

RECONOCIMIENTO DE OBJETOS

- INTRODUCCIÓN
 - Clasificación: enfoque basado en la teoría de la Decisión
- TECNICAS BÁSICAS DE CLASIFICACIÓN
 - Análisis discriminante
 - K-vecinos más cercanos
- DESARROLLO DE SISTEMAS DE RECONOCIMIENTO DE OBJETOS: ASPECTOS PRÁCTICOS
 - Análisis y pre-procesamiento de datos
 - Selección de atributos
 - Evaluación de modelos

RECONOCIMIENTO DE OBJETOS

□ INTRODUCCIÓN

- Clasificación: enfoque basado en la teoría de la Decisión

□ TECNICAS BÁSICAS DE CLASIFICACIÓN

- Análisis discriminante
- K-vecinos más cercanos

□ DESARROLLO DE SISTEMAS DE RECONOCIMIENTO DE OBJETOS: ASPECTOS PRÁCTICOS

- Análisis y pre-procesamiento de datos
- Selección de atributos
- Evaluación de modelos

RECONOCIMIENTO DE OBJETOS

- INTRODUCCIÓN
 - Clasificación: enfoque basado en la teoría de la Decisión
 - Introducción a problemas de clasificación
 - Enfoque basado en la teoría de la Decisión

RECONOCIMIENTO DE OBJETOS

- INTRODUCCIÓN
 - Clasificación: enfoque basado en la teoría de la Decisión
 - Introducción a problemas de clasificación
 - Enfoque basado en la teoría de la Decisión

PROBLEMA DE APRENDIZAJE BASADO EN DATOS: PLANTEAMIENTO GENERAL

Asumiendo un problema en el que se observa que una variable de salida Y presenta una cierta relación con un conjunto $X = (X_1, X_2, \dots, X_p)$ de p variables de entrada, de forma general, podemos decir que:

- **Un problema de aprendizaje basado en datos** se refiere al conjunto de aproximaciones o técnicas que permiten **encontrar la función f que establece la relación $X-Y$** :

$$Y = f(X) + \varepsilon$$

donde:

- f : alguna función fija de (X_1, X_2, \dots, X_p) .
 - Representa la información sistemática que X proporciona sobre Y .
 - En general, esta función f es desconocida. En esta situación, se debe estimar f basándose en las observaciones entrada-salida disponibles.
- ε : término de error aleatorio, de media cero, que es independiente de X .

- **OBJETIVO: predicción del valor de la variable de salida a partir de valores de las variables de entrada**

→ Dado un conjunto de variables entrada-salida $X-Y$, tales que $Y = f(X) + \varepsilon$, el objetivo es encontrar la función \hat{f} que permita predecir Y con un error mínimo:

donde:

$$\hat{Y} = \hat{f}(X)$$

- \hat{f} representa nuestra estimación sobre f
- \hat{Y} representa la predicción resultante de Y

PROBLEMA DE APRENDIZAJE BASADO EN DATOS: PLANTEAMIENTO GENERAL

NOTACIÓN Y TERMINOLOGÍA PARA UN PROBLEMA GENERAL:

Base de datos: compuesta por n *observaciones* (también llamadas *instancias, registros, muestras*)

➤ Variables entrada-salida del problema:

- **Variables de entrada:** $X = (X_1, X_2, \dots, X_p)$

También llamadas: *predictores, variables independientes, características, descriptores, atributos*

Número de variables de entrada: p

- **Variable de salida:** Y (también llamada *respuesta, variable dependiente*)

➤ Conjunto de datos disponibles: $\{ (x_1, y_1), (x_2, y_2), \dots, (x_n, y_n) \}$, con:

$x_i = (X_{i1}, X_{i2}, \dots, X_{ip})$ e y_i : *valores de las variable de entrada y salida para la observación i*

x_{ij} : *valor de la variable X_j para la observación i con $i = 1, \dots, n$ y $j = 1, \dots, p$*

Matricialmente:

$$X = \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} X_{11} & \cdots & X_{1p} \\ \vdots & \ddots & \vdots \\ X_{n1} & \cdots & X_{np} \end{bmatrix} \quad Y = \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix}$$

EJEMPLO: FILTRO AUTOMÁTICO DE CORREO ELECTRÓNICO ANTISPAM

- **Objetivo:** diseñar un sistema inteligente que etiquete los correos electrónicos que llegan en dos categorías, «SPAM» o «LEGÍTIMO»

PROCESO DE PREDICCIÓN:

- **1.- FUENTE DE DATOS:** hay que disponer de un conjunto de correos electrónicos etiquetados como «spam» o «legítimo» por los usuarios.
- **2.- DEFINIR Y EXTRAER ATRIBUTOS DE LOS CORREOS DISPONIBLES (PREDICTORES):** establecer y obtener características o propiedades de los datos que permitan discriminar correos de ambas categorías. Posibles predictores: dirección IP, dominio de la cuenta, contenido de ciertas palabras, tipo de archivos adjuntos, lugar de origen, tipo de caracteres o alfabeto utilizado etc...
- **3.- APRENDIZAJE DEL MODELO – FASE DE ENTRENAMIENTO:** seleccionar un tipo de modelo y “ajustarlo” en el conjunto de datos disponible para ello: conjunto de entrenamiento. Ejemplo: si se selecciona un modelo basado en reglas de decisión, hay que establecer dichas reglas → un correo es «spam» si viene de determinadas direcciones IP, y además contiene ciertas palabras, o presenta archivos adjuntos de ciertos tipos o tamaños, etc.
- **4.- EVALUACIÓN DEL MODELO – FASE DE TEST:** una vez entrenado el modelo, se evalúa en el llamado conjunto de test que, en este caso, se compone por un conjunto de correos no incluidos en el conjunto de entrenamiento. Por tanto, el conjunto de datos disponible, se dividen en dos conjuntos: entrenamiento y test.
- **5.- INTEGRACIÓN DEL MODELO – FASE DE FUNCIONAMIENTO:** si los resultados en el conjunto de test son satisfactorios, se puede plantear la integración del modelo entrenado en el sistema para enfrentarse a correos nuevos, no etiquetados por el usuario, y que se clasifiquen de forma automática.

→ ¿PROBLEMA DE CLASIFICACIÓN O REGRESIÓN ?

CLASIFICACIÓN: se predice una categoría (no el valor de una variable numérica – ejemplo: cuál va a ser el precio de un artículo, o el número de reservas que se harán en mayo en un hotel – PROBLEMA DE REGRESIÓN).

→ ¿CLASIFICACIÓN SUPERVISADA O NO SUPERVISADA ?

CLASIFICACIÓN SUPERVISADA: las observaciones incluyen una variable respuesta

CLASIFICACIÓN NO SUPERVISADA: las observaciones no incluyen una variable respuesta

EJEMPLO: PROBLEMA DE CLASIFICACIÓN ASOCIADO A LA VISIÓN ARTIFICIAL – RECONOCIMIENTO DE OBJETOS EN IMÁGENES

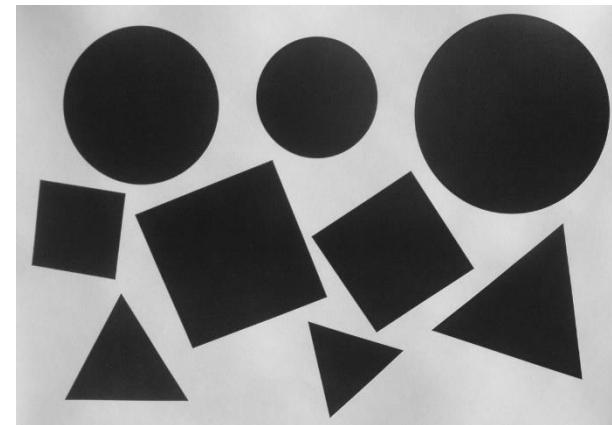
“Dado un objeto en una imagen, reconocer su forma geométrica”

Conocimiento a priori: el objeto únicamente puede presentar forma circular, cuadrada o triangular. En este ejemplo las clases de salida están definidas, pero... ¿y los predictores?

Planteamiento: cálculo de atributos o descriptores matemáticos que sean representativos de cada forma geométrica

❖ Ejemplo de descriptores matemáticos:

- Área
- Perímetro
- Relación perímetro² y área

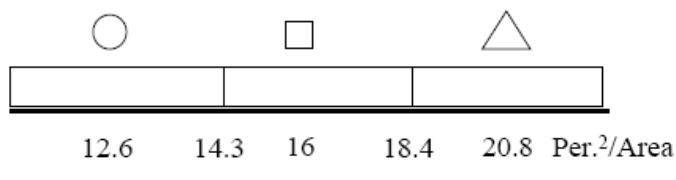


Análisis teórico de la idoneidad del descriptor

	Área	Perímetro	Perímetro ² / Área
O	πr^2	$2\pi r$	12.56
□	l^2	4l	16
Δ	$\sqrt{3}l^2/4$	3l	20.8

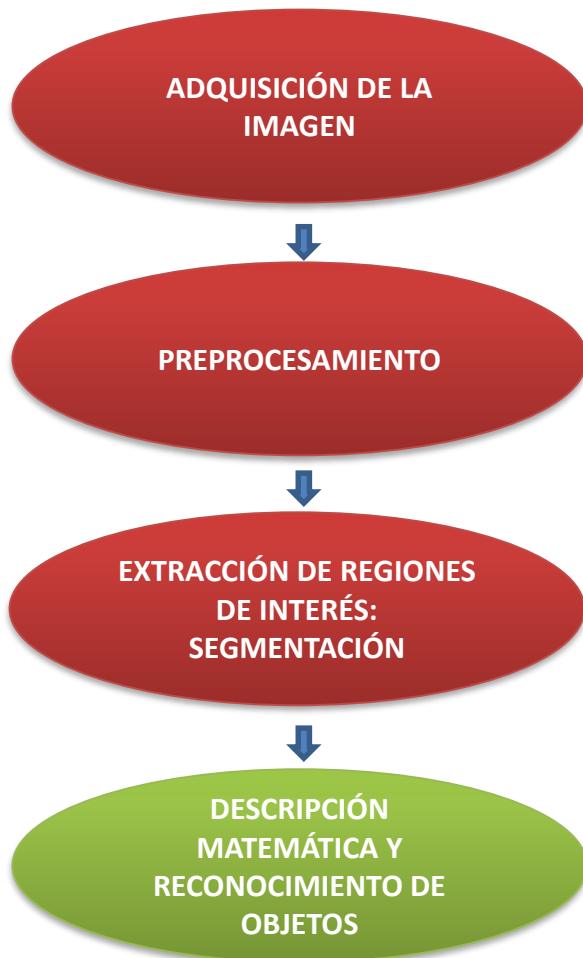
❖ Clasificador:

- Regla de decisión
 - Si $\text{per.}^2 / \text{área} < 14.3$ entonces objeto = círculo.
 - Si $14.3 < \text{per.}^2 / \text{área} < 18.4$ entonces objeto = cuadrado
 - Si $18.4 < \text{per.}^2 / \text{área}$ entonces objeto = triángulo



EJEMPLO: PROBLEMA DE CLASIFICACIÓN ASOCIADO A LA VISIÓN ARTIFICIAL

“Dado un objeto en una imagen, reconocer su forma geométrica”

FUENTE DE DATOS: IMAGEN

1. Calcular perímetro a partir del borde previamente detectado del objeto.
2. Calcular área a partir de la segmentación previa del objeto.
3. Calcular descriptor: $\text{perímetro}^2 / \text{área}$,
4. Aplicar el clasificador diseñado en base al conocimiento a priori del problema.

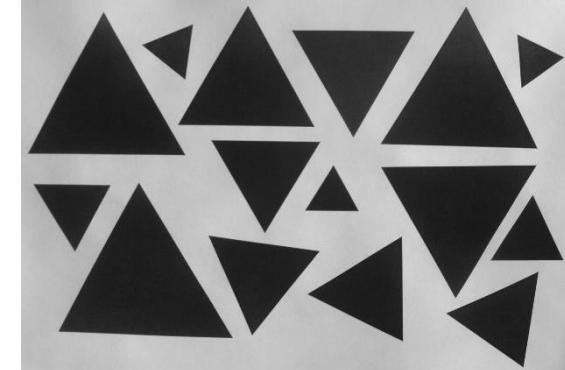
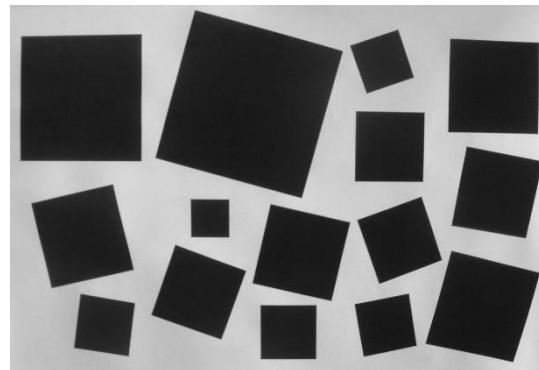
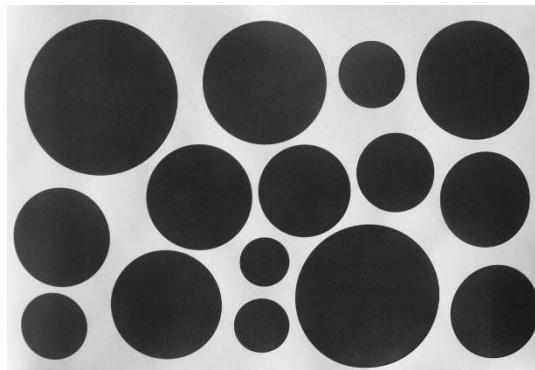
- Clasificador basado en Regla de decisión:
 - Si $\text{per.}^2 / \text{área} < 14.3$ entonces objeto = círculo.
 - Si $14.3 < \text{per.}^2 / \text{área} < 18.4$ entonces objeto = cuadrado
 - Si $18.4 < \text{per.}^2 / \text{área}$ entonces objeto = triángulo

ATRIBUTOS Y CLASIFICADOR: PLANTEAMIENTO TEÓRICO

FASE DE ENTRENAMIENTO: SELECCIÓN DE ATRIBUTOS, ELECCIÓN Y AJUSTE DEL MODELO SELECCIONADO

A PARTIR DE DATOS SIMILARES Y REPRESENTATIVOS DE LA SITUACIÓN REAL DE FUNCIONAMIENTO DEL CLASIFICADOR

EJEMPLO ANTERIOR: dado un objeto en una imagen, reconocer su forma geométrica

DISPONIBILIDAD DE IMÁGENES DE ENTRENAMIENTO**EJEMPLO DE IMÁGENES DE ENTRENAMIENTO PARA LA SELECCIÓN DE ATRIBUTOS Y DISEÑO DEL CLASIFICADOR**

- ❖ OBTENCIÓN DE DATOS: DATOS DE ATRIBUTOS O DESCRIPTORES MATEMÁTICOS «PROMETEDORES» SOBRE MUESTRAS DE FORMA GEOMÉTRICA ES CONOCIDA
- ❖ SELECCIÓN DE DESCRIPTORES ADECUADOS
- ❖ ELECCIÓN DE ESTRATEGIA DE CLASIFICACIÓN Y AJUSTE DEL MODELO

EJEMPLO: sistema que produce objetos en madera para su posterior decoración. Estos objetos deben clasificarse como pertenecientes a la clase Peras o Manzanas:

→ Clases: clase Manzanas (C_m) y de la clase Peras (C_p)

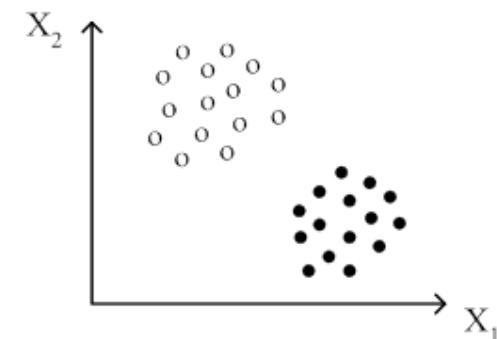


→ Atributos empleados para caracterizar cada objeto: compactidad y excentricidad.

Vector de atributos o características (describen cada instancia, cada objeto): $\vec{x} = (x_1, x_2)$ $\Rightarrow \begin{cases} x_1 \equiv \text{compactidad} \\ x_2 \equiv \text{excentricidad} \end{cases}$

→ Patrones de entrenamiento:

Se tienen disponibles para crear el modelo, 15 objetos «pera» y 15 objetos «manzana». De cada uno de ellos (de cada instancia), se obtienen sus atributos (valores de x_1 y x_2), que pueden ser representados en el espacio de características (espacio de dos dimensiones definidos por x_1 y x_2).

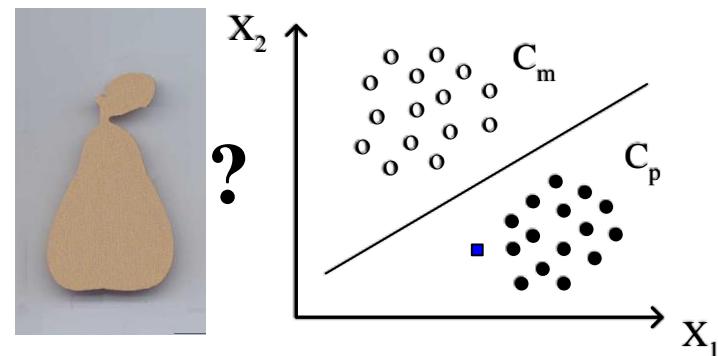


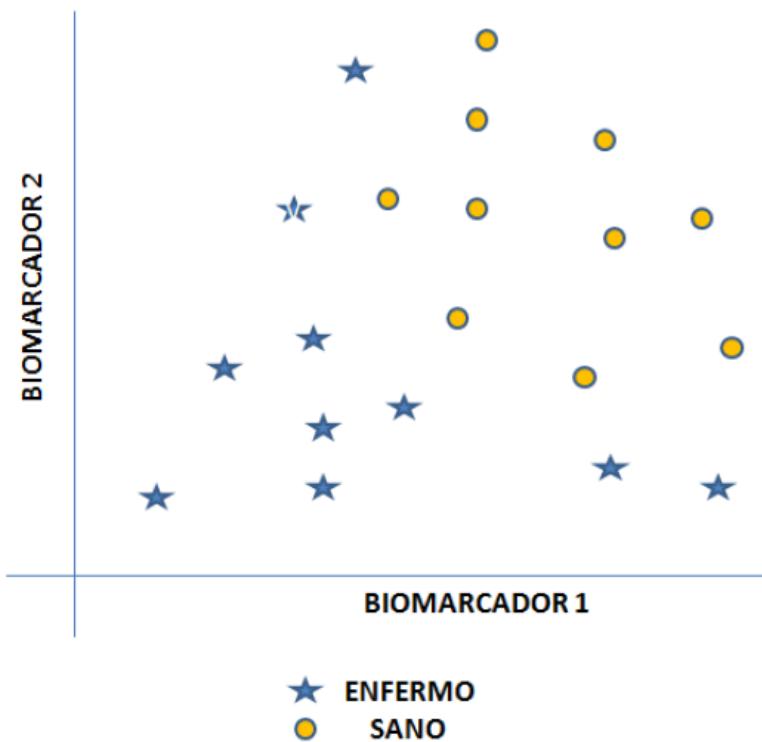
→ Planteamiento del modelo de clasificación:

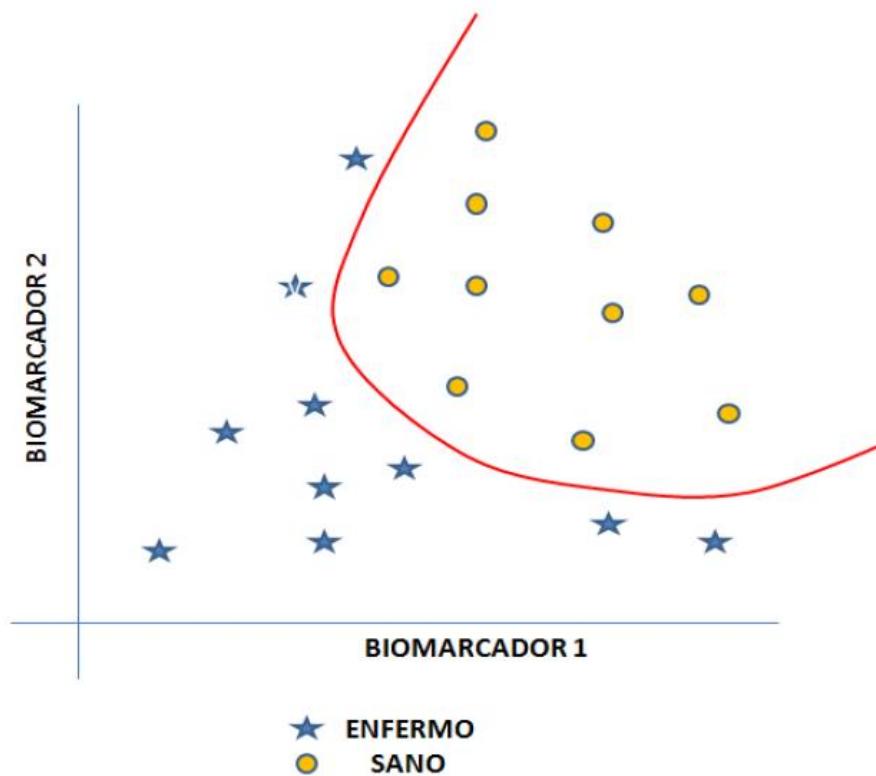
Si los atributos son representativos de cada clase, los puntos de las muestras de cada clase tienden a agruparse en regiones diferenciadas del espacio de características → Planteamiento: división del espacio de características en regiones correspondientes a las distintas clases bajo consideración.

