PRÁTICA 8

1) Medidas de Distância.

A **Distância Euclidiana** entre dois vetores n-dimensionais x e y é definida como o escalar:

$$d_e(x,y) = \|x-y\| = \|y-x\| = \left[(x_1-y_1)^2 + \cdots + (x_n-y_n)^2\right]^{1/2}$$

esta expressão é a Norma da diferença entre os vetores e pode ser computada através da função do MatLab:

d = norm(x – y)

T_1: Considerar o Vetor de Características y = [53 23 44 55 02 13] padrão gerado através do processamento de uma imagem e o Vetor de Características x = [53 23 43 55 02 13] padrão este armazenado. Calcular a Distância Euclidiana entre os dois padrões.

Quando existem diversos padrões armazenados (por exemplo, representativos de uma base de imagens) estes podem ser descritos através de uma Matriz ***X(pxn)***, onde ***p*** é o número de padrões e ***n*** é o número de característica de cada padrão.

Exemplo_1: A Matriz ***X(10x6)*** apresenta um conjunto de 10 Vetores de Características (Padrões) referentes a 10 imagens de uma base. Cada vetor possui 6 características (ou descritores) extraídos de cada uma das imagens.

$$X = \begin{bmatrix} 41 & 05 & 04 & 52 & 30 & 33 \\ 09 & 39 & 37 & 49 & 43 & 41 \\ 36 & 30 & 10 & 11 & 29 & 47 \\ 06 & 59 & 42 & 27 & 01 & 05 \\ 01 & 19 & 46 & 06 & 16 & 02 \\ 19 & 40 & 07 & 13 & 22 & 47 \\ 56 & 38 & 21 & 20 & 03 & 05 \\ 53 & 17 & 38 & 04 & 47 & 37 \\ 55 & 43 & 56 & 54 & 08 & 60 \\ 25 & 04 & 18 & 57 & 21 & 38 \end{bmatrix}$$

Considerando que o Vetor de Características ***y*** deve ser verificado se existe na base ou se existe um Vetor na Base que mais se aproxima dele, pode-se calcular a Distância Euclidiana entre o Vetor ***y*** e cada linha da Matriz X (Base X), da seguinte maneira:

d = sqrt(sum(abs(X – repmat(y, p, 1).^2,2))

T_2: O arquivo *dados_X.dat* contém os dados de todos os Vetores de Características no formato de um Vetor Coluna de 60 elementos. Montar a Matriz (X) de Vetores de Características (Base X) no formato (10 x 6) (10 vetores x 6 características).

```
load dados_X.dat
nx = numel(dados_X)/6;
X1 = reshape(dados_X,6,nx);
 X = X1.';
```

E_1: Usando a função dada, calcular a Distância Euclidiana entre o Vetor de Características y1 = [09 43 37 49 41 39] e os padrões armazenados na base X. Fazer o mesmo para o Vetor de Características y2 = [53 17 38 04 47 37]. Aplicar o mesmo cálculo para o padrão y3 = [25 05 19 57 20 38]. Verificar, em cada caso, se o padrão pode ser dito como pertencente à Base X ou não.

E_2: Descrever, utilizando o Help do MatLab, a operação completa realizada pelo cálculo realizado no exercício E_1.

Um outro conjunto de 10 Vetores de Características formam a Base Y (10 x 6) que está armazenada no arquivo ***dados_Y.dat*** como um Vetor Coluna de (1 x 60).

$$Y = \begin{bmatrix} 14 & 21 & 26 & 29 & 36 & 48 \\ 19 & 24 & 43 & 34 & 39 & 05 \\ 58 & 48 & 52 & 47 & 12 & 06 \\ 32 & 09 & 04 & 52 & 29 & 03 \\ 29 & 35 & 19 & 57 & 42 & 55 \\ 01 & 07 & 56 & 11 & 37 & 38 \\ 20 & 22 & 11 & 19 & 48 & 42 \\ 14 & 24 & 60 & 06 & 18 & 48 \\ 10 & 57 & 28 & 03 & 05 & 21 \\ 13 & 59 & 37 & 56 & 03 & 47 \end{bmatrix}$$

