)P(c=2) = N\left(\begin{bmatrix}1\\0\end{bmatrix}, \mu=\begin{bmatrix}0.0344\\0.777\end{bmatrix}, \Sigma=\begin{bmatrix}0.9988 & -0.2153\\-0.2153 & 0.4675\end{bmatrix}\right)\pi_2 = 0.1048\,x\,0.5828 = 0.0611$$

Tendo em conta a suposição MAP temos:

$$P(x_3,c=1) < P(x_3,c=2) \Rightarrow x_3 \in c=2$$

b)

Como podemos verificar na alínea anterior, o cluster 1 é composto apenas por x_1 e o cluster 2 é composto por x_2 e x_3, portanto o maior cluster é o cluster 2.

A silhouette para o cluster 2 é dada por:

$$s(c=2) = \frac{s(x_2)+s(x_3)}{2}$$

Onde:

$$s(x_i)=\begin{cases}1-\frac{a(x_i)}{b(x_i)}, se\, a(x_i)\leq b(x_i)\\ \frac{b(x_i)}{a(x_i)}-1, se\, a(x_i)>b(x_i)\end{cases}$$

Portanto a silhouette de x_2 é:

$$s(x_2)=1-\frac{\|x_2-x_3\|_2}{\|x_2-x_1\|_2}=1-\frac{\sqrt{(-1-1)^2+(1-0)^2}}{\sqrt{(-1-1)^2+(1-2)^2}}=1-\frac{\sqrt{5}}{\sqrt{5}}=0$$

E a silhouette de x_3 é:

$$s(x_3)=\frac{\|x_3-x_1\|_2}{\|x_3-x_2\|_2}-1=\frac{\sqrt{(1-1)^2+(0-2)^2}}{\sqrt{(1+1)^2+(0-1)^2}}-1=\frac{2}{\sqrt{5}}-1=-0.1056$$

Logo a silhouete do cluster 2 é:

$$s(c=2)=\frac{0-0.1056}{2}=-0.0528$$

II. Programming and critical analysis

1)

Silhouette:

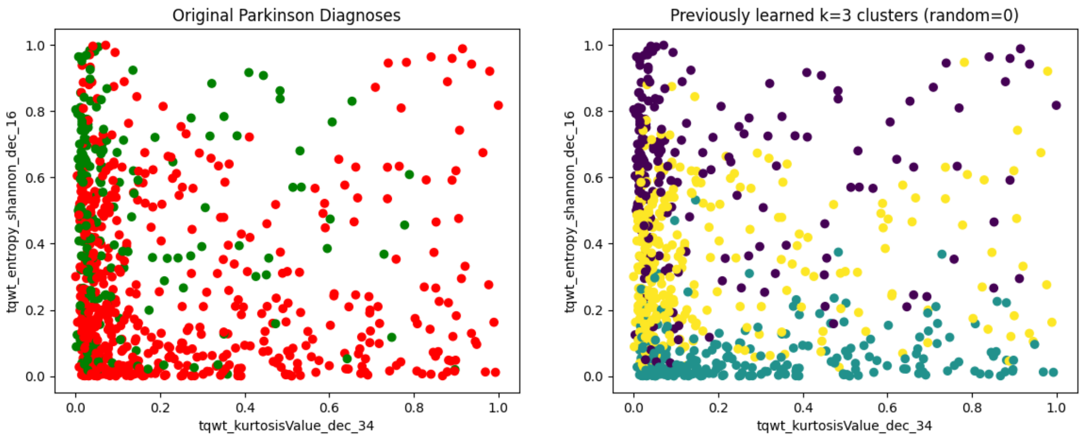
$$Silhouette_{Random=0}=0.1136$$
$$Silhouette_{Random=1}=0.1140$$
$$Silhouette_{Random=2}=0.1136$$

Purity:

$$Purity_{Random=0}=0.7672$$
$$Purity_{Random=1}=0.7632$$
$$Purity_{Random=2}=0.7672$$

2) O algoritmo do k-means não garante convergência para um ótimo global, sendo que os seus resultados dependem dos clusters iniciais. Tendo em conta que a escolha dos clusters iniciais vai depender do valor que passamos no random_state, podemos concluir que a causa para o não determinismo dos resultados obtidos são os diferentes valores do random_state, pois para cada valor podemos ter diferentes clusters iniciais o que, por sua vez, leva a diferentes resultados.