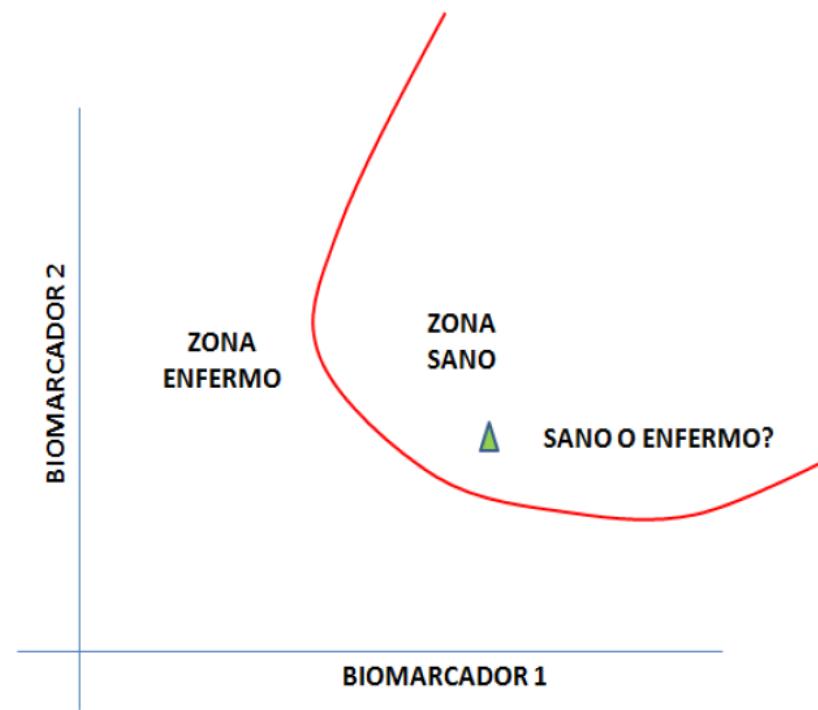
→ Fase de entrenamiento del modelo (fase de aprendizaje): diseño de una función (en el ejemplo: línea recta – clasificador lineal) que divide el espacio de características en regiones que corresponden a cada una de las clases del problema.

→ Aplicación del modelo entrenado: de un objeto de clase desconocida, se mide (x_1, x_2) y se evalúa el modelo que decidirá la clase dependiendo de la región donde se encuentre.

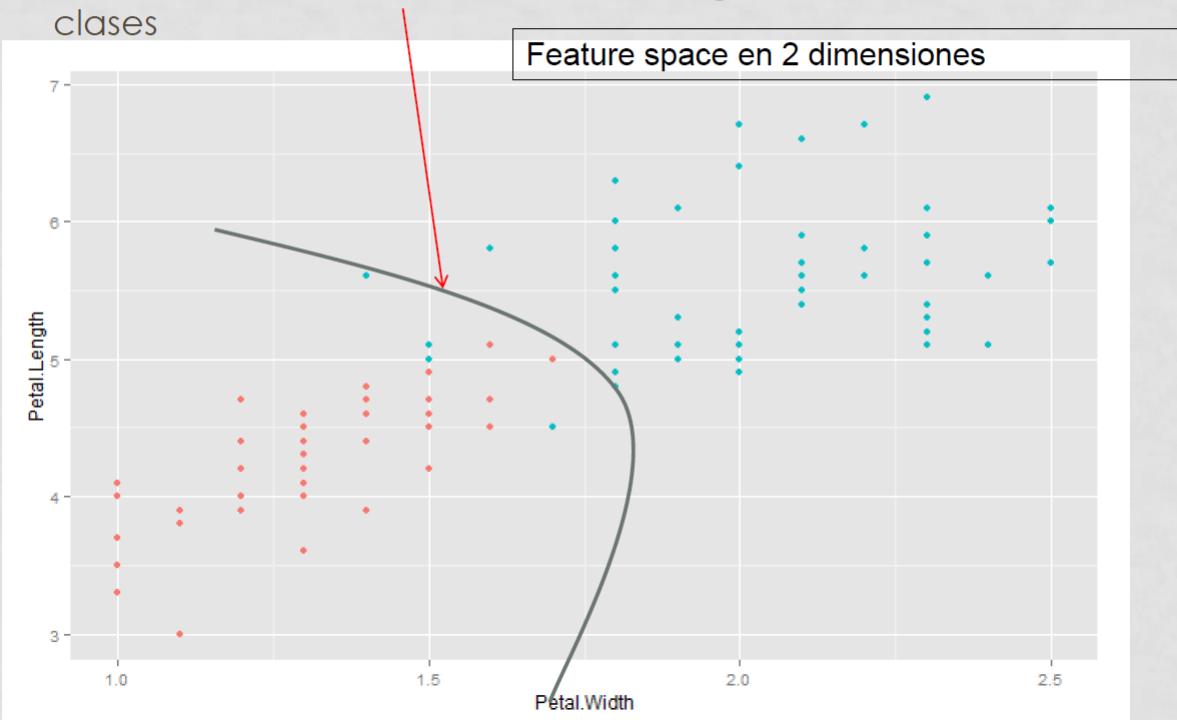


EJEMPLO**Sistema de diagnóstico**

EJEMPLO**Sistema de diagnóstico**

EJEMPLO**Sistema de diagnóstico**

- Ejemplo: clasificar plantas en dos clases ("versicolor" / roja vs. "virginica" / azul)
- 2 atributos = (Petal.Width, Petal.Length) = 2 dimensiones
- Clasificación = encontrar una función $g: X \rightarrow Y$ frontera entre las clases



PROBLEMA DE CLASIFICACIÓN: TERMINOLOGÍA Y DEFINICIONES

- **Clases:** son los posibles valores de la variable respuesta, de la salida del modelo de predicción, categorías o grupos representativos en los que se quieren clasificar los datos.
- **Instancia, ejemplo o registro (*instance, sample, record*):** cada una de las muestras disponibles para entrenar/validar/evaluar un modelo (en los ejemplos anteriores, cada uno de los correos electrónicos; cada objeto cuadrado, circular o triangular; cada figura de madera pera/manzana; cada persona sana/enferma; cada planta).
- **Característica, atributo, propiedad o campo (*feature, attribute, property, field*):** cada instancia se describe por medio de un conjunto de atributos, descriptores o características. (En los ejemplos anteriores, cada una de las medidas que se utilizan para describir un correo electrónico; relación $\text{perímetro}^2/\text{área}$, etc...)
- **Vector de características, atributos o predictores:** al conjunto de atributos que se utilizan para entrenar el modelo (predictores) y que definen una instancia se le denomina vector de predictores. Hay que tener en cuenta, que los atributos que forman parte de este vector pueden ser el resultado de un proceso de selección de características o filtrado de todos los atributos disponibles.
- **Espacio de características (*feature space*):** espacio definido por cada uno de los atributos que componen el vector de predictores. En este espacio, cada instancia se representa mediante un punto cuyas coordenadas están definidas por los valores que tienen los atributos de dicha instancia.
- **Conjunto de datos (*dataset*):** el conjunto de datos está formado por instancias; cada instancia se compone de los valores de los predictores que conforman el vector de atributos. Además, en aprendizaje supervisado, cada instancia está etiquetada con la codificación asignada a la clase a la que pertenece.
 - **Conjunto de entrenamiento (patrones de entrenamiento):** subconjunto de datos utilizados en la fase de aprendizaje para el diseño y entrenamiento del modelo (en ocasiones este conjunto se subdivide en entrenamiento + validación).
 - **Conjunto de test:** subconjunto de datos utilizados en la evaluación del modelo entrenado.

EJEMPLO: CONCESIÓN DE CRÉDITOS BANCARIOS

- Concesión de créditos bancarios
 - Un banco por Internet desea obtener reglas para predecir qué personas de las que solicitan un crédito no van a devolverlo.
 - La entidad bancaria cuenta con una gran base de datos correspondientes a los créditos concedidos (o no) a otros clientes con anterioridad.
 - Instancias (de la base de datos del banco):
 - Atributos de entrada: años del crédito, cuantía del crédito, tiene cuentas morosas, tiene casa propia
 - Clase: si/no
 - Modelo que se podría aprender:
 - SI (cuentas-morosas > 0) **ENTONCES** Devuelve-crédito = no
 - SI (cuentas-morosas = 0) **Y** ((salario > 2500) **O** (años > 10))
ENTONCES devuelve-crédito = si

EJEMPLO: CONCESIÓN DE CRÉDITOS BANCARIOS

Instancia de test

Años	Euros	Salario	Casa propia	Cuentas morosas	Crédito
10	50000	3000	Si	0	??

T = Conjunto de instancias de entrenamiento (o ejemplos, datos, patrones, ...)

Años	Euros	Salario	Casa propia	Cuentas morosas	Crédito
15	60000	2200	Si	2	No
2	30000	3500	Si	0	Si
9	9000	1700	Si	1	No
15	18000	1900	No	0	Si
10	24000	2100	No	0	No
...

Debido a esta columna, la tarea es supervisada

Algoritmo

Modelo

IF CM >0 THEN NO

IF CM =0 Y

S>2500 THEN SI

...



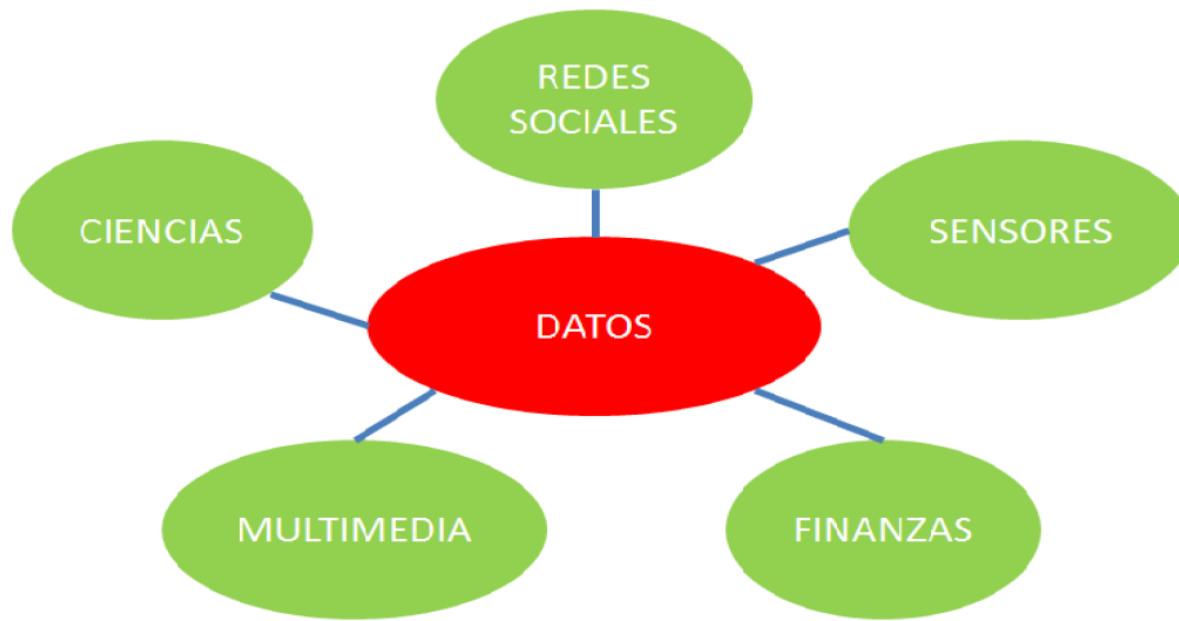
Crédito = Si

x: atributos (o características, predictores, variables independientes, variables de entrada, ...)

y: clase (o etiqueta, atributo de salida, variable dependiente, variable respuesta, ...)

CONSIDERACIONES: Fuente de datos

- Diversidad en la naturaleza de los datos



CONSIDERACIONES: Planteamiento Metodológico Común

➤ Clasificación Supervisada:

1. *Definición de la respuesta del modelo de predicción y sus posibles valores:* definen los grupos representativos o clases del problema.
2. *Creación de conjunto de datos*

2.1- EXTRACCIÓN DE ATRIBUTOS DE CADA INSTANCIA DISPONIBLE: de las muestras disponibles de cada clase del problema, se extraen o calculan propiedades de naturaleza cuantitativa/ordinal/categórica (atributos, características).

DEFINICIÓN: FEATURE SPACE (ESPACIO DE INSTANCIAS)

- Las instancias posibles “habitan” un espacio d-dimensional (donde d es el número de atributos de entrada)
 - Esta instancia tiene 5 atributos de entrada y 1 de salida

Años	Euros	Salario	Casa propia	Cuentas morosas	Crédito
10	50000	3000	Si	0	Si

- En 2 dimensiones (2 atributos), cada instancia es un punto en el feature space

CONSIDERACIONES: Planteamiento Metodológico Común

➤ Clasificación Supervisada:

1. *Definición de la respuesta del modelo de predicción y sus posibles valores:* definen los grupos representativos o clases del problema.
2. *Creación de conjunto de datos*

2.1- EXTRACCIÓN DE ATRIBUTOS DE CADA INSTANCIA DISPONIBLE: de las muestras disponibles de cada clase del problema, se extraen o calculan propiedades de naturaleza cuantitativa/ordinal/categórica (atributos, características).

- Tipos de atributos:
 - Nominales / categóricos: verde, rojo, amarillo
 - Ordinales: frío, templado, caliente
 - Reales / enteros: 1.3, 7.9, 10.798, ...
- $Y = \{C_1, C_2, \dots, C_K\}$ son las clases.
 - Si $k=2$, problema de clasificación binaria: cáncer / no-cáncer
 - Si $K>2$, problema de clasificación multi-clase: peligroso / normal / inofensivo

CONSIDERACIONES: Planteamiento Metodológico Común

➤ Clasificación Supervisada:

1. *Definición de la respuesta del modelo de predicción y sus posibles valores:* definen los grupos representativos o clases del problema.
2. *Creación de conjunto de datos*

2.1- EXTRACCIÓN DE ATRIBUTOS DE CADA INSTANCIA DISPONIBLE: de las muestras disponibles de cada clase del problema, se extraen o calculan propiedades de naturaleza cuantitativa/ordinal/categórica (atributos, características).

2.2- SELECCIÓN DE ATRIBUTOS - DEFINICIÓN DE UN VECTOR DE ATRIBUTOS: de todos los atributos, se seleccionan y extraen los predictores, esto es, aquellos atributos que describirán finalmente a las muestras involucradas en el problema de clasificación. Este conjunto de datos de clase conocida, constituye el conjunto de datos que se utilizará para diseñar, entrenar (conjunto de entrenamiento) y evaluar (conjunto de test) el modelo de clasificación.

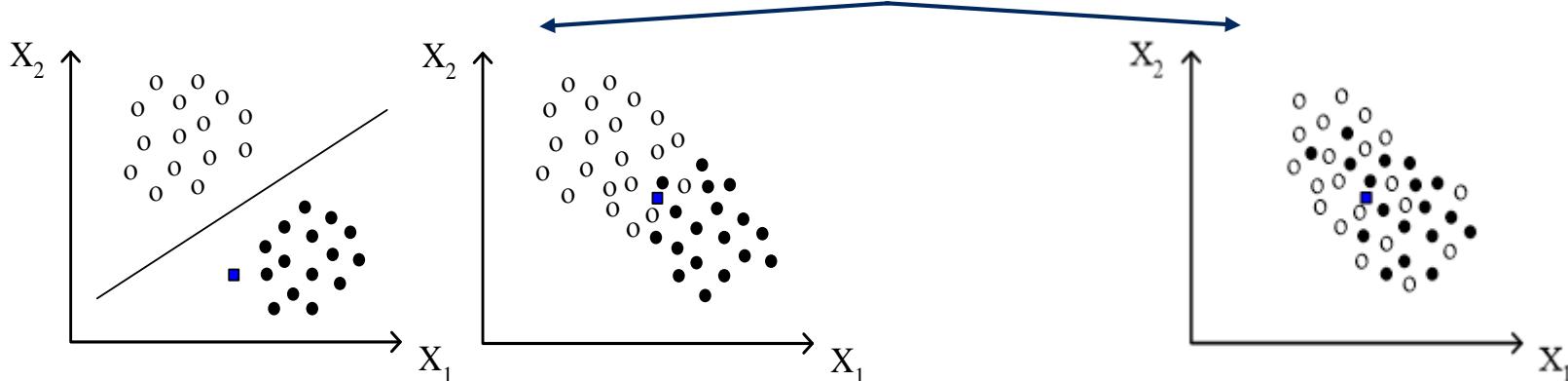
- ⇒ Deben ser, idealmente, discriminantes de las diferentes clases de interés e invariantes a todas sus posibles versiones (p. ej., en aplicaciones de reconocimiento de objetos en imágenes, los descriptores deben ser invariantes ante cambios en posición, tamaño, orientación, intensidad de color, timbre o velocidad de la voz, etc.).
- ⇒ Un conjunto de propiedades de mala calidad produce un solapamiento de clases y por tanto una gran probabilidad de error en la clasificación.

!!! IMPORTANCIA DE SELECCIÓN DE CARACTERÍSTICAS ADECUADAS !!!

* **OBJETIVO:** Proporcionar la mayor separabilidad posible entre las muestras de las clases

⇒ Ejemplo: dos atributos y dos clases.

⇒ Si tras calcular las dos características en las instancias disponibles de cada clase:



- Las características separan bien las dos clases.
- Los valores de las características de las muestras presentan cierta tolerancia o están entre cierto rango.