T_4: Carregar os arquivos referentes a cada uma das bases e gerar as Bases X e Y de população de Vetores de Características no formato (10 x 6).

```
load dados_X.dat
nx = numel(dados_X)/6;
X1 = reshape(dados_X,nx,6);
 X = X1.';

load dados_Y.dat
ny = numel(dados_Y)/6;
Y1 = reshape(dados_Y,ny,6);
Y = Y1.';
```

Para calcular a Distância Euclidiana entre as duas populações de Vetores (X de dimensão p x n) e (Y de dimensão q x n) pode-se utilizar a função proposta em (Gonzalez;Woods:Eddins,2004) com a seguinte sintaxe:

D = sqrt(sum(abs(repmat(permute(X, [1 3 2]),[1 q 1]) - repmat(permute(Y, [3 1 2]),[p 1 1])).^2,3));

Onde ***D(i,j)*** é a Distância Euclidiana entre a ***i-ésima*** e a ***j-ésima*** linhas da população de vetores, ou seja, a Distância Euclidiana entre X(i,:) e Y(j,:).

Exemplo_2: Considerando a população de Vetores de Características a formada pela Base X, calcular a Matriz de Distâncias Euclidianas entre cada Vetor e outro da Base. A Matriz de Distâncias Euclidiana terá a diagonal formada por zeros equivalendo a distância entre o Vetor e ele mesmo. A função do MatLab que realiza esta operação tem a seguinte sintaxe:

DE = pdist(X,'euclidean');

Que calcula a Distância Euclidiana entre cada linha da Matriz X e as outras, colocando o resultado em um Vetor de dimensão (1 x (n/2.(n-1)).
Para gerar uma Matriz de Distâncias Euclidianas deve-se utilizar a função:

MDE = squareform(DE);

Que monta a Matriz de Distâncias, matriz quadrada onde o MDE(i,j) significa a Distância Euclidiana entre o Vetor ***i*** e o Vetor ***j*** da Base X.

E_3: a) Calcular a Matriz de Distâncias Euclidianas entre as duas populações de Vetores de Características (X e Y)
b) Calcular a Matriz de Distâncias Euclidianas entre os Vetores da Base X.
c) Calcular a Matriz de Distâncias Euclidianas entre os Vetores da Base Y.
d) Verificar que a função que calcula a Distância Euclidiana entre duas Bases fornece o mesmo resultado dos itens b) e c) se as Bases envolvidas forem as mesmas.

A **Distância de Mahalanobis** entre dois vetores n-dimensionais x e y é definida como o escalar:

$$d_m(x,y) = (y-x)^T C_x^{-1} (y-x)$$

onde $\boldsymbol{C_x}$ é a Matriz de Covariância da população de Vetores da Base X.
A função do MatLab que realiza esta operação tem a seguinte sintaxe:

DM = pdist(X,'mahalanobis');

Que calcula a Distância de Mahalanobis entre cada linha da Matriz X e as outras, colocando o resultado em um Vetor de dimensão (1 x (n/2.(n-1)).
Para gerar uma Matriz de Distâncias de Mahalanobis deve-se utilizar a função:

MDM = squareform(DM);

Que monta a Matriz de Distâncias, matriz quadrada onde o MDM(i,j) significa a Distância de Mahalanobis entre o Vetor ***i*** e o Vetor ***j*** da Base X.

A Distância de Mahalanobis calculada entre os Vetores da base Y e o centróide que representa os vetores da Base X (média de X ou protótipo de X) pode ser calculada como:

$$d_m(x,m_x) = (y-m_x)^T C_x^{-1} (y-m_x)$$

onde $\boldsymbol{m_x}$ é ao centróide da população de vetores da Base X.
A função do MatLab que realiza esta operação tem a seguinte sintaxe:

DM = mahal(Y,X)

E_4: a) Calcular a Matriz de Distâncias de Mahalanobis entre os Vetores da Base X.
b) Calcular a Matriz de Distâncias de Mahalanobis entre os Vetores da Base Y.
c) Calcular a Distância Mahalanobis dos Vetores da Base Y para o Vetor Protótipo representado pelo centróide da Base X.
Concluir a respeito.

E_5: Dado um novo Vetor de Características z = [20 30 05 50 43 02] determinar a qual base X ou Y ele está mais próximo, ou seja, determinar a que classe de Imagens (X ou Y) pertence a Imagem representada pelo Vetor z e qual é o grau de similaridade com os Vetores da Classe.

