TEMA 5 – RECONOCIMIENTO DE OBJETOS

……………………………………………………………………………

EJERCICIOS TEÓRICO-PRÁCTICOS

……………………………………………………………………………

EJERCICIO 1

A partir de los datos de la tabla, construir un Clasificador de Mínima Distancia Euclídea:

CLASE 1

PATRON	1	2	3	4	5	6	7	8	9	10
x_1	1	2	2	2	2	3	3	4	5	1
x_2	3	1	2	3	4	2	3	3	2	2

CLASE 2

PATRON	1	2	3	4	5	6	7	8	9	10
x_1	4	5	5	4	6	6	6	7	4	8
x_2	5	5	6	7	5	6	7	6	6	7

- Funciones de decisión de cada clase expresadas en función de una instancia X genérica definida por x1 y x2. Se deben presentar las expresiones teóricas de forma matricial sin desarrollar, presentando los valores de las variables que intervienen.

- Función discriminante entre clases expresadas en función de una instancia X genérica definida por x1 y x2.

- Establecer la regla de decisión del clasificador para predecir la clase de una instancia X* dada por x1* y x2*.

- En el espacio de características definido por x1 y x2, representar el conjunto de instancias de cada clase y la frontera lineal de separación entre clases del clasificador. ¿Qué significado tienen los puntos de esta frontera lineal?

Variante: LDA Balanceado

EJERCICIO 2

Teniendo en cuenta la muestra de la tabla, diseñar un Clasificador de Mínima Distancia Mahalanobis:

CLASE 1

PATRON	1	2	3	4
x_1	2	3	3	4
x_2	1	2	3	2

CLASE 2

PATRON	1	2	3
x_1	6	5	7
x_2	1	2	3

- Sin hacer las operaciones, plantee el cálculo de la matriz de covarianzas de la clase 2 y de una matriz de covarianzas estimada utilizando las observaciones de ambas clases.

- Funciones de decisión de cada clase expresadas en función de una instancia X genérica definida por x1 y x2. Se deben presentar las expresiones teóricas de forma matricial sin desarrollar, presentando los valores de las variables que intervienen.

- Función discriminante entre clases expresadas en función de una instancia X genérica definida por x1 y x2.

- Establecer la regla de decisión del clasificador para predecir la clase de una instancia X* dada por x1* y x2*.

- En el espacio de características definido por x1 y x2, representar el conjunto de instancias de cada clase y la frontera lineal de separación entre clases del clasificador. ¿Qué significado tienen los puntos de esta frontera lineal?

Variante: LDA

EJERCICIO 3

a) Diseñar un Clasificador de Mínima Distancia Mahalanobis suponiendo 3 clases de patrones, cada uno de ellos representados por 2 características, y los siguientes datos:

$$X = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \; ; \; P(C_1) = P(C_2) = P(C_3)$$

$$Vectores\ Promedio\ de\ cada\ clase\text{:}\;\; M^1 = \begin{bmatrix} 0 \\ 3 \end{bmatrix} \; ; \; M^2 = \begin{bmatrix} 5 \\ 2 \end{bmatrix} \; ; \quad M^3 = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

$$Matrices\ Covariaza\ de\ cada\ clase\text{:}\; C^1 = C^2 = C^3 = C = \begin{bmatrix} 1/2 & 0 \\ 0 & 1/4 \end{bmatrix}; \; C^{-1} = \begin{bmatrix} 2 & 0 \\ 0 & 4 \end{bmatrix}$$

El diseño del clasificador implica la obtención de las siguientes funciones de x1 y x2:

- Funciones de decisión de cada clase
- Funciones discriminantes entre las muestras de las clases dos a dos.
- Establecer la regla de decisión del clasificador a partir de las funciones discriminantes anteriores.

b) Aplicar el clasificador diseñado para predecir la clase de las instancias dadas por (2, 4), (2,0) y (6,2).

EJERCICIO 4

Para realizar un sistema diagnóstico de una determinada enfermedad, se han obtenido datos de 5 biomarcadores diferentes sobre personas sanas y personas afectadas por la enfermedad. Para ello se han realizado 2 experimentos, el primero con los dos primeros biomarcadores y el segundo con los tres restantes. Los resultados de estos experimentos se facilitan en el directorio *datos_ejercicio4*.

a) Representa para cada experimento los datos de los diferentes biomarcadores, distinguiendo en la representación aquellas muestras que se corresponden a personas sanas de aquellas tomadas a personas afectadas por la enfermedad.

b) Razone si la aplicación de un clasificador LDA es adecuada para predecir la enfermedad para cada experimento. En caso afirmativo, justifica si la aplicación de un clasificador *mínima distancia euclídea* es adecuada.