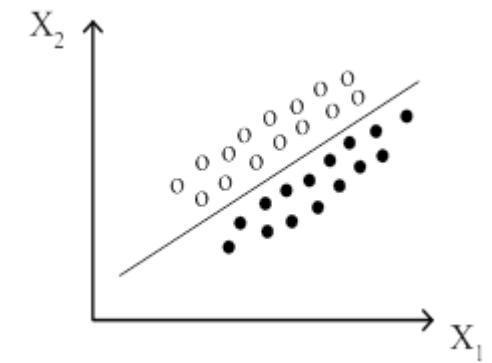
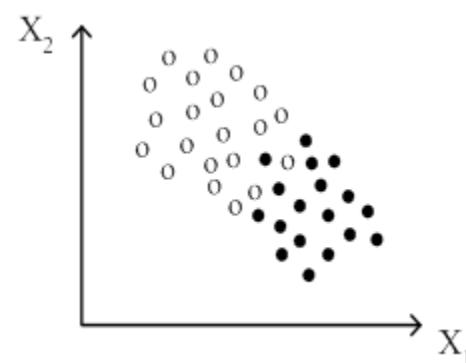
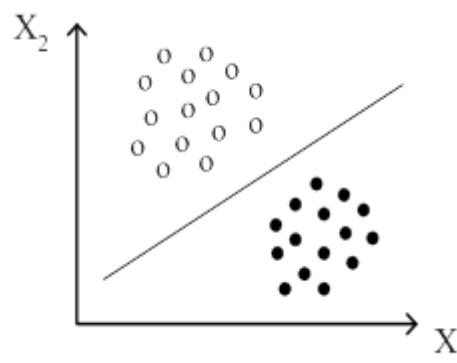
- El clasificador, por muy sofisticado que sea, no será capaz de crear una frontera de decisión que discrimine las dos clases (tan sólo podremos intentar conseguir minimizar el error en la predicción)

- Solución: buscar nuevas características.

CONCLUSIÓN: cuando el conjunto de propiedades obtenidas de las instancias es suficientemente discriminante, la complejidad del clasificador se reduce sensiblemente. En caso contrario, estrategias más complejas contribuirían, al menos, a disminuir la proporción de errores.

Requisitos de atributos

- Poder de discriminación: las características a escoger deben tomar valores distintos para muestras que pertenecen a clases diferentes.
 - Sensibles: deben reflejar diferencias para muestras «similares» de diferentes clases.
- Representatividad: las características han de tomar valores similares para muestras de la misma clase.
- Número de atributos: debe ser el más pequeño posible ya que la complejidad del modelo aumenta rápidamente con la dimensión del vector de atributos.

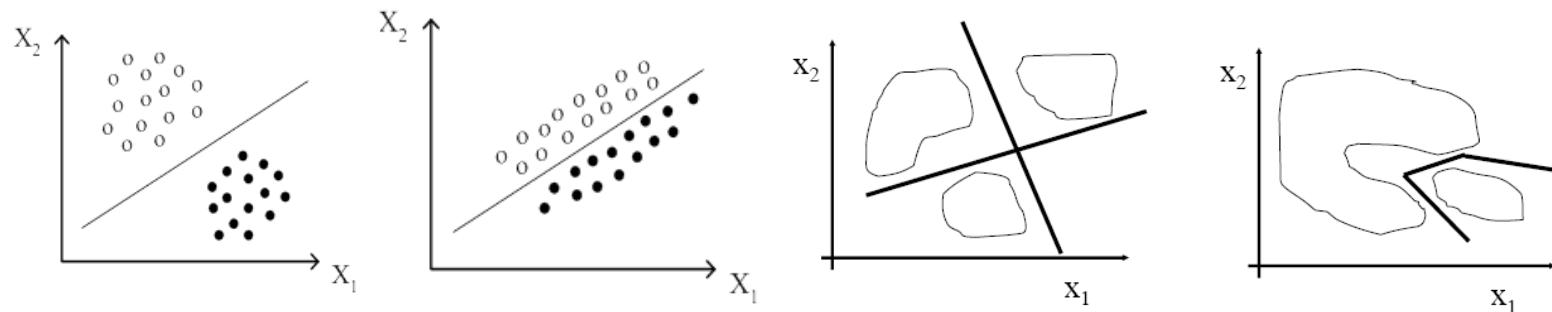


¡¡ CUIDADO: los atributos pueden no tener individualmente ningún poder de discriminación, pero sí cuando son considerados de forma conjunta !!!

CONSIDERACIONES: Planteamiento Metodológico Común

➤ Clasificación Supervisada:

1. *Definición de grupos representativos o clases*
2. *Creación de conjunto de datos*
3. *Clasificación*

3.1. Fase de entrenamiento: diseño y entrenamiento del modelo de clasificación

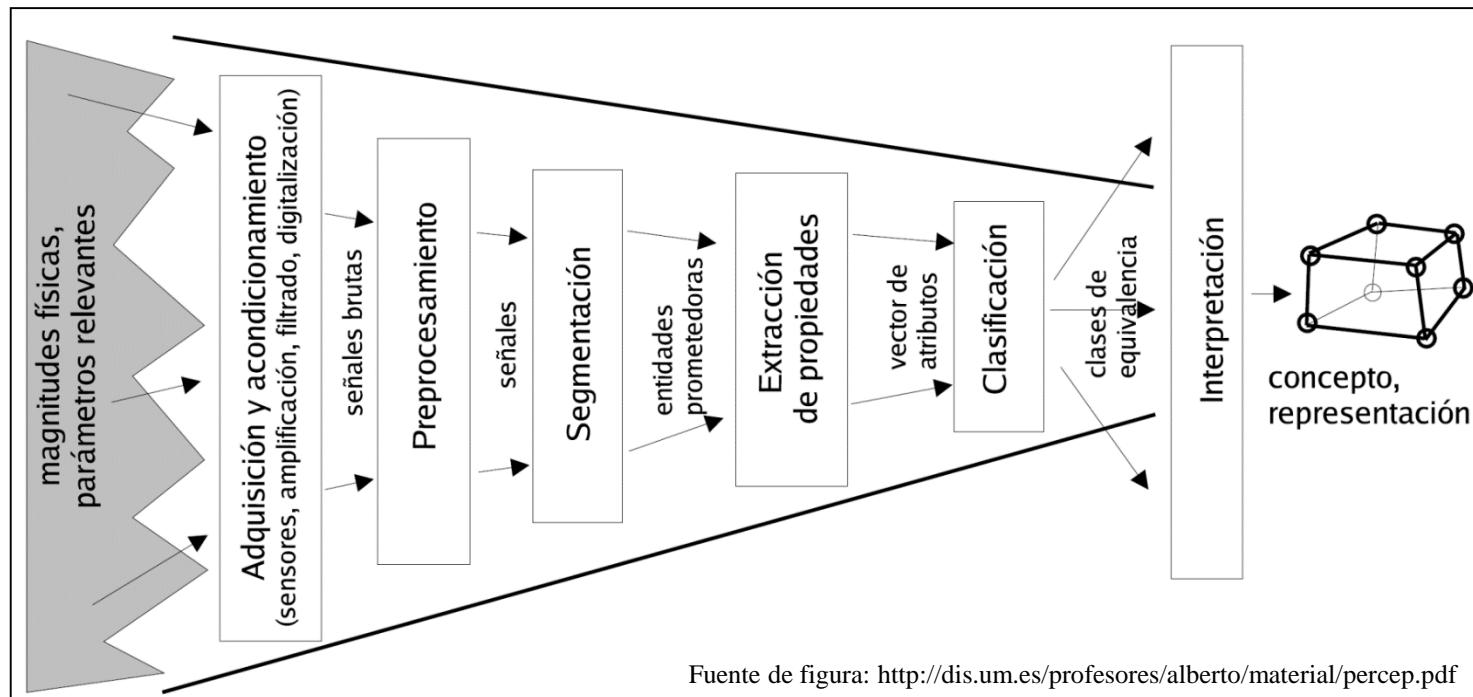
3.2. Fase de test: evaluación del modelo entrenado sobre un conjunto de datos, distinto al utilizado en el entrenamiento del modelo.

3.3. Fase de funcionamiento: aplicación del modelo entrenado para predecir la clase de una nueva muestra caracterizada por su vector de atributos.

CONSIDERACIONES: Planteamiento Metodológico Común

Técnicas de Reconocimiento de Patrones (Pattern Recognition): Técnicas encaminadas a detectar patrones o regularidades existentes en un conjunto de datos con el objetivo de clasificarlos dentro de un conjunto de categorías de interés.

- Descompone la etapa de aprendizaje y aplicación de un problema de clasificación en una serie de etapas bien definidas, independientemente de la naturaleza de la fuente de datos del proceso.



- Dependiendo de la naturaleza de los datos, para la extracción del conjunto de datos pueden ser necesarias la aplicación de nuevas etapas de adquisición y pre-procesamiento de la información, segmentación de los elementos de interés.

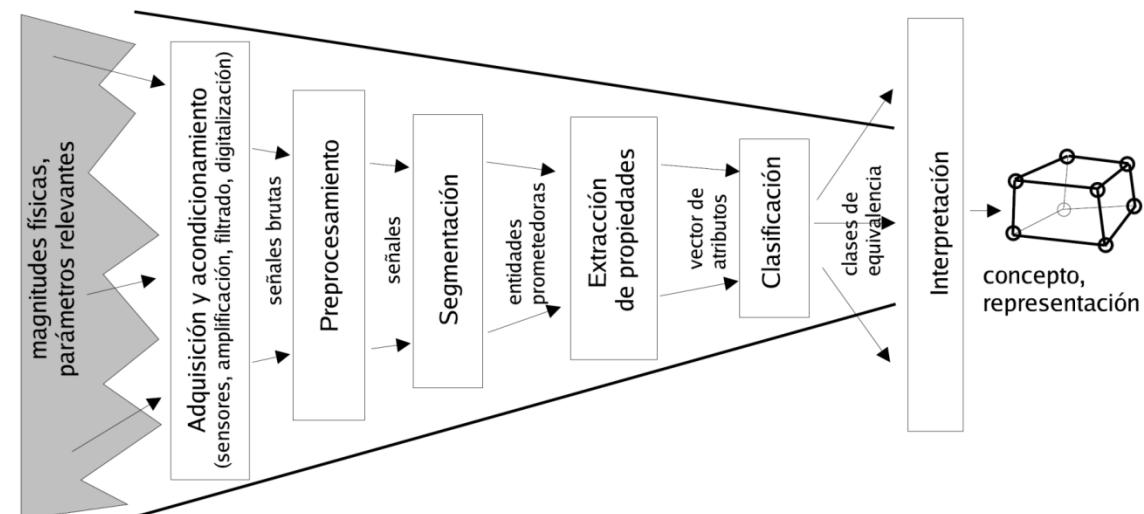
Introducción

➤ Ejemplo: reconocimiento de caracteres de matrícula a partir de imágenes

ADQUISICIÓN, PREPROCESAMIENTO, SEGMENTACIÓN

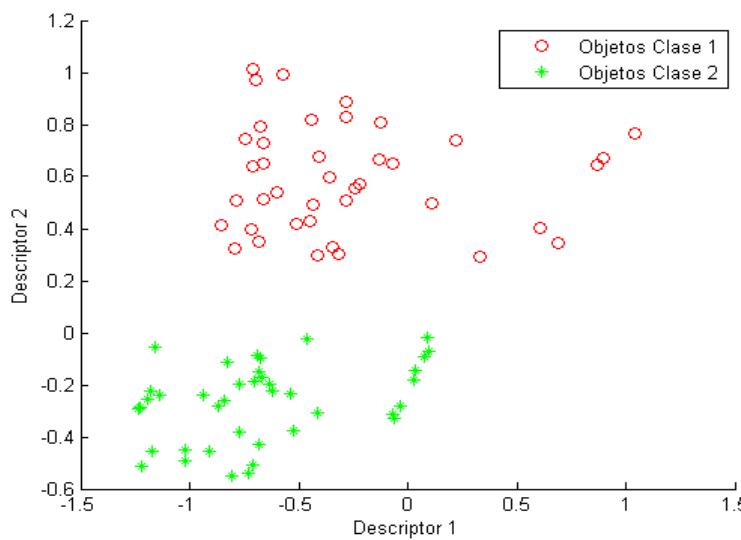
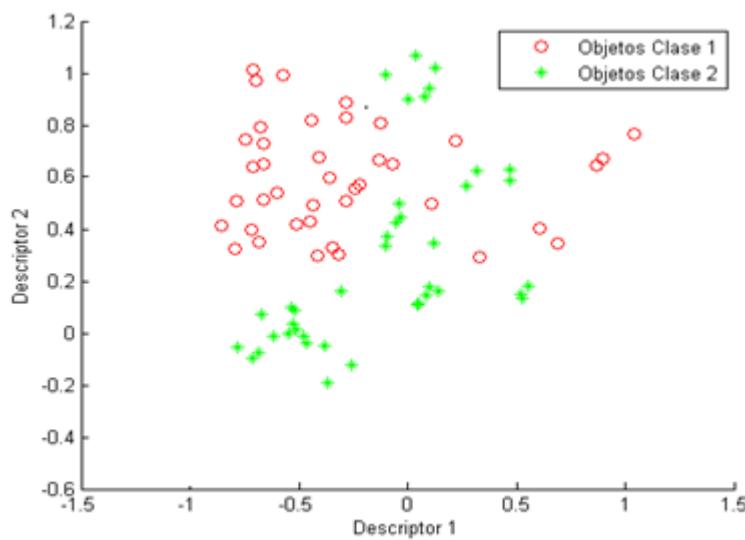
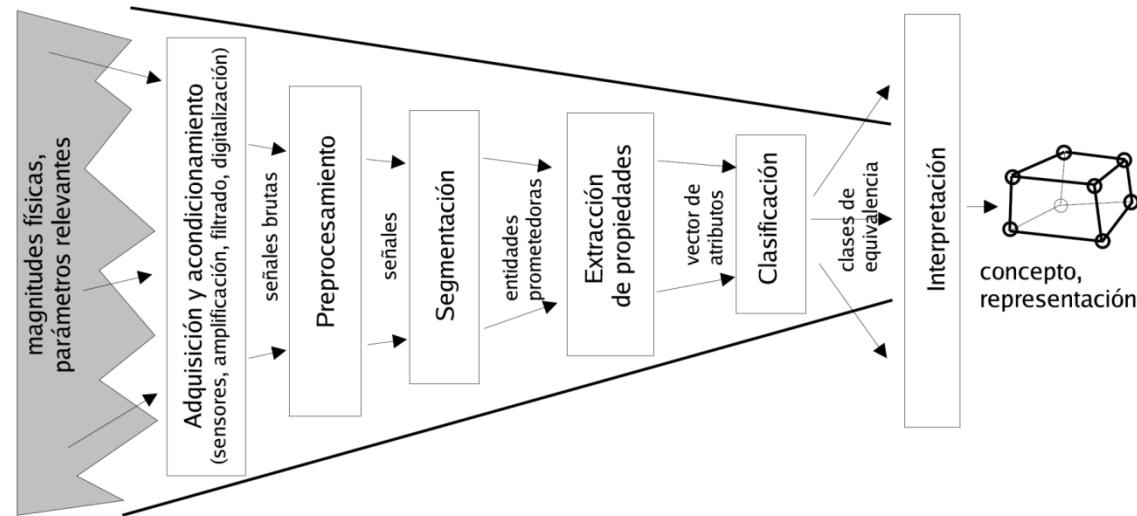
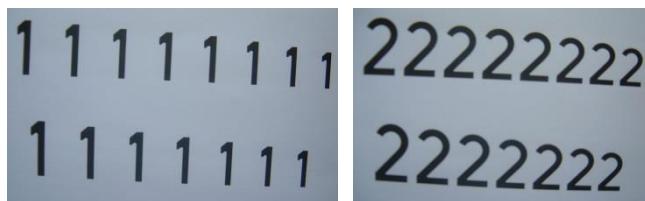


H 2305 AB



- Ejemplo: reconocimiento de caracteres de matrícula a partir de imágenes

EXTRACCIÓN DE PROPIEDADES Y CLASIFICACIÓN



- Ejemplo: reconocimiento de caracteres de matrícula a partir de imágenes

DISEÑO DEL CLASIFICADOR

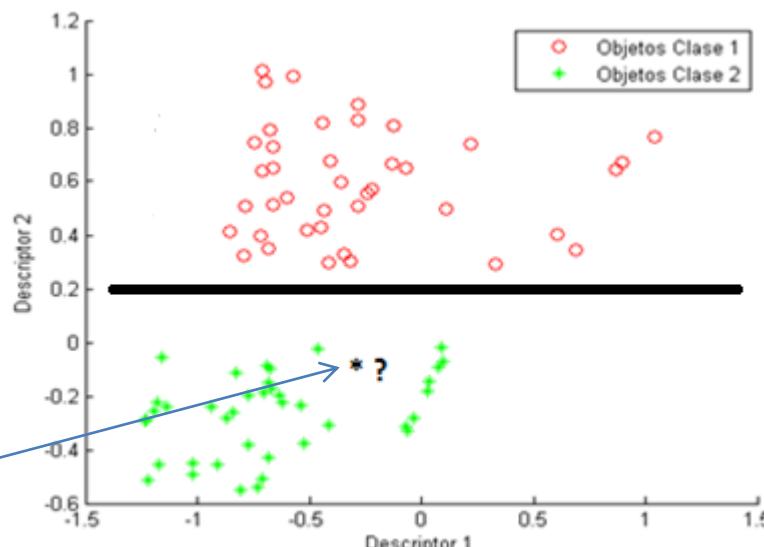
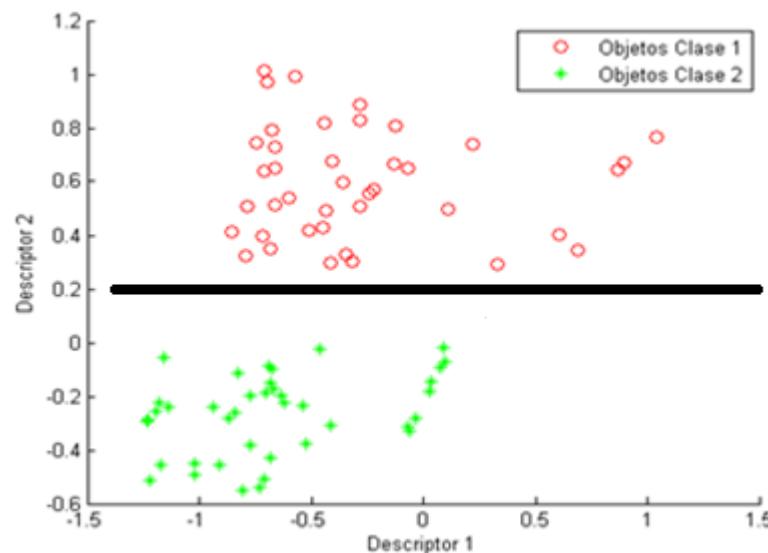


RECONOCIMIENTO: APLICACIÓN DEL CLASIFICADOR



↓
2

Medida de Descriptor 1 y 2



EN ESTE BLOQUE DE LA ASIGNATURA - TÉCNICAS DE CLASIFICACIÓN

➤ Vamos a partir de un conjunto de datos:

1. *Definición de grupos representativos o clases:* posibles tipos de objetos que, según el conocimiento a priori del problema, se espera puedan aparecer.
2. *Creación de conjunto de datos:*

 2.1- Extracción de atributos de cada instancia disponible

 2.2- Selección de atributos - definición de un vector de atributos

➤ Vamos a explicar y aplicar sobre el conjunto de datos ya definido:

Técnicas de Aprendizaje:

- *Análisis discriminante*
- *K-vecinos más cercanos*

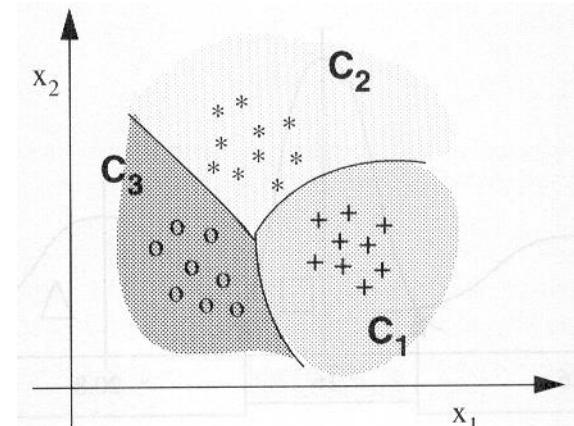
RECONOCIMIENTO DE OBJETOS

- INTRODUCCIÓN
 - Clasificación: enfoque basado en la teoría de la Decisión
 - Introducción a problemas de clasificación
 - Enfoque basado en la teoría de la Decisión

PROBLEMA DE CLASIFICACIÓN: ENFOQUE BASADO EN LA TEORÍA DE LA DECISIÓN

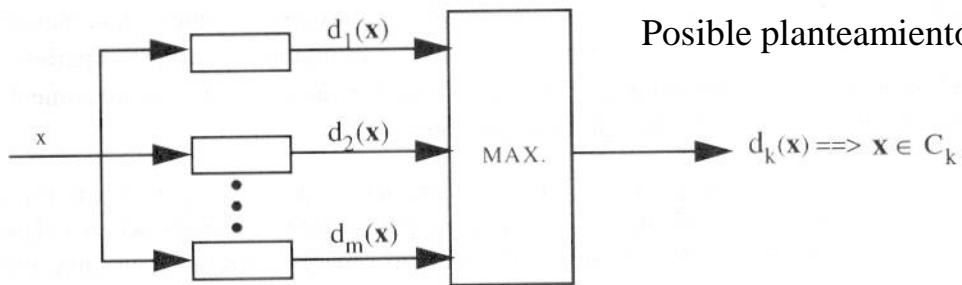
Problema de clasificación:

- ❖ Planteamiento matemático bien definido: Teoría de la Decisión, enfoque probabilístico-estadístico, enfoque basado en la optimización de funciones discriminantes:
 - División del espacio de características en regiones o subespacios representativos de cada clase considerada en el problema.
 - Implica la definición de funciones de decisión o discriminantes entre las clases del problema.