3)



4)

Number of principal components necessary to explain more than 80% of variability = 31

III. APPENDIX

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from scipy.io.arff import loadarff
from sklearn.cluster import KMeans
from sklearn.decomposition import PCA
from sklearn.metrics import silhouette_score, cluster
from sklearn.feature_selection import VarianceThreshold
from sklearn.preprocessing import MinMaxScaler

def purity_score(y_true, y_pred):
       confusion_matrix = cluster.contingency_matrix(y_true,y_pred)
       return np.sum(np.amax(confusion_matrix, axis=0)) / np.sum(confusion_matrix)

data = loadarff('pd_speech.arff')
df = pd.DataFrame(data[0])
df['class'] = df['class'].str.decode('utf-8')
X = df.drop('class',axis=1)
y = df['class']
scaler = MinMaxScaler().fit(X)
X_norm = scaler.transform(X)
df_X_norm = pd.DataFrame(X_norm,columns=X.columns)

kmeans_algo_0 = KMeans(n_clusters=3,random_state=0)
kmeans_algo_1 = KMeans(n_clusters=3, random_state=1)
kmeans_algo_2 = KMeans(n_clusters=3, random_state=2)

kmeans_model_0 = kmeans_algo_0.fit(X_norm)
kmeans_model_1 = kmeans_algo_1.fit(X_norm)
kmeans_model_2 = kmeans_algo_2.fit(X_norm)

y_pred_model_0 = kmeans_model_0.labels_
y_pred_model_1 = kmeans_model_1.labels_
y_pred_model_2 = kmeans_model_2.labels_
```

```
silhouette_model_0 = silhouette_score(X_norm, y_pred_model_0)
silhouette_model_1 = silhouette_score(X_norm, y_pred_model_1)
silhouette_model_2 = silhouette_score(X_norm, y_pred_model_2)

print("Silhouette (Random = 0) =", silhouette_model_0)
print("Silhouette (Random = 1) =", silhouette_model_1)
print("Silhouette (Random = 2) =", silhouette_model_2)

purity_model_0 = purity_score(y,y_pred_model_0)
purity_model_1 = purity_score(y,y_pred_model_1)
purity_model_2 = purity_score(y,y_pred_model_2)

print("Purity (Random = 0) =", purity_model_0)
print("Purity (Random = 1) =", purity_model_1)
print("Purity (Random = 2) =", purity_model_2)

selection = VarianceThreshold().fit(df_X_norm)
variances=selection.variances_
features = selection.feature_names_in_
d_feature_variance = {}
for i in range(len(features)):
       d_feature_variance[features[i]] = variances[i]

sort_d_feature_variance = sorted(d_feature_variance.items(), key=lambda x: x[1], reverse=True)
y_variable = df_X_norm[sort_d_feature_variance[0][0]]
x_variable = df_X_norm[sort_d_feature_variance[1][0]]

plt.figure(figsize=(14,5))
plt.subplot(121)
plt.scatter(x_variable,y_variable, c=y.map({'0':'green','1':'red'}))
plt.title("Original Parkinson Diagnoses")
plt.ylabel(sort_d_feature_variance[0][0])
plt.xlabel(sort_d_feature_variance[1][0])

plt.subplot(122)
plt.scatter(x_variable,y_variable,c=y_pred_model_0)
plt.title("Previously learned k=3 clusters (random=0)")
plt.ylabel(sort_d_feature_variance[0][0])
plt.xlabel(sort_d_feature_variance[1][0])
plt.show()

pca = PCA(svd_solver='full')
pca.fit(df_X_norm)

l_explained_variance = pca.explained_variance_ratio_
explained_variance = 0
n_components = 0
while explained_variance < 0.8:
       explained_variance += l_explained_variance[n_components]
       n_components+=1

print("Number of principal components necessary to explain more than 80% of variability =",
n_components)
```

END