Aplicación de clasificador LDA simplificado para el caso de clases balanceadas:

Para cada experimento:

c) Plantee teóricamente las funciones cuadráticas de decisión de cada clase, expresadas en función de una instancia X genérica definida por $\begin{bmatrix} \boldsymbol{x_1} \\ \boldsymbol{x_2} \end{bmatrix}$ en el caso del experimento 1 o por $\begin{bmatrix} \boldsymbol{x_1} \\ \boldsymbol{x_2} \\ \boldsymbol{x_3} \end{bmatrix}$, en el caso del experimento 2.

d) Escriba la expresión de la función discriminante entre las dos clases del problema en función de una instancia X genérica (definida tal como expone el apartado anterior)

e) Aplique la función anterior para justificar la predicción de la clase de la instancia dada por $X^* = \begin{bmatrix} \mathbf{0.5} \\ \mathbf{-1} \end{bmatrix}$ o por $X^* = \begin{bmatrix} \mathbf{-1} \\ \mathbf{1.6} \\ \mathbf{1.5} \end{bmatrix}$, para el experimento 1 y 2, respectivamente.

f) Sólo en el caso del experimento 1, represente la frontera de decisión del clasificador y utilícela para justificar la predicción de la clase de la instancia del apartado anterior.

g) Determine la tasa de acierto en las observaciones disponibles de cada experimento.

EJERCICIO 5

Utilizando los datos X-Y facilitados en el archivo *datos_ejercicio5.mat*:

1.- Divide el conjunto completo en dos subconjuntos: entrenamiento (70% de los datos seleccionados de forma aleatoria) y test (30% de los datos restantes). Para ello, utiliza el siguiente código:

```
numDatos = size(X,1);
porcentajeTrain = 0.7;
numDatosTrain = round(porcentajeTrain*numDatos);
numerosMuestrasTrain = randsample(numDatos,numDatosTrain);
numerosMuestrasTest = find(not(ismember(1:numDatos,numerosMuestrasTrain)));

% Conjunto de Train
XTrain = X(numerosMuestrasTrain,:);
YTrain = Y(numerosMuestrasTrain);
% Conjunto de Test
XTest = X(numerosMuestrasTest,:);
YTest = Y(numerosMuestrasTest);
```

2.- Representa en dos gráficas independientes las muestras de entrenamiento y de test en el espacio de características.

3.- Utilizando el conjunto de entrenamiento, diseña un clasificador LDA y QDA, obteniendo las variables simbólicas d1(X), d2(X) y d12(X) = d1(X)-d2(X), donde X es un vector columna dato por x1, x2 y x3.

4.- Incorpora a la representación del punto 2, la frontera de separación lineal que utiliza el clasificador diseñado para particionar el espacio de características. Para ello utiliza la función de Matlab *fimplicit3(d12)*.

5.- Determine la tasa de acierto de los clasificadores en el conjunto de observaciones de test. En el caso de LDA, se debe utilizar la función d12(X), mientras que en el caso del clasificador QDA, deben utilizarse las funciones d1(X) y d2(X).

EJERCICIO 6

En el archivo *datos_ejercicio6.mat* se facilitan conjuntos de datos de entrenamiento y test: XTrain-YTrain y XTest-YTest:

1.- Utilizando el conjunto de datos de entrenamiento, diseña un clasificador LDA y un clasificador QDA obteniendo para cada caso las variables simbólicas d1(X), d2(X) y d12(X) = d1(X)-d2(X), donde X es un vector columna dato por x1 y x2. Representa su frontera de separación utilizando la función de Matlab *fimplicit(d12)*.

3.- Evalúa la tasa de acierto en el conjunto de test de ambos clasificadores, LDA y QDA.

4.- Clasificador KNN:

a) Dado un problema de clasificación *p-dimensional* de 3 clases (X = (x1, x2, …, xp)', Y = {1, 2, 3}) , describa cualitativamente los pasos metodológicos que componen la aplicación de un clasificador *KNN* para predecir la clase de una instancia X* = (x1*, x2*, …, xp*)', utilizando un conjunto de datos Xtrain-Ytrain compuesto por *n* observaciones.

   (ver Anexo V de la práctica 5, *fundamentos de programación del clasificador K-NN*)

Observación: la realización de los apartados c), y d), requiere tener implementada la función *funcion_knn* a la que hace referencia el Anexo V de la práctica 5. En la evaluación teórica, únicamente se podrá preguntar sobre el procedimiento de ajuste del hiperparámetro K y aplicación del clasificador. Este procedimiento se ilustra a nivel práctico en estos apartados b, c y d).

b) Divide el conjunto de entrenamiento XTrain-YTrain en dos subconjuntos: entrenamiento (70% de los datos seleccionados de forma aleatoria), XTrain2-YTrain2, y validación (30% de los datos restantes), XVal-YVal. Para ello, utiliza el siguiente código:

```
numDatos = size(XTrain,1);
porcentajeTrain2 = 0.7;
numDatosTrain2 = round(porcentajeTrain2*numDatos);
numerosMuestrasTrain2 = randsample(numDatos,numDatosTrain2);
numerosMuestrasVal = find(not(ismember(1:numDatos,numerosMuestrasTrain2)));

% Conjunto de Train2
XTrain2 = XTrain(numerosMuestrasTrain2,:);
YTrain2 = YTrain(numerosMuestrasTrain2);
% Conjunto de Val
XVal = XTrain(numerosMuestrasVal,:);
YVal = YTrain(numerosMuestrasVal);
```

c) Determine el valor de K como aquel que tiene la mejor tasa de acierto en el conjunto de validación cuando se entrena con el conjunto de entrenamiento reducido XTrain2-YTrain2. Probar valores impares de K desde 1 a 39.

d) Utilizando el valor de K anterior, determine la tasa de acierto del clasificador KNN en conjunto de test cuando se entrena con el conjunto de entrenamiento completo XTrain-YTrain.