Partición del espacio de características x_1-x_2 en 3 regiones correspondientes a 3 clases

➤ La clasificación se formula en base a unas *funciones denominadas de decisión o discriminantes que son evaluadas para decidir la clase de una muestra «desconocida» descrita mediante su vector de atributos.*



Possible planteamiento:

- Se diseña una función de decisión para cada clase del problema.
- Estas funciones de decisión se evalúan para una muestra descrita por un vector de atributos x .
- La muestra se asigna a la clase C_k cuya función de decisión sea mayor.

- Ejemplo – **Clasificador de Bayes**: asigna una observación dada por x a la clase más probable. En el caso de este clasificador, la función de decisión asociada a la clase j (la variable de respuesta Y tiene el valor j), sería:

$\Pr(Y=j | X=x)$ – Probabilidad condicionada: probabilidad que una muestra sea de la clase j ($Y=j$) condicionada a que la muestra esté descrita por x ($X=x$).

□ Ejemplo para Clasificador de Bayes:

- ❖ Implica el diseño de una función de decisión para cada clase del problema de acuerdo a su probabilidad condicionada: $\Pr(Y=j | X=x)$ (probabilidad que una instancia dada por x ($X=x$) sea de la clase j ($Y=j$))
- ❖ **Ejemplo:** supongamos que tenemos 200 observaciones descritas por dos atributos X_1 y X_2 pertenecientes a dos categorías (Clasificación binaria: $Y = \{1, 2\}$, clases 1 y 2, 100 observaciones por clase)
 - **Diseño del clasificador:** con las 100 observaciones disponibles de cada clase, diseñamos una función de decisión para cada del problema:

$$d_1 = \Pr(Y=1 | X=x) ; \quad d_2 = \Pr(Y=2 | X=x)$$

Observación: este es un ejemplo de datos simulados, se conoce la función distribución de probabilidad con la que han sido generados los datos de una determinada clase (esto es, en este caso, conocemos las funciones reales d_1 y d_2 (en la práctica, se deben estimar para clasificar una observación a la clase con mayor probabilidad estimada))

➤ **Aplicación del clasificador.** Criterio de clasificación:

→ Una observación dada por $X=x_0$ se asocia a la clase 1 si $d_1 > d_2$, esto es, si

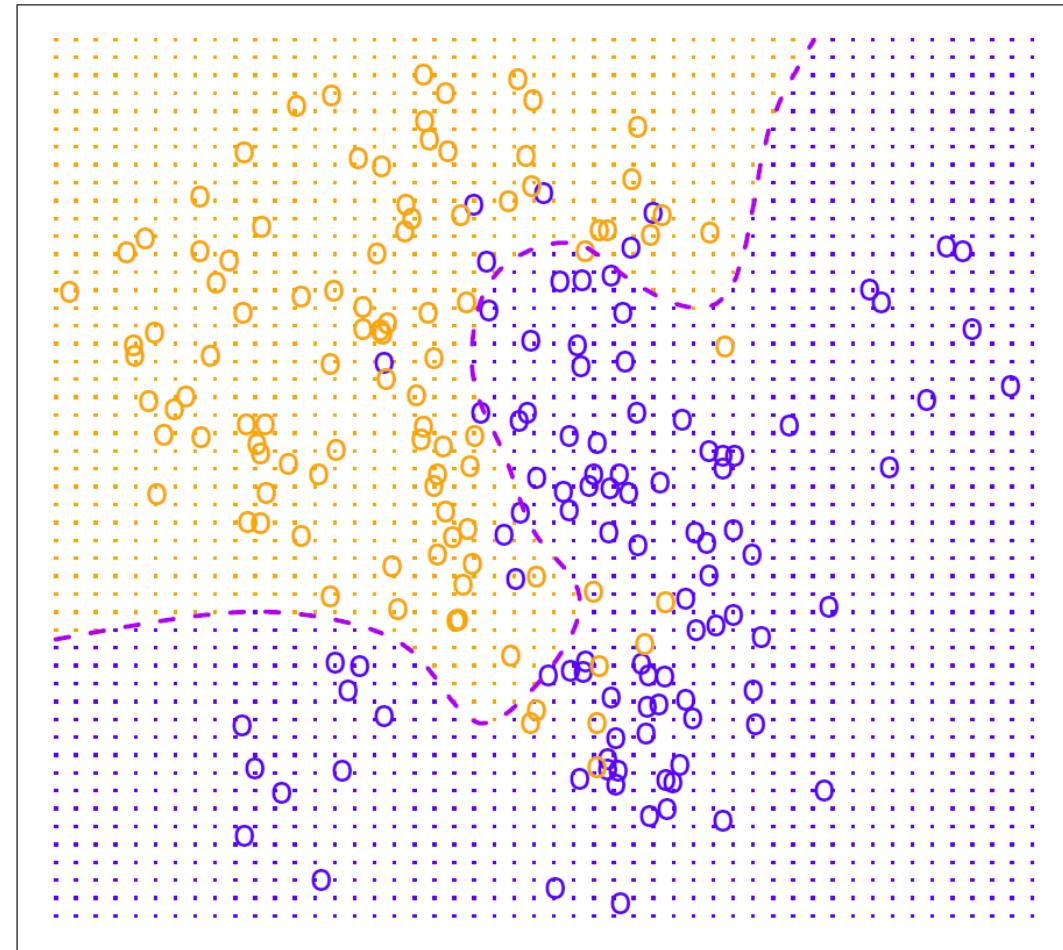
$$\Pr(Y=1 | X=x_0) > \Pr(Y=2 | X=x_0). \text{ En caso contrario, la muestra se asigna a la clase 2.}$$

- La aplicación de este criterio provoca una partición del espacio de características bidimensional, dado por X_1 y X_2 en las dos clases del problema.
- La frontera de separación entre las dos clases en este espacio de características (**Frontera de decisión de Bayes**) vendrá dada por los puntos $x = (X_1, X_2)$ para los que

$$\Pr(Y=1 | X=x) = \Pr(Y=2 | X=x_0)$$

- Ejemplo para Clasificador de Bayes:** conjunto de datos simulados de 200 observaciones de dos clases (100 de cada clase)

- La figura muestra la representación del conjunto de datos disponible (círculos). En color naranja y azul, se indican la clase a la que pertenece cada observación.
- La línea punteada púrpura representa la frontera de decisión de Bayes.
- Los puntos de fondo naranja indican la región en la que se asignará una observación de prueba a la clase naranja.
- Los puntos de fondo azul indican la región en la que se asignará una observación de prueba a la clase azul.
- En estos datos simulados, la tasa de error se sitúa en 0.1304.



- Aunque sean datos simulados y el clasificador de Bayes utilice como funciones de decisión de cada clase las funciones reales de distribución de probabilidad que generan los datos de cada clase (clasificador ideal), el error es mayor que cero porque las muestras presentan cierto solapamiento entre clases.

□ Ejemplo de estimación de probabilidades: Clasificador K-vecinos más próximos (K-NN, *K-Nearest Neighbors*)

❖ Clasificador K-NN:

- También calcula la probabilidad condicional para cada clase del problema dada una determinada observación y clasifica dicha observación a la clase con mayor probabilidad estimada.
- Dado un entero positivo K y una observación de test $X=x_0$:

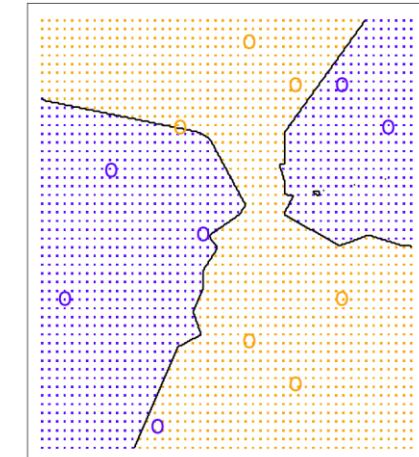
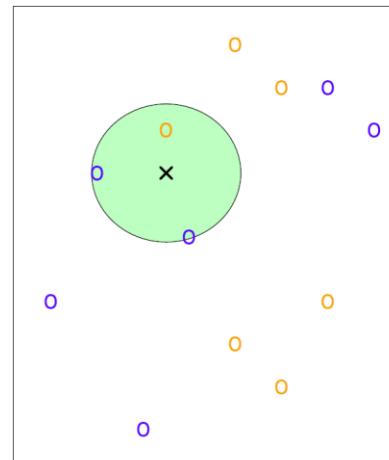
 1. El clasificador calcula N_0 : conjunto de K muestras del conjunto de entrenamiento que están más cerca de x_0 (cercanía medida en términos de distancia entre los puntos que representan las muestras en el espacio de características).
 2. El clasificador estima la probabilidad condicional de una clase como la fracción de puntos de N_0 que son de la clase en cuestión
⇒ Probabilidad que una muestra dada por x_0 sea de la clase j :

$$P(Y = j|X = x_0) = \frac{1}{K} \sum_{i \in N_0} I(y_i = j)$$

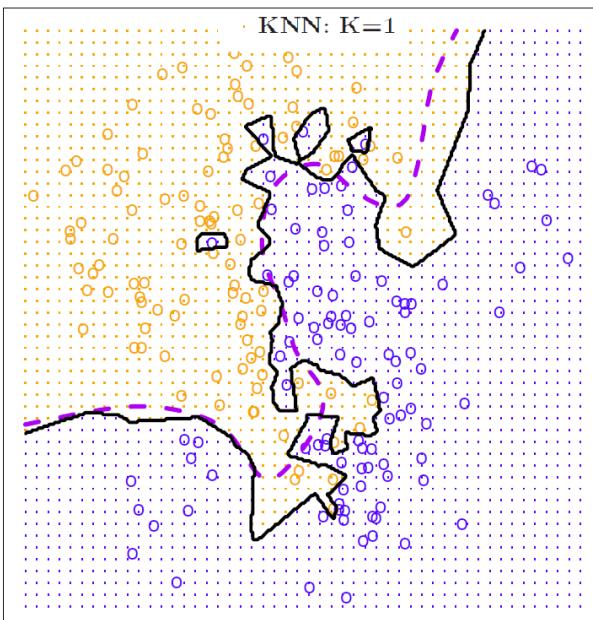
$$\text{con } I(m, n) = \begin{cases} 1 & \text{si } m = n \\ 0 & \text{si } m \neq n \end{cases}$$

Ejemplo: clasificación binaria utilizando un K-NN con $K=3$. Se dispone de un conjunto de entrenamiento formado por 6 observaciones para cada clase (círculos naranjas y azules).

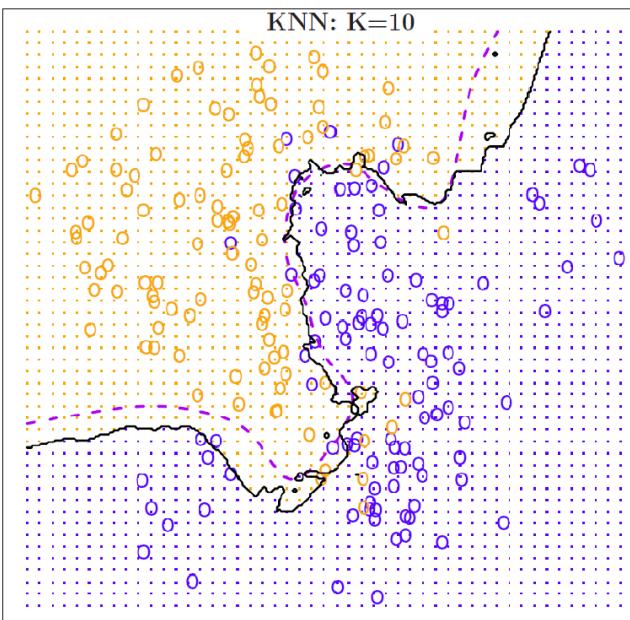
- Izquierda: clasificación de una muestra de test (cruz negra). Se identifican las tres muestras del conjunto de entrenamiento más cercanas a la de test. Se clasifica esta muestra como de la clase azul (clase más numerosa).
- Derecha: frontera de decisión KNN y partición del espacio de características según un clasificador 3-NN.



Ejemplo para Clasificador K-NN: conjunto de datos simulados de 200 observaciones de dos clases



→ Curva negra continua : frontera de decisión del KNN



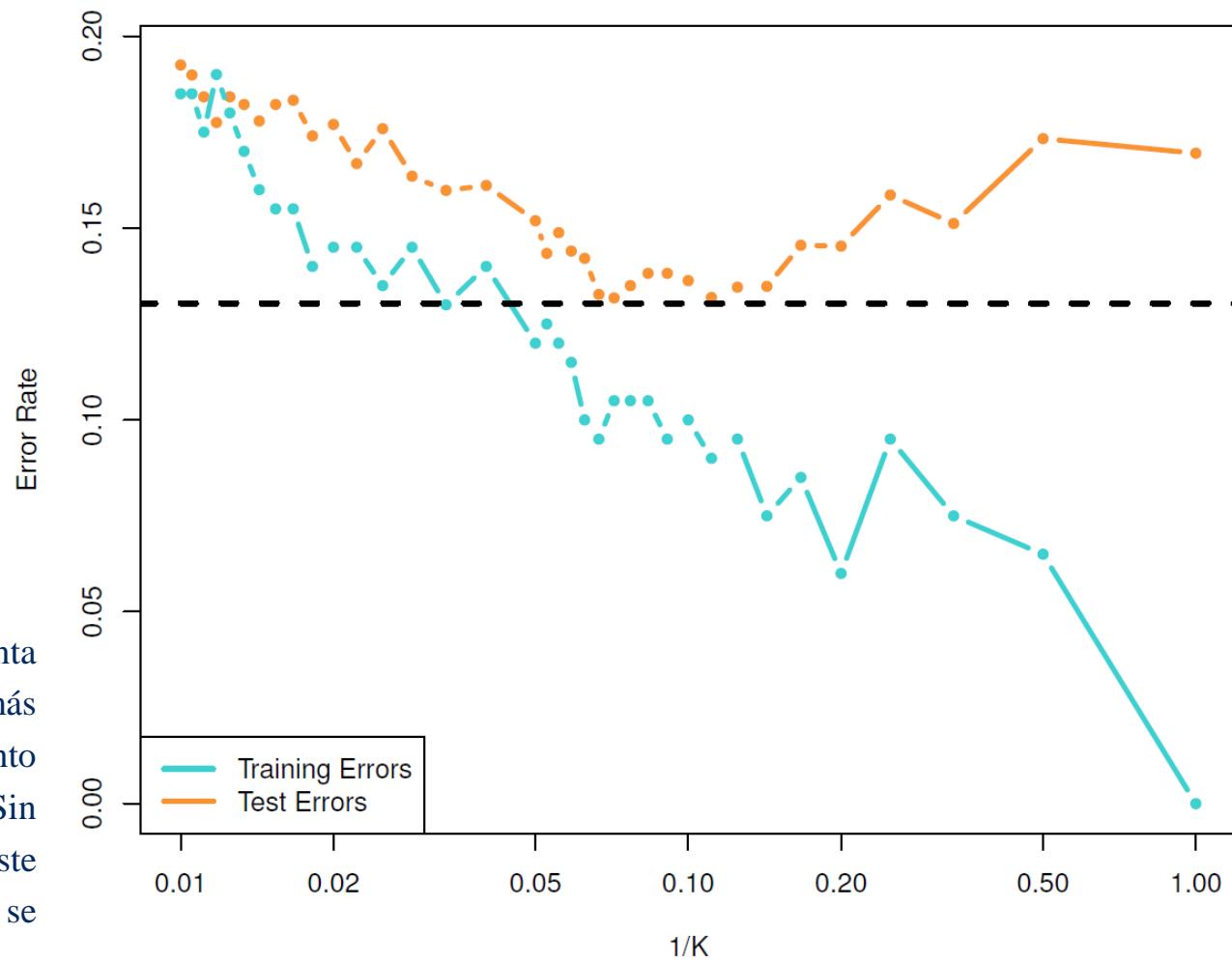
→ Curva discontinua: frontera de decisión de Bayes

- ❖ A medida que K aumenta, ¿¿¿ el método se hace más o menos flexible ???
- ❖ Analiza cómo varía el comportamiento del clasificador en función de K en términos de capacidad de ajuste vs capacidad de generalización
 - Con $K = 1$, la frontera de decisión es demasiado flexible, todo lo contrario que con $K = 100$, que genera una frontera de decisión cercana a la lineal.
 - Con $K = 10$, el clasificador de Bayes y el 10-NN generan fronteras de decisión muy parecidas.

Ejemplo para Clasificador K-NN: importancia de la elección del valor de K

Evaluación de los clasificadores anteriores sobre el conjunto de entrenamiento de 200 observaciones y sobre un conjunto de test de 5000 observaciones generadas de forma similar. La línea discontinua muestra el error del Clasificador de Bayes.

- Para $K = 100$ ($1/K = 0.01$) : altos errores en ambos conjuntos de entrenamiento y test (contorno de decisión cercano al lineal).
- A medida que disminuye K (aumenta $1/K$), el modelo es cada vez más flexible y error en el entrenamiento baja, siendo 0 para $K = 1$. Sin embargo el error de test para este valor de K es muy elevado → se produce sobreaprendizaje.
- Curva de Error en Test con forma de U característica del sobreaprendizaje. El mínimo error en test se produce para $K = 10$; para valores menores, el error en test tiende a aumentar → los modelos son tan flexibles que sobreajustan.