2) Análise de Agrupamentos (Cluster Analysis).

No MatLab a função ***Z = linkage(Y,'method')*** computa uma árvore de agrupamento hierárquico usando o algoritmo especificado na variável ***'method'***.

O ***'method'*** pode ser:

'single'	→ Ligação Simples (default)
'complete'	→ Ligação Completa
'average'	→ Ligação Média
'weighted'	→ Ligação Média Ponderada
'centroid'	→ Ligação Centróide (Y deve conter Distâncias Euclidianas)
'median'	→ Ligação Centro de Massa Ponderado
'ward'	→ Ligação Distância Quadrada Interna (algoritmo de mínima variância)

Uma outra função do MatLab, ***H = dendrogram(Z)*** plota um dendrograma do cluster hierárquico representado por ***Z***, onde ***Z*** é uma matriz ***(m-1) x 3*** , gerada através de uma Função de Ligação e ***m*** é o número de objetos no conjunto de dados original. A saída ***H***, é um vetor que produz as linhas do dendrograma.

Procedimento Básico para a Análise Hierárquica através do MatLab:

1) Encontrar a similaridade ou dissimilaridade entre cada par de objetos no conjunto de dados. (função ***pdist***)
2) Agrupar os objetos em uma Árvore Hierárquica de Grupos . (função ***linkage***)
3) Determinar onde dividir a Árvore Hierárquica em grupos. (função ***cluster***)

Exemplo_1:
Considere o conjunto de objetos (1 2 3 4 5) (pontos no espaço xy) dados pela **Figura 1**, ou seja:

X = [1 2; 2.5 4.5; 2 2; 4 1.5; 4 2.5]

X é a matriz composta dos Vetores de Características dos objetos (1 2 3 4 5).

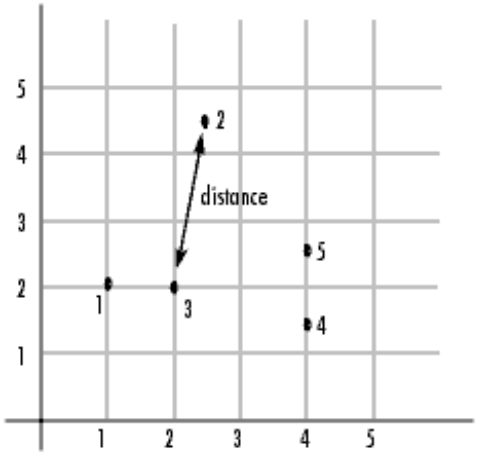

Figura 1 - Conjunto de pontos (objetos) exemplo

T_5: Encontrar a similaridade entre os objetos da população de Vetores dada na Figura 1 (gerar a Matriz de Distâncias, ou Matriz de Similaridades)

Y = pdist(X)

D = squareform(Y)

E_6: Definir as Ligações para agrupar os objetos/Vetores. Testar para os diferentes métodos. Listar a Matriz de Ligações e explicar suas linhas usando a Figura 1 como exemplo (Obs: ver Help do MatLab).

Z = linkage(Y)

E_7: Plotar o Dendrograma de Z. Explicar usando o Dendrograma e a Figura 1, a análise dos grupos.

H = dendrogram(Z)

Exemplo_2:

O arquivo ***M.dat*** refere-se a um vetor coluna com uma população de 1924 Vetores de Características no formato [6 x 1] concatenados em um único vetor de [11544 x 1].

T_6: Para ler o Arquivo M.dat e colocá-lo no formato de uma Matriz de Vetores de Características utilizar:

```
load M.dat
g = numel(M)/6;          %Número de Vetores de Características
V = reshape(M,g,6);      %Matriz dos dados no formato g x 6
```

E_8: Realizar a análise de Agrupamentos dos Vetores de Características (6 x 1) contidos no arquivo M.dat, obtendo:

a) ***A Matriz de Similaridades***
b) ***O Dendrograma***
c) ***Variar o Threshold e analisar os grupos gerados, obtendo através do Método de Ligação, os centróides (Vetores centróides) para os 3 maiores agrupamentos.***
d) ***Escolher k-centróides obtidos com a análise hierárquica anterior e aplicar o método K-means para tentar agrupar os vetores. Imprimir os valores dos centróides e os agrupamentos, quando o método convergir.***