PLANTEAMIENTO: CLASES LINEALMENTE SEPARABLES

□ FUNCIONES DE DECISIÓN LINEALES:

SUPONGAMOS EN PRIMER LUGAR EL CASO MÁS SENCILLO:

- ❖ **ESPACIO DE CARACTERÍSTICAS BI-DIMENSIONAL (2 predictores x_1 y x_2) y PROBLEMA DE CLASIFICACIÓN BINARIA (2 clases, C_1 y C_2)**

⇒ En el espacio de características, las dos poblaciones de patrones pueden separarse mediante una recta (frontera de decisión lineal):

$$w_1x_1 + w_2x_2 + w_3 = 0 \Leftrightarrow [w_1 \ w_2 \ w_3] \begin{bmatrix} x_1 \\ x_2 \\ 1 \end{bmatrix} = 0 \Leftrightarrow W^T X = 0$$

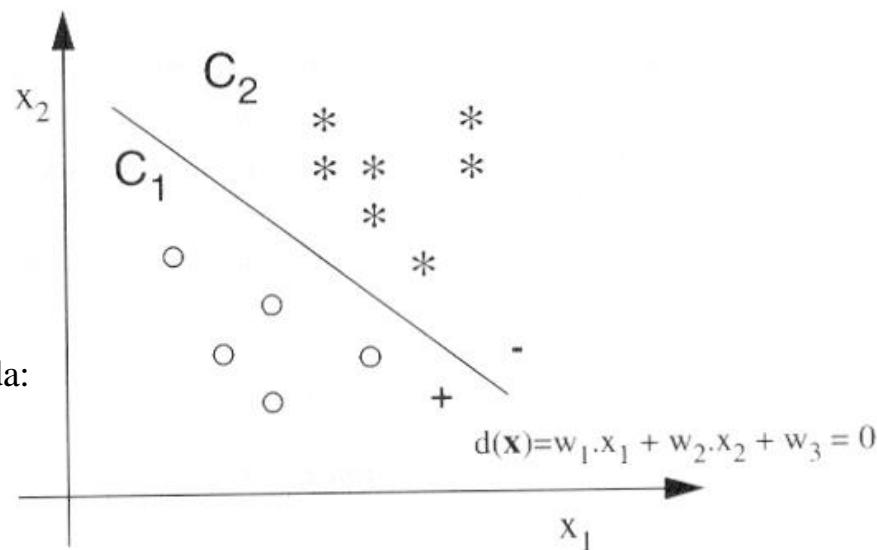
$$W = [w_1 \ w_2 \ w_3]^T \equiv \text{Vector de pesos} ; \quad X = [x_1 \ x_2 \ 1]^T \equiv \text{Vector de características}$$

⇒ La frontera de decisión lineal establece una función de decisión lineal que permite discriminar entre las dos clases del problema:

$$d(X) = w_1x_1 + w_2x_2 + w_3 = W^T X$$

Para un vector de características de clasificación desconocida:

$$d(X) \begin{cases} > 0 \Rightarrow X \in C_1 \\ < 0 \Rightarrow X \in C_2 \\ = 0 \Rightarrow X \in \text{frontera de separación} \end{cases}$$



PLANTEAMIENTO: CLASES LINEALMENTE SEPARABLES

□ FUNCIONES DE DECISIÓN LINEALES:

SUPONGAMOS EN PRIMER LUGAR EL CASO MÁS SENCILLO:

- ❖ ESPACIO DE CARACTERÍSTICAS BI-DIMENSIONAL (2 predictores x_1 y x_2) y PROBLEMA DE CLASIFICACIÓN BINARIA (2 clases, C_1 y C_2)

⇒ En el espacio de características, las dos poblaciones de patrones pueden separarse mediante una recta (frontera de decisión lineal):

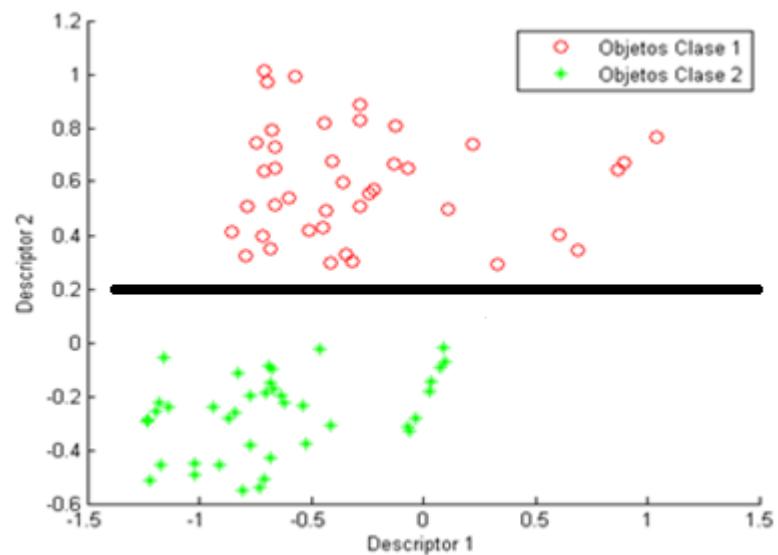
$$w_1x_1 + w_2x_2 + w_3 = 0 \Leftrightarrow [w_1 \ w_2 \ w_3] \begin{bmatrix} x_1 \\ x_2 \\ 1 \end{bmatrix} = 0 \Leftrightarrow W^T X = 0$$

$$W = [w_1 \ w_2 \ w_3]^T \equiv \text{Vector de pesos} ; \quad X = [x_1 \ x_2 \ 1]^T \equiv \text{Vector de características}$$

$$x_2 = 0,2 \rightarrow \text{Frontera de Separación (FS): } x_2 - 0,2 = 0$$

$$d(x_1, x_2) = |x_2 - 0,2|$$

$$d(x_1, x_2) = \begin{cases} > 0 & \rightarrow X \in C_1 \\ < 0 & \rightarrow X \in C_2 \\ = 0 & \rightarrow X \in FS \end{cases}$$



PLANTEAMIENTO: CLASES LINEALMENTE SEPARABLES

□ FUNCIONES DE DECISIÓN LINEALES:

SUPONGAMOS EN PRIMER LUGAR EL CASO MÁS SENCILLO:

- ❖ ESPACIO DE CARACTERÍSTICAS BI-DIMENSIONAL (2 predictores x_1 y x_2) y PROBLEMA DE CLASIFICACIÓN BINARIA (2 clases, C_1 y C_2)

⇒ En el espacio de características, las dos poblaciones de patrones pueden separarse mediante una recta (frontera de decisión lineal):

$$w_1x_1 + w_2x_2 + w_3 = 0 \Leftrightarrow [w_1 \ w_2 \ w_3] \begin{bmatrix} x_1 \\ x_2 \\ 1 \end{bmatrix} = 0 \Leftrightarrow W^T X = 0$$

$$W = [w_1 \ w_2 \ w_3]^T \equiv \text{Vector de pesos} ; \quad X = [x_1 \ x_2 \ 1]^T \equiv \text{Vector de características}$$

- ❖ SI EL ESPACIO DE CARACTERÍSTICAS ES n-DIMENSIONAL:

$$d(X) = w_1x_1 + w_2x_2 + \dots + w_nx_n + w_{n+1} = W^T X \left\{ \begin{array}{l} > 0 \Rightarrow X \in C_1 \\ < 0 \Rightarrow X \in C_2 \\ = 0 \Rightarrow X \in \text{frontera de separación} \end{array} \right.$$

$$W = [w_1 \ w_2 \ \dots \ w_n \ w_{n+1}]^T \equiv \text{Vector de pesos} ; \quad X = [x_1 \ x_2 \ \dots \ x_n \ 1]^T \equiv \text{Vector de características}$$

PLANTEAMIENTO: CLASES LINEALMENTE SEPARABLES

□ FUNCIONES DE DECISIÓN LINEALES:

SUPONGAMOS EL CASO GENERALIZADO:

- ❖ ESPACIO DE CARACTERÍSTICAS n-DIMENSIONAL (n predictores x_1, x_2, \dots, x_n) y PROBLEMA DE CLASIFICACIÓN DE K CLASES (C_1, C_2, \dots, C_K)

Distintas posibilidades de clasificación:

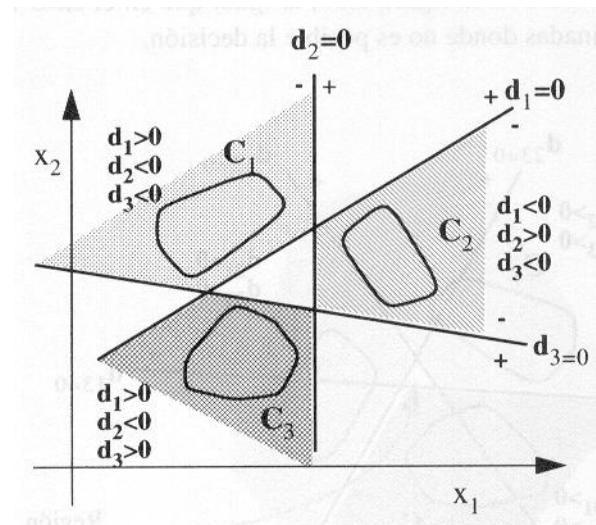
⇒ CASO 1: se consideran K funciones de decisión (una por cada clase), de forma que la función de decisión de una determinada clase discrimina las muestras de la clase en cuestión de las muestras del resto de las clases del problema (frontera de decisión de cada clase con el resto).

$$d_i(X) = w_{i1}x_1 + w_{i2}x_2 + \dots + w_{in}x_n + w_{in+1} = W_i^T X \begin{cases} > 0 \text{ si } X \in C_i \\ < 0 \text{ en otro caso} \end{cases}$$

Vector de pesos asociado a la función de decisión i-ésima:

$$W_i = [w_{i1} \ w_{i2} \ \dots \ w_{in} \ w_{in+1}]^T$$

Vector de características: $X = [x_1 \ x_2 \ \dots \ x_n \ 1]^T$



PLANTEAMIENTO: CLASES LINEALMENTE SEPARABLES

 FUNCIONES DE DECISIÓN LINEALES:

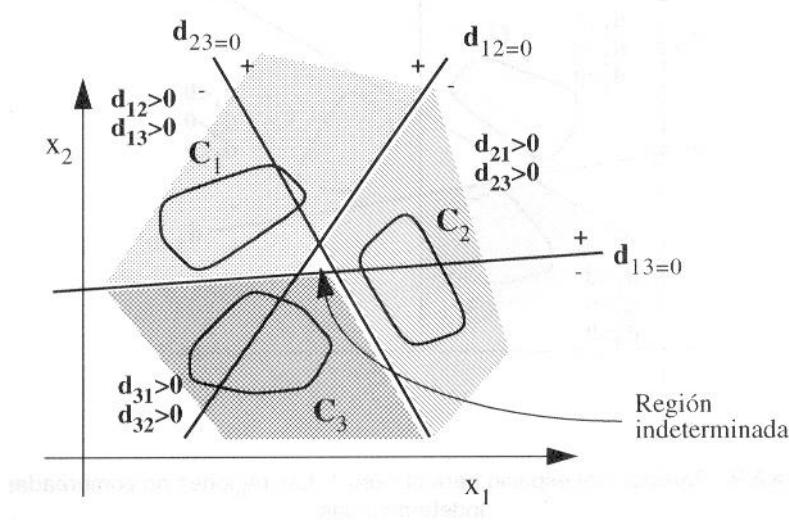
SUPONGAMOS EL CASO GENERALIZADO:

- ❖ ESPACIO DE CARACTERÍSTICAS n-DIMENSIONAL (**n** predictores x_1, x_2, \dots, x_n) y PROBLEMA DE CLASIFICACIÓN DE K CLASES (C_1, C_2, \dots, C_K)

Distintas posibilidades de clasificación:

⇒ **CASO 2:** se consideran $K(K-1)/2$ funciones de decisión lineales para particionar el espacio de características mediante fronteras de decisión lineales y separar las muestras de las clases dos a dos (combinaciones de K clases tomadas de dos en dos).

$$d_{ij}(X) = W_{ij}^T X > 0 \text{ si } X \in C_i \quad \forall j \neq i$$



PLANTEAMIENTO: CLASES LINEALMENTE SEPARABLES

□ FUNCIONES DE DECISIÓN LINEALES:

SUPONGAMOS EL CASO GENERALIZADO:

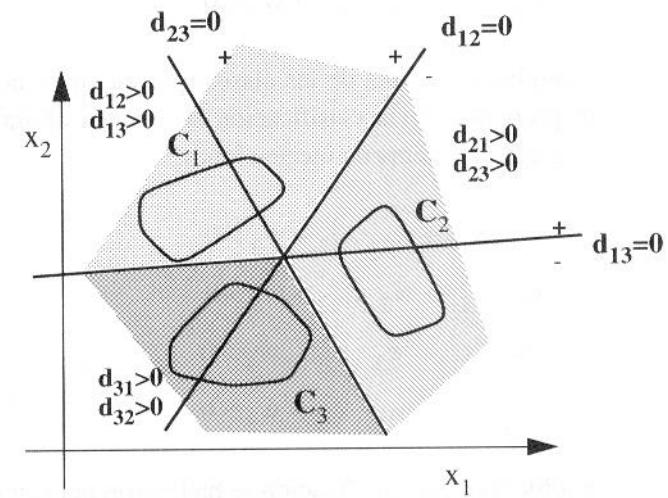
- ❖ ESPACIO DE CARACTERÍSTICAS n-DIMENSIONAL (n predictores x_1, x_2, \dots, x_n) y PROBLEMA DE CLASIFICACIÓN DE K CLASES (C_1, C_2, \dots, C_K)

Distintas posibilidades de clasificación:

⇒ CASO 3: se consideran K funciones de decisión (tantas como clases) de la forma que una observación dada por X se asocia a la clase cuya función de decisión es máxima:

$$d_k(X) = W_k^T X \text{ con } k = 1, 2, \dots, m,$$

Cumpliéndos que si $X \in C_i \Rightarrow d_i(X) > d_j(X) \quad \forall j \neq i$

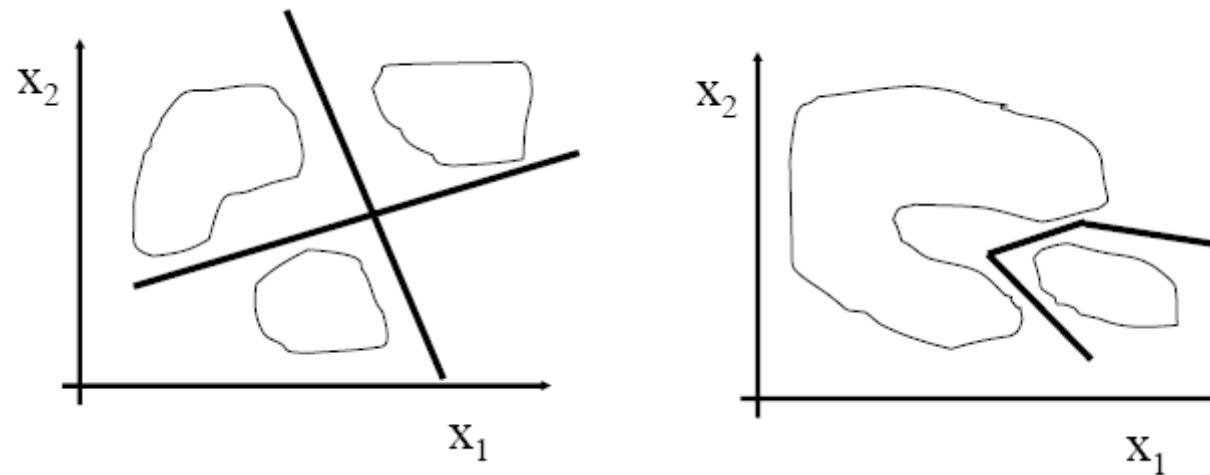


PLANTEAMIENTO: ¿ Y SI LAS CLASES NO SON LINEALMENTE SEPARABLES ?Funciones de decisión generalizadas:

➤ Las clases cuya envolvente conexa no se corta son separables:

⇒ Clases de patrones linealmente separables: si pueden separarse mediante funciones de decisión lineales.

⇒ Clases no linealmente separables: se incluyen situaciones en las que para separar las clases hay que recurrir a fronteras lineales a trozos o a fronteras no lineales.



Clases separables linealmente y clases separables no linealmente

PLANTEAMIENTO: CLASES NO LINEALMENTE SEPARABLES

Funciones de decisión generalizadas:

➤ Objetivo: generalizar el concepto de función de decisión lineal, tratando funciones de decisión en principio complejas como si fueran lineales.

$$d(X) = w_1 f_1(X) + w_2 f_2(X) + \dots + w_k f_k(X) + w_{k+1} \quad ; \quad k \geq n \quad ; \quad X = [x_1 \ x_2 \ \dots \ x_n \ 1]^T$$

$$d(X) = [w_1 \ w_2 \ \dots \ w_{k+1}] \begin{bmatrix} f_1(X) \\ f_2(X) \\ \vdots \\ f_k(X) \\ 1 \end{bmatrix} = W^T X^* \quad ; \quad W \equiv [w_1 \ w_2 \ \dots \ w_{k+1}]^T \quad ; \quad X^* = \begin{bmatrix} f_1(X) \\ f_2(X) \\ \vdots \\ f_k(X) \\ 1 \end{bmatrix}$$

⇒ Permite la representación de una gran variedad de funciones de decisión, dependiendo de la elección de las $\{f_i(X)\}$, y el número de términos usados en la expansión.

⇒ En el espacio transformado X^* (transformación del vector de atributos original X en X^* mediante las funciones $\{f_i(X)\}$), se diseñan funciones de decisión lineales:

→ Ventaja: simplificación del problema de separación de clases que requieren funciones de decisión no lineales (permite aplicar funciones lineales mediante la transformación del espacio de características).

→ Inconveniente: aumento de la dimensionalidad del problema ($k \geq n$).

Funciones de decisión generalizadas: ejemplo caso cuadrático (polinomio de grado 2) bidimensional (dos atributos)

Para el caso bidimensional: $X = [x_1 \ x_2 \ 1]^T$

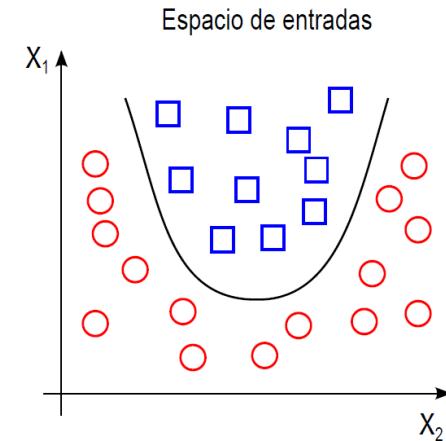
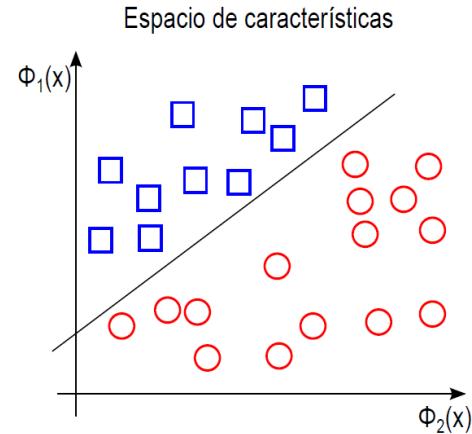
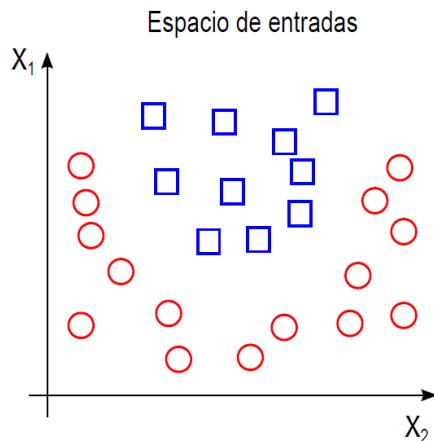


$$d(X) = w_{11}x_1^2 + w_{12}x_1x_2 + w_{22}x_2^2 + w_1x_1 + w_2x_2 + w_3$$



En forma lineal:

$$d(X) = [w_{11} \ w_{12} \ w_{22} \ w_1 \ w_2 \ w_3] \begin{bmatrix} x_1^2 \\ x_1x_2 \\ x_2^2 \\ x_1 \\ x_2 \end{bmatrix} = W^T X^* \quad ; \quad W \equiv [w_{11} \ w_{12} \ w_{22} \ w_1 \ w_2 \ w_3]^T \quad ; \quad X^* = \begin{bmatrix} x_1^2 \\ x_1x_2 \\ x_2^2 \\ x_1 \\ x_2 \end{bmatrix}$$



$$\chi = (x_1, x_2)$$

$$\phi: \chi \rightarrow F$$

$$\Phi(x) = [\Phi_1(x), \Phi_2(x)]$$

$$\phi^{-1}: F \rightarrow \chi$$

$$x = (x_1, x_2)$$

□ FUNCIONES DE DECISIÓN LINEALES:

Análisis discriminante lineal:

- Una función de decisión para cada clase del problema basada en la probabilidad condicional de Y dada X.
- Aplicación del teorema de Bayes para determinar la probabilidad condicional de Y dada X.
- Supone que los valores de X del conjunto de entrenamiento disponibles para cada clase del problema pueden modelarse por medio de una distribución normal multivariante.
- Asumiendo que las matrices de covarianza de estas distribuciones para cada clase son iguales, la formulación de este clasificador deriva en funciones de decisión lineales para discriminar entre las clases del problema.

□ FUNCIONES DE DECISIÓN GENERALIZADAS:

Fundamento de las *Máquinas de Vector Soporte (SVM)*

BIBLIOGRAFÍA PRINCIPAL

- James G., Witten D.,Hastie T. y Tibshirani R (2017). "An Introduction to Statistical Learning, with applications in R", Springer, Recurso libre: <http://faculty.marshall.usc.edu/gareth-james/ISL/index.html>

Otras referencias consultadas y fuente de figuras:

- "VISIÓN POR COMPUTADOR" (Paraninfo, 1999), González Jiménez, J.
- "Aprendizaje automático para el análisis de datos", Grado en Estadística y Empresa, Ricardo Aler, Universidad Carlos III de Madrid. OpenCourseWare: <http://ocw.uc3m.es/ingenieria-informatica/aprendizaje-automatico-para-el-analisis-de-datos>
- TUTORIAL SOBRE MÁQUINAS VECTOR SOPORTE, Enrique J. Carmona Suárez (Dpto. Inteligencia Artificial, Escuela Técnica Superior de Ingeniería Informática, Universidad Nacional de Educación a Distancia). Recurso libre: https://www.cartagena99.com/recursos/alumnos/apuntes/Tema8._Maquinas_de_Vectores_Soporte.pdf
- "Some of the figures in this presentation are taken from "An Introduction to Statistical Learning, with applications in R" (Springer, 2013) with permission from the authors: G. James, D. Witten, T. Hastie and R. Tibshirani "

RECONOCIMIENTO DE OBJETOS

- INTRODUCCIÓN
 - Clasificación: enfoque basado en la teoría de la Decisión
- TECNICAS BÁSICAS DE CLASIFICACIÓN
 - Análisis discriminante
 - K-vecinos más cercanos
- DESARROLLO DE SISTEMAS DE RECONOCIMIENTO DE OBJETOS: ASPECTOS PRÁCTICOS
 - Análisis y pre-procesamiento de datos
 - Selección de atributos
 - Evaluación de modelos

RECONOCIMIENTO DE OBJETOS

- INTRODUCCIÓN
 - Clasificación: enfoque basado en la teoría de la Decisión
- TECNICAS BÁSICAS DE CLASIFICACIÓN
 - Análisis discriminante
 - K-vecinos más cercanos
- DESARROLLO DE SISTEMAS DE RECONOCIMIENTO DE OBJETOS: ASPECTOS PRÁCTICOS
 - Análisis y pre-procesamiento de datos
 - Selección de atributos
 - Evaluación de modelos

RECONOCIMIENTO DE OBJETOS

TECNICAS BÁSICAS DE CLASIFICACIÓN

- Análisis discriminante
 - ✓ Clasificación basada en el Teorema de Bayes
 - ✓ Clasificación basada en distribución normal multivariante
 - ✓ Clasificador QDA: análisis discriminante cuadrático
 - ✓ Clasificador LDA: análisis discriminante lineal
 - ✓ Casos particulares LDA: clasificadores mínima distancia

RECONOCIMIENTO DE OBJETOS

TECNICAS BÁSICAS DE CLASIFICACIÓN

■ Análisis discriminante

- ✓ Clasificación basada en el Teorema de Bayes
- ✓ Clasificación basada en distribución normal multivariante
- ✓ Clasificador QDA: análisis discriminante cuadrático
- ✓ Clasificador LDA: análisis discriminante lineal
- ✓ Casos particulares LDA: clasificadores mínima distancia

Clasificación basada en el Teorema de Bayes:

Supongamos un problema de clasificación con K clases (en nuestro problema de predicción la variable respuesta Y puede tomar K valores cualitativos).

□ Clasificador Bayesiano:

- ⇒ Una instancia u observación dada por $X = x$, se clasifica/asocia con la clase cuya probabilidad de pertenencia sea mayor.
- ⇒ Se diseña una función de decisión por cada clase del problema basada en la probabilidad que tiene una observación descrita por $X = x$ de pertenecer a la clase en cuestión:

$$d_k = p_k(x) = \Pr(Y=k | X=x) : \text{probabilidad de pertenencia a la clase } k \text{ de una muestra descrita por } x$$

- ⇒ Una observación dada por $X = x$ se clasifica o se asocia con la clase cuya probabilidad de pertenencia sea mayor.

$$X=x \in Y = i \quad \text{si} \quad p_i(x) > p_j(x) \quad \forall \quad j \neq i$$

□ Clasificación basada en Teorema de Bayes:

- ⇒ La probabilidad $p_k(x) = \Pr(Y=k | X=x)$ se estima aplicando el Teorema de Bayes.

□ TEOREMA DE BAYES APLICADO A UN PROBLEMA DE CLASIFICACIÓN CON K CLASES DE SALIDA:

$$p_k(x) = Pr(Y=k | X=x) = \frac{Pr(X=x | Y=k) Pr(Y=k)}{\sum_{i=1}^K (Pr(X=x | Y=i) Pr(Y=i))} = \frac{f_k(x) \pi_k}{\sum_{i=1}^K (f_i(x) \pi_i)}$$

⇒ **Probabilidad a posteriori (condicional) de pertenencia de una observación descrita por x a la clase k :**

$$p_k(x) = Pr(Y = k | X = x)$$

(Probabilidad a posteriori o probabilidad condicional: es una probabilidad que se estima después de que la evidencia sea tenida en cuenta. En nuestro caso, la evidencia son los valores de los predictores que describen a la observación que queremos clasificar $X = x$).

⇒ **Probabilidad a posteriori de la observación $X = x$ en las muestras de la clase k :**

$$f_k(x) = Pr(X = x | Y = k)$$

(Término de verosimilitud: probabilidad de tener una descripción $X = x$ en las muestras disponibles de la clase k . Esta probabilidad será alta si hay una probabilidad alta de encontrar observaciones de la clase k con $X \approx x$).

⇒ **Probabilidad a priori (incondicional) de pertenencia de una observación a la clase k , independientemente de su descripción X :**

$$Pr(Y=k) = \pi_k$$

(Probabilidad de pertenencia a la clase de una observación cualquiera, probabilidad estimada sin tener en cuenta los valores de sus predictores X)

Ejemplo: moneda trucada

Se dispone de una caja cerrada con muchas monedas de dos tipos, una diseñada con acabado tipo brillo y la otra con acabado tipo mate.

Las monedas de cada tipo son idénticas y están cargadas, esto es, no existe la misma probabilidad de sacar cara o cruz en el lanzamiento de una moneda.

Se diseña un experimento en el que se saca al azar una moneda de la caja, se lanza, se anota el resultado y se vuelve a introducir en la caja para volver a repetir el proceso. Tras realizar 200 lanzamientos:

- 40 lanzamientos salieron “cara”, de los cuales 36 se realizaron con monedas de acabado brillo y 4 con monedas de acabado mate.
- 160 lanzamientos resultaron “cruz”, de los cuales 96 se realizaron con monedas de acabado brillo y 64 con monedas de acabado mate

Objetivo: utilizando los resultados anteriores, determinar la probabilidad de que al lanzar una moneda de la caja, salga cara o cruz, sabiendo el tipo de acabado de la moneda que se lanza.

⇒ **Problema de clasificación:**

- Predictores: 1 predictor, $X =$ Tipo de acabado → variable categórica con dos posibles valores: Brillo o Mate
- Salida: $Y =$ Resultado del lanzamiento → variable categórica con dos posibles valores: Cara o Cruz (problema de clasificación binaria)

Teorema de Bayes aplicado a un problema de clasificación con K clases de salida:

$$p_k(x) = Pr(Y=k | X=x) = \frac{Pr(X=x | Y=k) Pr(Y=k)}{\sum_{i=1}^K (Pr(X=x | Y=i) Pr(Y=i))} = \frac{f_k(x) \pi_k}{\sum_{i=1}^K (f_i(x) \pi_i)}$$

⇒ Si al sacar la moneda de la caja, observamos que es de tipo BRILLO. Evaluamos las probabilidades a posteriori de, sabiendo que el acabado es tipo brillo, el lanzamiento salga cara o cruz:

$$Pr(Y=cara | X=brillo) = \frac{Pr(X=brillo | Y=Cara) Pr(Y = Cara)}{Pr(X=brillo | Y=Cara) Pr(Y = Cara) + Pr(X=brillo | Y=Cruz) Pr(Y = Cruz)}$$

$$Pr(Y=cruz | X=brillo) = \frac{Pr(X=brillo | Y=Cruz) Pr(Y = Cruz)}{Pr(X=brillo | Y=Cara) Pr(Y = Cara) + Pr(X=brillo | Y=Cruz) Pr(Y = Cruz)}$$

Considerando los resultados del experimento sobre los 200 lanzamientos:

Probabilidad a posteriori de la observación «BRILLO» en los lanzamientos «CARA» y lanzamientos «CRUZ» :

$$Pr(X=brillo | Y=cara) = \frac{36}{40} = 0,9 ; Pr(X=brillo | Y=cruz) = \frac{96}{160} = 0,6$$

Probabilidad a priori de que al lanzar la moneda salga cara o cruz, sin atender al tipo de acabado:

$$Pr(Y=cara) = \frac{40}{200} = 0,2 ; Pr(Y=cruz) = \frac{160}{200} = 0,8$$

$$Pr(Y=cara | X=brillo) = \frac{0,9 * 0,2}{0,9 * 0,2 + 0,6 * 0,8} = 0,2727$$

$$Pr(Y=cruz | X=brillo) = \frac{0,6 * 0,8}{0,9 * 0,2 + 0,6 * 0,8} = 0,7273$$

Teorema de Bayes aplicado a un problema de clasificación con K clases de salida:

$$p_k(x) = Pr(Y=k | X=x) = \frac{Pr(X=x | Y=k) Pr(Y=k)}{\sum_{i=1}^K (Pr(X=x | Y=i) Pr(Y=i))} = \frac{f_k(x) \pi_k}{\sum_{i=1}^K (f_i(x) \pi_i)}$$

⇒ Si al sacar la moneda de la caja, observamos que es de tipo MATE. Evaluamos las probabilidades a posteriori de, sabiendo que el acabado es tipo mate, el lanzamiento salga cara o cruz:

$$Pr(Y=cara | X=mate) = \frac{Pr(X=mate | Y=Cara) Pr(Y = Cara)}{Pr(X=mate | Y=Cara) Pr(Y = Cara) + Pr(X=mate | Y=Cruz) Pr(Y = Cruz)}$$

$$Pr(Y=cruz | X=mate) = \frac{Pr(X=mate | Y=Cruz) Pr(Y = Cruz)}{Pr(X=mate | Y=Cara) Pr(Y = Cara) + Pr(X=mate | Y=Cruz) Pr(Y = Cruz)}$$

Considerando los resultados del experimento sobre los 200 lanzamientos:

Probabilidad a posteriori de la observación «MATE» en los lanzamientos «CARA» y lanzamientos «CRUZ» :

$$Pr(X=mate | Y=cara) = \frac{4}{40} = 0,1 ; Pr(X=mate | Y=cruz) = \frac{64}{160} = 0,4$$

Probabilidad a priori de que al lanzar la moneda salga cara o cruz, sin atender al tipo de acabado:

$$Pr(Y=cara) = \frac{40}{200} = 0,2 ; Pr(Y=cruz) = \frac{160}{200} = 0,8$$

$$Pr(Y=cara | X=mate) = \frac{0,1 * 0,2}{0,1 * 0,2 + 0,4 * 0,8} = 0,0588$$

$$Pr(Y=cruz | X=mate) = \frac{0,4 * 0,8}{0,1 * 0,2 + 0,4 * 0,8} = 0,9412$$

Teorema de Bayes aplicado a un problema de clasificación con K clases de salida:

$$p_k(x) = Pr(Y=k | X=x) = \frac{Pr(X=x | Y=k) Pr(Y = k)}{\sum_{i=1}^K (Pr(X=x | Y=i) Pr(Y = i))} = \frac{f_k(x) \pi_k}{\sum_{i=1}^K (f_i(x) \pi_i)}$$

Resumen. Procedimiento de clasificación:

1.- Diseño de clasificador basado en el Teorema de Bayes:

- *Se diseña una función de decisión para cada clase del problema (en el ejemplo anterior, salir cara o cruz), que depende de X, valores de los predictores de la observación a clasificar (en el ejemplo anterior, un solo predictor – tipo de acabado de la moneda que se lanza).*
- *Esta función de decisión representa una estimación, realizada sobre el conjunto de datos de entrenamiento y basada en el Teorema de Bayes, de la probabilidad de pertenencia de la observación a clasificar a la clase en cuestión.*

2.- Aplicación del clasificador para predecir la clase de una observación dada por X:

- *Las funciones de decisión de cada clase se evalúan para los valores de los predictores de la observación a clasificar.*
- *La observación se clasifica a la clase cuya función de decisión es mayor (la clase que tiene la mayor probabilidad de pertenencia de la observación).*

En la práctica, este esquema de clasificación basado en el Teorema de Bayes deriva en los siguientes clasificadores:

⇒ **DECISOR MÁXIMO A POSTERIORI y CLASIFICADOR SEGÚN MÁXIMA VERO SIMILITUD**

Teorema de Bayes aplicado a un problema de clasificación con K clases de salida:

$$p_k(x) = Pr(Y=k | X=x) = \frac{Pr(X=x | Y=k) Pr(Y=k)}{\sum_{i=1}^K (Pr(X=x | Y=i) Pr(Y=i))} = \frac{f_k(x) \pi_k}{\sum_{i=1}^K (f_i(x) \pi_i)}$$

CLASIFICADOR DECISOR MÁXIMO A POSTERIORI:

- Según el planteamiento anterior, se diseñan funciones de decisión para cada clase del problema que representan la probabilidad de pertenencia a la clase. Estas funciones tienen el mismo denominador.
- Por tanto, este denominador no tiene influencia en la clasificación de la observación. La clase más probable será aquella cuya función de decisión es máxima, que es aquella que el numerador más alto.

$$Y(x) = \arg \max_{k=1, \dots, K} p_k(x) = \arg \max_{k=1, \dots, K} \left(\frac{f_k(x) \pi_k}{\sum_{i=1}^K (f_i(x) \pi_i)} \right) = \arg \max_{k=1, \dots, K} (f_k(x) \pi_k)$$

Ejemplo de la moneda anterior: si lanzamos una moneda de acabado tipo BRILLO:

$$Y(brillo) = \arg \max_{cara, cruz} (f_{CARA}(brillo)\pi_{CARA}, f_{CRUZ}(brillo)\pi_{CRUZ}) = \arg \max_{cara, cruz} (0.18, 0.48) = CRUZ$$

Decisor Máximo a Posteriori (MAP):

Funciones de decisión MAP: $d_k(x) = Pr(X=x | Y=k) Pr(Y=k) = f_k(x) \pi_k$

Teorema de Bayes aplicado a un problema de clasificación con K clases de salida:

$$p_k(x) = Pr(Y=k | X=x) = \frac{Pr(X=x | Y=k) Pr(Y=k)}{\sum_{i=1}^K (Pr(X=x | Y=i) Pr(Y=i))} = \frac{f_k(x) \pi_k}{\sum_{i=1}^K (f_i(x) \pi_i)}$$

CLASIFICADOR MÁXIMA VERO SIMILITUD:

- Si por el conocimiento a priori que tenemos del problema, las clases son equiprobables (se dispone de un conjunto de datos balanceado, mismo número de observaciones de cada clase): $\pi_1 = \pi_2 = \dots = \pi_K$
- La clasificación de una observación dada por X se realiza atendiendo al término $Pr(X=x | Y=k)$ (verosimilitud).

$$Y(x) = \arg \max_{k=1, \dots, K} p_k(x) = \arg \max_{k=1, \dots, K} \left(\frac{f_k(x) \pi_k}{\sum_{i=1}^K (f_i(x) \pi_i)} \right) = \arg \max_{k=1, \dots, K} (f_k(x))$$

Ejemplo de la moneda anterior: si lanzamos una moneda de acabado tipo BRILLO:

$$Y(brillo) = \arg \max_{cara, cruz} (f_{CARA}(brillo), f_{CRUZ}(brillo)) = \arg \max_{cara, cruz} (0.9, 0.6) = CARA$$

Decisor Máxima Verosimilitud (ML, maximum likelihood):

Funciones de decisión ML: $d_k(x) = Pr(X=x | Y=k) = f_k(x)$

➤ TEOREMA DE BAYES ENFOCADO A CLASIFICACIÓN: EJEMPLO – JUGAR AL TENIS

EJERCICIO PROPUESTO: CONTESTAR DE FORMA RAZONADA A LAS SIGUIENTES PREGUNTAS UTILIZANDO UN SISTEMA DE CLASIFICACIÓN BAYESIANO DECISOR MÁXIMO A POSTERIORI.

¿Se podrá jugar al tenis con un día soleado?

¿Se podrá jugar al tenis con un día frío?

Day	outlook	temperature	humidity	windy	play
D1	sunny	hot	high	weak	no
D2	sunny	hot	high	strong	no
D3	overcast	hot	high	weak	yes
D4	rainy	mild	high	weak	yes
D5	rainy	cool	normal	weak	yes
D6	rainy	cool	normal	strong	no
D7	overcast	cool	normal	strong	yes
D8	sunny	mild	high	weak	no
D9	sunny	cool	normal	weak	yes
D10	rainy	mild	normal	weak	yes
D11	sunny	mild	normal	strong	yes
D12	overcast	mild	high	strong	yes
D13	overcast	hot	normal	weak	yes
D14	rainy	mild	high	strong	no

➤ TEOREMA DE BAYES ENFOCADO A CLASIFICACIÓN: EJEMPLO – JUGAR AL TENIS

CASO DE INTERÉS PRÁCTICO:

¿Se podrá jugar al tenis con un día soleado, frío, de humedad alta y viento fuerte ?

Day	outlook	temperature	humidity	windy	play
D1	sunny	hot	high	weak	no
D2	sunny	hot	high	strong	no
D3	overcast	hot	high	weak	yes
D4	rainy	mild	high	weak	yes
D5	rainy	cool	normal	weak	yes
D6	rainy	cool	normal	strong	no
D7	overcast	cool	normal	strong	yes
D8	sunny	mild	high	weak	no
D9	sunny	cool	normal	weak	yes
D10	rainy	mild	normal	weak	yes
D11	sunny	mild	normal	strong	yes
D12	overcast	mild	high	strong	yes
D13	overcast	hot	normal	weak	yes
D14	rainy	mild	high	strong	no

➤ TEOREMA DE BAYES ENFOCADO A CLASIFICACIÓN: EJEMPLO – JUGAR AL TENIS

- **Predictores:** $X = (X_1, X_2, X_3, X_4)$ \equiv (estado del cielo, temperatura, humedad, viento)

Variables categóricas: $X_1 = \{\text{soleado}, \text{nublado}, \text{lluvioso}\}$; $X_2 = \{\text{fría}, \text{suave}, \text{alta}\}$; $X_3 = \{\text{normal}, \text{alta}\}$; $X_4 = \{\text{flojo}, \text{fuerte}\}$

- **Salida:** $Y \equiv$ posibilidad de jugar al tenis

Variable categórica con dos posibles valores (2 clases, clasificación binaria): $Y = \{C_1, C_2\} = \{\text{sí}, \text{no}\}$

- **Conjunto de datos:** observaciones realizadas en 14 días previos, donde se ha observado el estado del cielo, temperatura, humedad y viento y se ha verificado si se podía jugar o no al tenis

- **Objetivo:**

- Saber si podremos jugar o no al tenis un día concreto, del que conocemos sus condiciones meteorológicas.

→ Clasificar una determinada observación descrita por sus predictores.

- Ejemplo: en base a los datos de la tabla, predecir si podremos jugar al tenis un día soleado, frío, de humedad alta y viento fuerte.

Day	outlook	temperature	humidity	windy	play
D1	sunny	hot	high	weak	no
D2	sunny	hot	high	strong	no
D3	overcast	hot	high	weak	yes
D4	rainy	mild	high	weak	yes
D5	rainy	cool	normal	weak	yes
D6	rainy	cool	normal	strong	no
D7	overcast	cool	normal	strong	yes
D8	sunny	mild	high	weak	no
D9	sunny	cool	normal	weak	yes
D10	rainy	mild	normal	weak	yes
D11	sunny	mild	normal	strong	yes
D12	overcast	mild	high	strong	yes
D13	overcast	hot	normal	weak	yes
D14	rainy	mild	high	strong	no

➤ TEOREMA DE BAYES ENFOCADO A CLASIFICACIÓN: EJEMPLO – JUGAR AL TENIS

¿Se podrá jugar al tenis con un día soleado, frío, de humedad alta y viento fuerte?

- ⇒ Hay que decidir qué clase ($C_1 = \text{«sí»}$ o $C_2 = \text{«no»}$) es más probable que suceda para una observación descrita por un vector de atributos de valores $\langle \text{sunny, cool, high, strong} \rangle$

Decisor Máximo a Posteriori (MAP):

Funciones de decisión MAP: $d_k(x) = Pr(X=x | Y=k) Pr(Y = k) = f_k(x) \pi_k$



$$x_0 = (X_{01}, X_{02}, X_{03}, X_{04}) = (\text{sunny}, \text{cool}, \text{high}, \text{strong})$$

$$d_1(x_0) = f_1(x_0) \pi_1 ; \quad d_2(x_0) = f_2(x_0) \pi_2$$

- ⇒ El problema implica determinar las siguientes probabilidades:

❖ **Probabilidades a priori de las clases del problema , π_k con $k = 1, 2$**

$$\pi_1 = \frac{9}{14} = 0.64 ; \quad \pi_2 = \frac{5}{14} = 0.36$$

❖ **Probabilidad de los valores de la observación x en los datos disponibles de la clase 1 «yes»: $f_1(x_0) = ???$**

❖ **Probabilidad de los valores de la observación x en los datos disponibles de la clase 2 «no»: $f_2(x_0) = ???$**

- **CLASIFICADOR NAIVE BAYES – CLASIFICADOR BAYESIANO INOCENTE (INGENUO)**
- ⇒ **ASUME INDEPENDENCIA ENTRE LOS ATRIBUTOS:** las variables que componen el vector de atributos son estadísticamente independientes:

Sea un problema de clasificación de p predictores: $X = [X_1, X_2, \dots, X_p]$

Sea una observación: $x_0 = [X_{01}, X_{02}, \dots, X_{0p}]$

La probabilidad a posteriori o verosimilitud de la observación con las muestras de una clase dada por $Y = k$:

$$f_k(x_0) = Pr(X=x_0 | Y=k) = [Pr(X_1 = X_{01} | Y=k)] * [Pr(X_2 = X_{02} | Y=k)] * \dots * [Pr(X_p = X_{0p} | Y=k)] = \prod_{i=1}^p Pr(X_i = X_{0i} | Y=k)$$

EN EL EJEMPLO ANTERIOR: $f_1(x_0) = ??? ; f_2(x_0) = ???$

$$\begin{aligned} X &= (X_1, X_2, X_3, X_4) \equiv (\text{estado del cielo, temperatura, humedad, viento}) \\ x_0 &= (X_{01}, X_{02}, X_{03}, X_{04}) = (\text{sunny, cool, high, strong}) \end{aligned}$$

Aplicando Naive Bayes

$$f_1(x_0) = Pr(X_1 = \text{sunny} | \text{yes}) * Pr(X_2 = \text{cool} | \text{yes}) * Pr(X_3 = \text{high} | \text{yes}) * Pr(X_4 = \text{strong} | \text{yes}) = \frac{2}{9} * \frac{3}{9} * \frac{3}{9} * \frac{3}{9} = 0.0082$$

$$f_2(x_0) = Pr(X_1 = \text{sunny} | \text{no}) * Pr(X_2 = \text{cool} | \text{no}) * Pr(X_3 = \text{high} | \text{no}) * Pr(X_4 = \text{strong} | \text{no}) = \frac{3}{5} * \frac{1}{5} * \frac{4}{5} * \frac{3}{5} = 0.0576$$

Funciones de decisión: $d_1(x_0) = f_1(x_0) * \pi_1 = 0,005 ; d_2(x_0) = f_2(x_0) * \pi_2 = 0,021 \rightarrow Y(x_0) = \text{NO}$

➤ CONSIDERACIONES A LA APLICACIÓN DEL CLASIFICADOR NAIVE BAYES

- ⇒ **Si el ejemplo a clasificar, no tiene el valor de algún atributo:** se omite dicho atributo
- ⇒ **Si no hay valores de algún atributo en las muestras de entrenamiento de un atributo:** esas instancias no cuentan en la estimación de probabilidades de ese atributo.
- ⇒ **Si alguna clase no presenta ningún valor posible de un determinado atributo (muestras poco representativas):**

□ **Problema:** no es posible evaluar la probabilidad de ese valor del atributo en la clase, saldría siempre cero por no haber ningún caso. De esta forma, una observación que presentara ese valor del atributo, tendría probabilidad nula de pertenecer a la clase, independientemente de los valores del resto de los atributos de la instancia

Ejemplo: si en el conjunto de datos de “Jugar al Tenis” no hubiera ninguna muestra con el valor de temperatura fría «cool» en la clase «no», siempre se podría jugar al tenis en un día «frío»

$$X = (\text{estado del cielo}, \text{temperatura}, \text{humedad}, \text{viento}) ; \quad x_0 = (\text{sunny}, \text{cool}, \text{high}, \text{strong})$$

$$f_2(x_0) = Pr(X_1 = \text{sunny} | \text{no}) * Pr(X_2 = \text{cool} | \text{no}) * Pr(X_3 = \text{high} | \text{no}) * Pr(X_4 = \text{strong} | \text{no}) = \frac{3}{5} * \frac{0}{5} * \frac{4}{5} * \frac{3}{5} = 0$$

$$d_1(x_0) = f_1(x_0) * \pi_1 = 0,005 ; \quad d_2(x_0) = f_2(x_0) * \pi_2 = 0 \rightarrow Y(x_0) = \text{SI}$$

→ **Probabilidades nulas o muy bajas, por ausencia en el conjunto de entrenamiento de algunos valores de atributos en algunas categorías.**

□ **Solución :** se supone que hay M ejemplos virtuales en cada clase por cada posible valor de cada atributo – M suele tener valores bajos (1-5) (*Suavizado de Laplace – Laplace smoothing*).

$$P(X_i = V_i | Y = k) = \frac{(\text{número de ejemplos de la clase } k \text{ con } X_i = V_i) + M}{(\text{número de ejemplos de la clase } k) + M * (\text{número de posibles valores del atributo } X_i)}$$

EN EL EJEMPLO “JUGAR AL TENIS”: SE HA APLICADO CLASIFICADOR NAIVE BAYES A UN PROBLEMA DE CLASIFICACIÓN:

- ⇒ **RESPUESTA DEL SISTEMA:** variable de salida de naturaleza cualitativa (variable nominal o categórica).
- ⇒ **ENTRADAS DEL SISTEMA O PREDICTORES:** atributos de naturaleza cualitativa (nominales o categóricos)
- LAS PROBABILIDADES IMPLICADAS EN EL DISEÑO DE LAS FUNCIONES DE DECISIÓN SE HAN ESTIMADO ATENDIENDO A LA FRECUENCIA RELATIVA DE APARICIÓN DE UN DETERMINADO CASO EN EL CONJUNTO TOTAL DE CASOS DISPONIBLES EN LOS DATOS:

→ **PROBABILIDAD A PRIORI DE LAS CLASES DEL PROBLEMA:**

$$Y = \{C_1, C_2, \dots, C_K\} ; Pr(Y = C_i) = \frac{N_i}{N} \text{ donde } N = \sum_{i=1}^K N_i ; N_i: \text{número de instancias de la clase } C_i$$

→ **PROBABILIDAD A POSTERIORI DEL VALOR DE UN ATRIBUTO EN LAS INSTANCIAS DE UNA DETERMINADA CLASE:**

$$X = [X_1, X_2, \dots, X_p] ; Pr(X_i = V_i | Y = C_k) = \frac{\text{número de ejemplos de la clase } C_k \text{ con } X_i = V_i}{\text{número de ejemplos de la clase } C_k}$$



¿ QUÉ OCURRE SI ALGÚN PREDICTOR TIENEN NATURALEZA NUMÉRICA CUANTITATIVA ?

- ❖ **SI ATRIBUTO X_i DE NATURALEZA NUMÉRICA/CUANTITATIVA:** asumir que sus valores en las muestras de cada clase pueden modelarse según una determinada función de distribución de probabilidad f .
 - ⇒ **Para cada clase:** estimar los parámetros representativos de la distribución de probabilidad elegida a partir de los valores del atributo disponibles en el conjunto de observaciones. Por ejemplo, en el caso de una distribución normal o gaussiana, estos parámetros son su valor nominal o media y la tolerancia o desviación respecto a ese valor (desviación típica).
 - ⇒ **Estimar la probabilidad a posteriori del valor del atributo en la clase en cuestión:** evaluando la función de distribución de probabilidad en el valor del atributo y considerando sus parámetros representativos calculados según el punto anterior.

$$X = [X_1, X_2, \dots, X_p] ; Y = \{C_1, C_2, \dots, C_K\}: \quad Pr(X_i = V_i | Y = C_k) \rightarrow f(X_i = V_i | Y = C_k)$$

CLASIFICADOR NAIVE BAYES GAUSSIANO:

- ❑ Asume que los valores de los atributos continuos para cada clase del problema siguen una distribución Normal o Gaussiana $N(\mu, \sigma)$:
$$f(x) = \frac{1}{\sqrt{2\pi} \sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$
- ❑ Para los datos del atributo X_i de la clase en cuestión, se calcula la media y desviación típica: μ_{ik} , σ_{ik}
- ❑ Asumir como valor representativo de la probabilidad a posteriori del valor del atributo en la clase en cuestión, el valor de la función de densidad de probabilidad para ese valor del atributo:

$$Pr(X_i = V_i | Y = C_k) \rightarrow f(X_i = V_i | Y = C_k) = \frac{1}{\sqrt{2\pi} \sigma_{ik}} e^{-\frac{(V_i - \mu_{ik})^2}{2\sigma_{ik}^2}}$$

Clasificador Naive Bayes Gaussiano: ¿es posible jugar al tenis en un día soleado, frío, viento fuerte y de humedad 89?

Ejemplo	Vista	Temperatura	Humedad	Viento	Jugar
1	Soleado	Alta (85)	Alta (85)	No	No
2	Soleado	Alta (80)	Alta (90)	Sí	No
3	Nublado	Alta (83)	Alta (86)	No	Sí
4	Lluvioso	Media (70)	Alta (96)	No	Sí
5	Lluvioso	Baja (68)	Normal (80)	No	Sí
6	Lluvioso	Baja (65)	Normal (70)	Sí	No
7	Nublado	Baja (64)	Normal (65)	Sí	Sí
8	Soleado	Media (72)	Alta (95)	No	No
9	Soleado	Baja (69)	Normal (70)	No	Sí
10	Lluvioso	Media (75)	Normal (80)	No	Sí
11	Soleado	Media (75)	Normal (70)	Sí	Sí
12	Nublado	Media (72)	Alta (90)	Sí	Sí
13	Nublado	Alta (81)	Normal (75)	No	Sí
14	Lluvioso	Media (71)	Alta (91)	Sí	No

$$X = (\text{estado del cielo}, \text{temperatura}, \text{humedad}, \text{viento})$$

$$x_0 = (\text{sunny}, \text{cool}, 89, \text{strong})$$

$$d_1(x_0) = f_1(x_0) \pi_1 ; \quad d_2(x_0) = f_2(x_0) \pi_2$$

$$\pi_1 = \frac{9}{14} = 0.64 ; \quad \pi_2 = \frac{5}{14} = 0.36$$

$$f_1(x_0) = Pr(X_1 = \text{sunny} | \text{yes}) * Pr(X_2 = \text{cool} | \text{yes}) * \\ Pr(\mathbf{x}_3 = 89 | \text{yes}) * Pr(X_4 = \text{strong} | \text{yes})$$

$$Pr(X_3 = 89 | \text{yes}) \rightarrow$$

$$\rightarrow f(X_3 = 89 | \text{yes}) = \frac{1}{\sqrt{2\pi} \sigma_{\text{hum,yes}}} e^{-\frac{(89 - \mu_{\text{hum,yes}})^2}{2\sigma_{\text{hum,yes}}^2}}$$

$$f_2(x_0) = Pr(X_1 = \text{sunny} | \text{no}) * Pr(X_2 = \text{cool} | \text{no}) * \\ Pr(\mathbf{x}_3 = 89 | \text{no}) * Pr(X_4 = \text{strong} | \text{no})$$

$$Pr(X_3 = 89 | \text{yes}) \rightarrow$$

$$\rightarrow f(X_3 = 89 | \text{no}) = \frac{1}{\sqrt{2\pi} \sigma_{\text{hum,no}}} e^{-\frac{(89 - \mu_{\text{hum,no}})^2}{2\sigma_{\text{hum,no}}^2}}$$

EJERCICIO

En las tablas adjuntas, se facilitan registros de un banco con decisiones relativos a la concesión de créditos. Estas decisiones fueron adoptadas en función del salario, edad, número de hijos y si el interesado era o no cliente del banco. Los datos de salario e hijos se facilitan en formato categórico (tabla de la izquierda) y numérico (tabla de la derecha).

SALARIO	CLIENTE	EDAD	HIJOS	CREDITO
Poco	Si	Joven	Uno	NO
Mucho	Si	Joven	Uno	SI
Mucho	Si	Joven	Uno	SI
Poco	Si	Joven	Uno	NO
Mucho	Si	Joven	Dos	SI
Poco	Si	Joven	Dos	NO
Mucho	Si	Adulto	Dos	SI
Mucho	Si	Adulto	Dos	SI
Poco	No	Adulto	Dos	NO
Mucho	Si	Adulto	Dos	SI
Medio	No	Adulto	Tres	NO
Mucho	Si	Adulto	Dos	SI
Medio	Si	Adulto	Dos	SI
Medio	No	Adulto	Tres	NO
Medio	No	Adulto	Dos	SI
Mucho	No	Mayor	Tres	NO
Poco	No	Mayor	Tres	SI
Poco	No	Mayor	Tres	SI
Mucho	No	Mayor	Tres	NO
Mucho	No	Mayor	Tres	SI

SALARIO	CLIENTE	EDAD	HIJOS	CREDITO
525	Si	Joven	1	NO
2000	Si	Joven	1	SI
2500	Si	Joven	1	SI
470	Si	Joven	1	NO
3000	Si	Joven	2	SI
510	Si	Joven	2	NO
2800	Si	Adulto	2	SI
2700	Si	Adulto	2	SI
550	No	Adulto	2	NO
2600	Si	Adulto	2	SI
1100	No	Adulto	3	NO
2300	Si	Adulto	2	SI
1200	Si	Adulto	2	SI
900	No	Adulto	3	NO
800	No	Adulto	2	SI
800	No	Mayor	3	NO
1300	No	Mayor	3	SI
1100	No	Mayor	3	SI
1000	No	Mayor	3	NO
4000	No	Mayor	3	SI

EJERCICIO

- a) Con los datos facilitados, diseña un clasificador Naive Bayes asumiendo que todos los predictores tienen naturaleza categórica y aplícalo para predecir si el banco concederá un crédito a una persona de las siguientes condiciones:
 - Persona adulta, cliente del banco, con 3 hijos y cuyo salario es bajo.
 - Persona mayor, también cliente del banco, 3 hijos y salario bajo.
 - Persona de las mismas condiciones que las anteriores, pero el banco no dispone de información relativa a su edad.
- b) Repite el apartado anterior utilizando los datos de salario y número de hijos en formato numérico, suponiendo que el salario de la persona de interés es de 700 euros.
- c) Repite el ejercicio asumiendo 5 ejemplos virtuales en cada clase por cada posible valor de cada atributo categórico.

Observación: deben plantearse teóricamente las expresiones que definen las funciones del Decisor Máximo a Posteriori y evaluarlas para cada instancia de test para obtener la clase de salida.

RECONOCIMIENTO DE OBJETOS

TECNICAS BÁSICAS DE CLASIFICACIÓN

- Análisis discriminante
 - ✓ Clasificación basada en el Teorema de Bayes
 - ✓ Clasificación basada en distribución normal multivariante
 - ✓ Clasificador QDA: análisis discriminante cuadrático
 - ✓ Clasificador LDA: análisis discriminante lineal
 - ✓ Casos particulares LDA: clasificadores mínima distancia

□ **Problema de clasificación:** recordando la notación

➤ Variables entrada-salida:

- **Variables de entrada o predictores:** $X = (X_1, X_2, \dots, X_p)$ (p predictores)
- **Variable de salida o respuesta:** $Y = \{C_1, C_2, \dots, C_K\} = \{1, 2, \dots, K\}$ (K clases)

➤ Conjunto de datos de entrenamiento (n datos): $\longrightarrow X = \begin{bmatrix} X_1 \\ \vdots \\ X_n \end{bmatrix} = \begin{bmatrix} X_{11} & \cdots & X_{1p} \\ \vdots & \ddots & \vdots \\ X_{n1} & \cdots & X_{np} \end{bmatrix}; Y = \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix}$
 $\{(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)\}$, con:

$x_i = (X_{i1}, X_{i2}, \dots, X_{ip})$ e y_i : *valores de las variable de entrada y salida para la observación i*

X_{ij} : *valor de la variable X_j para la observación i con $i = 1, \dots, n$ $j = 1, \dots, p$*

➤ Observación a clasificar:

Observación específica: $x_0 = (X_{01}, X_{02}, \dots, X_{0p})$

De forma genérica, los valores de los predictores de la observación a clasificar se denotarán por :

$X = x = (X_1, \dots, X_p)$

Teorema de Bayes aplicado para clasificar una muestra descrita por $X = x = (X_1, \dots, X_p)$

$$p_k(x) = Pr(Y=k | X=x) = \frac{Pr(X=x | Y=k) Pr(Y=k)}{\sum_{i=1}^K ((X=x | Y=i) Pr(Y=i))} = \frac{f_k(x) \pi_k}{\sum_{i=1}^K (f_i(x) \pi_i)}$$

CLASIFICADOR DECISOR MÁXIMO A POSTERIORI:

$$d_k(x) = Pr(X=x | Y=k) Pr(Y=k) = f_k(x) \pi_k$$

ESTIMACIÓN DE LA VERO SIMILITUD O PROBABILIDAD A POSTERIORI DE x EN LA CLASE k , $f_k(x)$:

→ **CLASIFICADOR NAIVE BAYES:** asume independencia estadística entre los atributos:

$$x = x_0 = (X_{01}, \dots, X_{0p})$$

$$f_k(x_0) = Pr(X=x_0 | Y=k) = [Pr(X_1 = X_{01} | Y=k)] * [Pr(X_2 = X_{02} | Y=k)] * \dots * [Pr(X_p = X_{0p} | Y=k)] = \prod_{i=1}^p Pr(X_i = X_{0i} | Y=k)$$

- **Atributos categóricos de naturaleza cualitativa:** estimar $Pr(X_i = X_{0i} | Y=k)$ como la frecuencia de los valores del atributo del conjunto de datos para la clase.
- **Atributos numéricos de naturaleza cuantitativa:** estimar $Pr(X_i = X_{0i} | Y=k)$ asumiendo que los datos de X_i en la clase k provienen de una determinada función densidad de probabilidad.

Si distribución normal o gausiana – Clasificador Naive Bayes Gaussiano:

$$\text{Para la clase } k\text{-ésima: } X_i \sim N(\mu_{ik}, \sigma_{ik}) \Rightarrow Pr(X_i = X_{0i} | Y=k) \sim \frac{1}{\sqrt{2\pi} \sigma_{ik}} \exp\left(-\frac{(X_{0i} - \mu_{ik})^2}{2\sigma_{ik}^2}\right)$$

□ **Teorema de Bayes aplicado para clasificar una muestra descrita por $X = x = (X_1, \dots, X_p)$**

$$p_k(x) = Pr(Y=k | X=x) = \frac{Pr(X=x | Y=k) Pr(Y=k)}{\sum_{i=1}^K ((X=x | Y=i) Pr(Y=i))} = \frac{f_k(x) \pi_k}{\sum_{i=1}^K (f_i(x) \pi_i)}$$

CLASIFICADOR DECISOR MÁXIMO A POSTERIORI:

$$d_k(x) = Pr(X=x | Y=k) Pr(Y=k) = f_k(x) \pi_k$$

ESTIMACIÓN DE LA VERO SIMILITUD O PROBABILIDAD A POSTERIORI DE x EN LA CLASE k , $f_k(x)$:

→ **ANÁLISIS DISCRIMINANTE: Clasificación basada en distribución normal multivariante**

- ❖ Vector de predictores continuos: $X = (X_1, X_2, \dots, X_p)$
- ❖ Asume que las observaciones provienen de una *distribución gaussiana multivariante (normal multivariante)* con vector de medias y matriz de covarianzas específicos para cada clase, μ_k y Σ_k , respectivamente.

Para la clase k -ésima: $X \sim N(\mu_k, \Sigma_k) \rightarrow f_k(x) = \frac{1}{(2\pi)^{p/2} |\Sigma_k|^{1/2}} \exp \left[-\frac{1}{2} (x - \mu_k)^T (\Sigma_k)^{-1} (x - \mu_k) \right]$

$x = [X_1, X_2, \dots, X_p]^T$; $\mu_k = [\mu_1, \mu_2, \dots, \mu_p]^T_k$ (vectores columna $px1$) ; Σ_k : matriz de covarianzas (pxp)

→ **Observación:** para aplicar análisis discriminante en un problema que incluye predictores categóricos habría que transformar previamente sus valores a numéricos (por ejemplo: mediante la creación de variables dummy, ficticias, tantas como posibles valores tenga la variable categórica, que adoptan valores 0 o 1 según el valor del predictor categórico en la instancia bajo consideración).

DISTRIBUCIÓN NORMAL MULTIVARIANTE

PARÉNTESIS – NOCIONES MATEMÁTICAS: MATRIZ DE COVARIANZAS

- Sea $X = [X_1, X_2, \dots, X_p]^T$ un vector aleatorio p -dimensional ($p \times 1$). Si disponen de n muestras de X , su vector de medias y matriz de covarianzas se definen de la siguiente forma:

$$\rightarrow \text{Vector de medias: } \mu = [\mu_1, \mu_2, \dots, \mu_p]^T, \text{ donde } \mu_i = E[X_i] = \frac{1}{n} \sum_{z=1}^n X_{zi}$$

$$\rightarrow \text{Matriz de covarianzas: } \Sigma = E[(X - \mu)(X - \mu)^T] = \begin{bmatrix} \sigma_{11} & \sigma_{12} & \cdots & \sigma_{1p} \\ \sigma_{21} & \sigma_{22} & \cdots & \sigma_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{p1} & \sigma_{p2} & \cdots & \sigma_{pp} \end{bmatrix}$$

donde:

$$\sigma_{ij} = Cov(X_i, X_j) = E[(X_i - \mu_i)(X_j - \mu_j)] = \frac{1}{n-1} \sum_{z=1}^n (X_{zi} - \mu_i)(X_{zj} - \mu_j)$$

Notar que Σ es simétrica ($\sigma_{ij} = \sigma_{ji}$) y que $\sigma_{ii} = Var(X_i) = \sigma_i^2 = E[(X_i - \mu_i)^2] = \frac{1}{n-1} \sum_{z=1}^n (X_{zi} - \mu_i)^2$

DISTRIBUCIÓN NORMAL MULTIVARIANTE

PARÉNTESIS – NOCIONES MATEMÁTICAS: MATRIZ DE COVARIANZAS

EJEMPLOS SIGNIFICADO MATRIZ DE COVARIANZAS PARA DISTINTOS CONJUNTO DE n DATOS DESCRITOS POR $X = (X_1, X_2)$

$$\begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} X_{11} & X_{12} \\ \vdots & \vdots \\ X_{n1} & X_{n2} \end{bmatrix}$$

□ **EJEMPLO 1:** variables correlacionadas

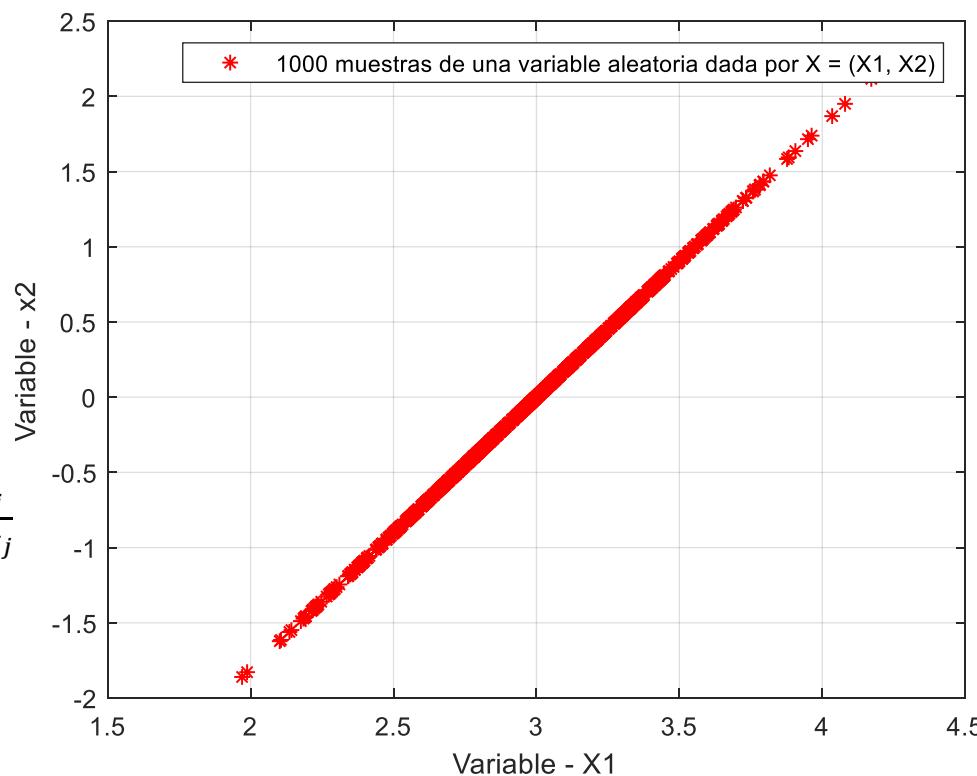
$$\Sigma = \begin{bmatrix} \sigma_{11} & \sigma_{12} \\ \sigma_{21} & \sigma_{22} \end{bmatrix} = \begin{bmatrix} \sigma_1^2 & \sigma_{12} \\ \sigma_{21} & \sigma_2^2 \end{bmatrix} = \begin{bmatrix} 0,1296 & 0,2341 \\ 0,2341 & 0,4228 \end{bmatrix}$$

→ La covarianza entre (X_i, X_j) (términos cruzados σ_{ij}) son una medida del grado de dependencia lineal que presentan las variables (X_i, X_j) :

Coeficiente de correlación de Pearson de X_i, X_j : $\rho_{ij} = \frac{\sigma_{ij}}{\sigma_i \sigma_j}$

→ En el ejemplo:

$$\rho_{12} = \frac{\sigma_{12}}{\sigma_1 \sigma_2} = \frac{0,2341}{\sqrt{0,1296} \sqrt{0,4228}} = 1$$



DISTRIBUCIÓN NORMAL MULTIVARIANTE

PARÉNTESIS – NOCIONES MATEMÁTICAS: MATRIZ DE COVARIANZAS

EJEMPLOS SIGNIFICADO MATRIZ DE COVARIANZAS PARA DISTINTOS CONJUNTO DE n DATOS DESCRITOS POR $X = (X_1, X_2)$

$$\begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} X_{11} & X_{12} \\ \vdots & \vdots \\ X_{n1} & X_{n2} \end{bmatrix}$$

- EJEMPLO 2: variables no correlacionadas

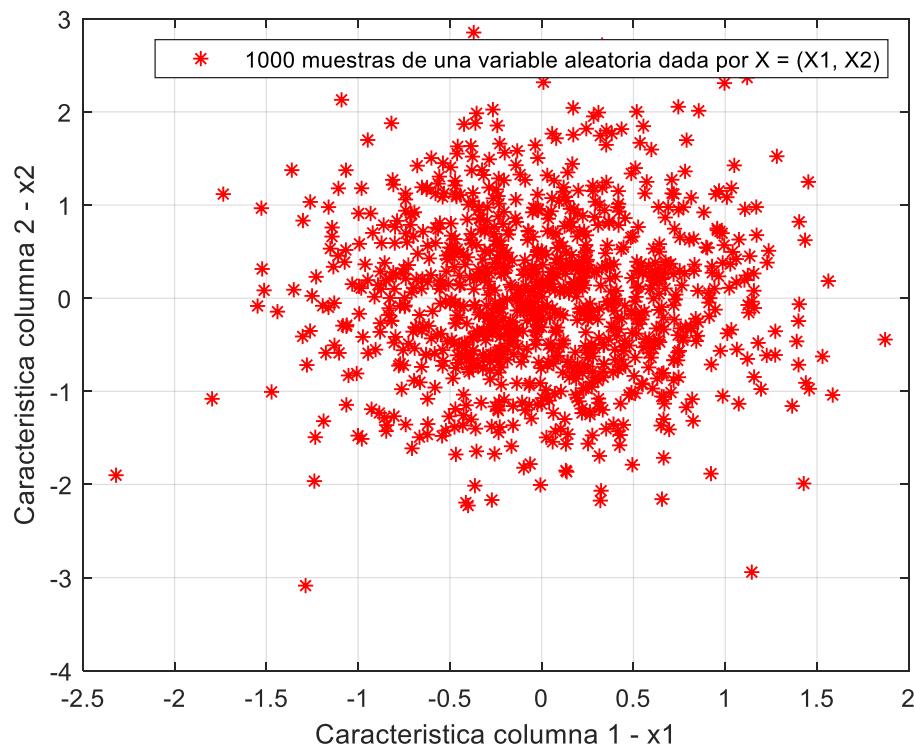
$$\Sigma = \begin{bmatrix} \sigma_{11} & \sigma_{12} \\ \sigma_{21} & \sigma_{22} \end{bmatrix} = \begin{bmatrix} \sigma_1^2 & \sigma_{12} \\ \sigma_{21} & \sigma_2^2 \end{bmatrix} = \begin{bmatrix} 0,3596 & 0,003 \\ 0,003 & 0,7218 \end{bmatrix}$$

→ Variables (X_i, X_j) estadísticamente independientes (no presentan correlación $\rho_{ij} = 0$):

$$\text{Cov}(X_i, X_j) = \sigma_{ij} = 0$$

→ En el ejemplo:

$$\rho_{12} = \frac{\sigma_{12}}{\sigma_1 \sigma_2} = \frac{0,003}{\sqrt{0,3596} \sqrt{0,7218}} = 5,77 * 10^{-4}$$



DISTRIBUCIÓN NORMAL MULTIVARIANTE

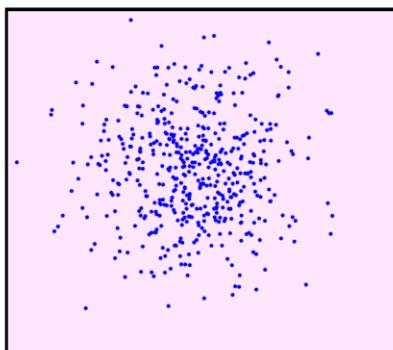
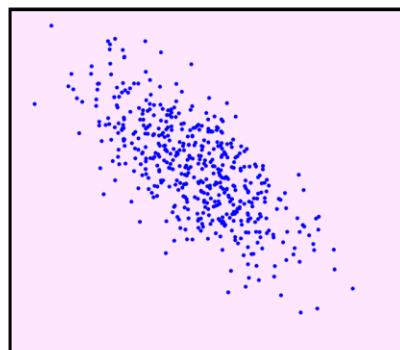
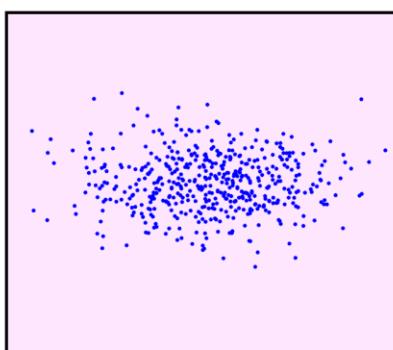
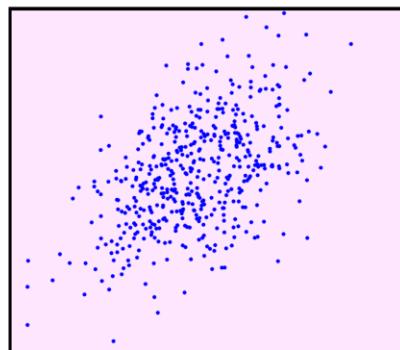
PARÉNTESIS – NOCIONES MATEMÁTICAS: MATRIZ DE COVARIANZAS

EJEMPLOS SIGNIFICADO MATRIZ DE COVARIANZAS PARA

DISTINTOS CONJUNTO DE n DATOS DESCRITOS POR $X = (X_1, X_2)$

$$\rightarrow X = \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} X_{11} & X_{12} \\ \vdots & \vdots \\ X_{n1} & X_{n2} \end{bmatrix}$$

- **EJEMPLO 3:** relaciona cada matriz de covarianzas con el conjunto de datos



$$\Sigma_1 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

$$\Sigma_2 = \begin{pmatrix} 1 & -0.7 \\ -0.7 & 1 \end{pmatrix}$$

$$\Sigma_3 = \begin{pmatrix} 1 & 0,5 \\ 0,5 & 1 \end{pmatrix}$$

$$\Sigma_4 = \begin{pmatrix} 5 & 0 \\ 0 & 1 \end{pmatrix}$$

Notar que cuando las desviaciones estándar de dos variables son 1, su covarianza coincide con el coeficiente de correlación.

$$\rho_{ij} = \frac{\sigma_{ij}}{\sigma_i \sigma_j}$$

DISTRIBUCIÓN NORMAL MULTIVARIANTE DE UNA VARIABLE ALEATORIA $x = (x_1, \dots, x_p)$

DEFINICIÓN: El vector aleatorio $X = [X_1, X_2, \dots, X_p]^T$ es normal p -dimensional con vector de medias

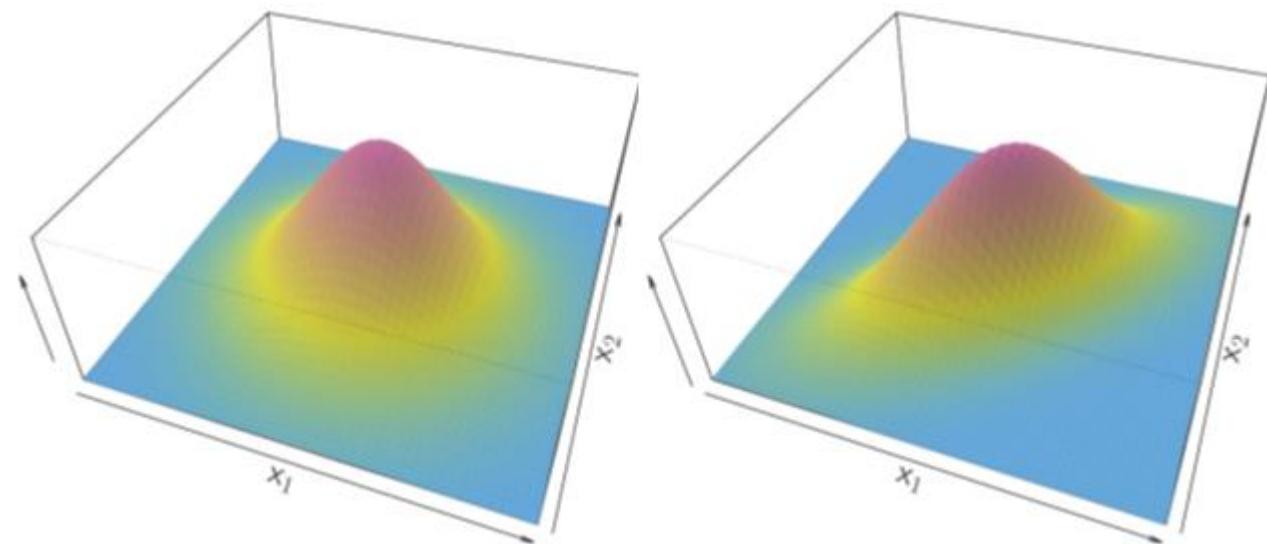
$\mu = [\mu_1, \mu_2, \dots, \mu_p]^T$ y matriz de covarianzas $\Sigma = \begin{bmatrix} \sigma_{11} & \sigma_{12} & \cdots & \sigma_{1p} \\ \sigma_{21} & \sigma_{22} & \cdots & \sigma_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{p1} & \sigma_{p2} & \cdots & \sigma_{pp} \end{bmatrix}$, si su función densidad de probabilidad

está dada por: dada por:

$$X \sim N(\mu, \Sigma) \rightarrow f(x) = \frac{1}{(2\pi)^{p/2} |\Sigma|^{1/2}} \exp \left[-\frac{1}{2} (x - \mu)^T \Sigma^{-1} (x - \mu) \right], x \in R^p$$

Ejemplos de funciones de densidad gaussianas multivariantes (caso de dos dimensiones):

- Figura de la izquierda: caso de variables no correlacionadas.
- Figura de la derecha, caso de variables que presentan una correlación de 0,7.

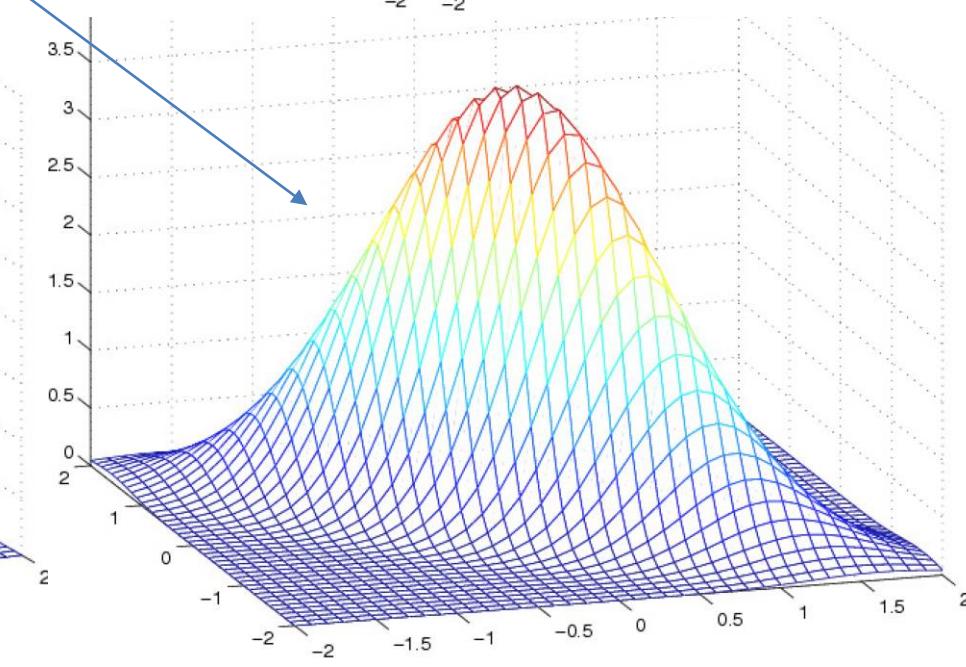
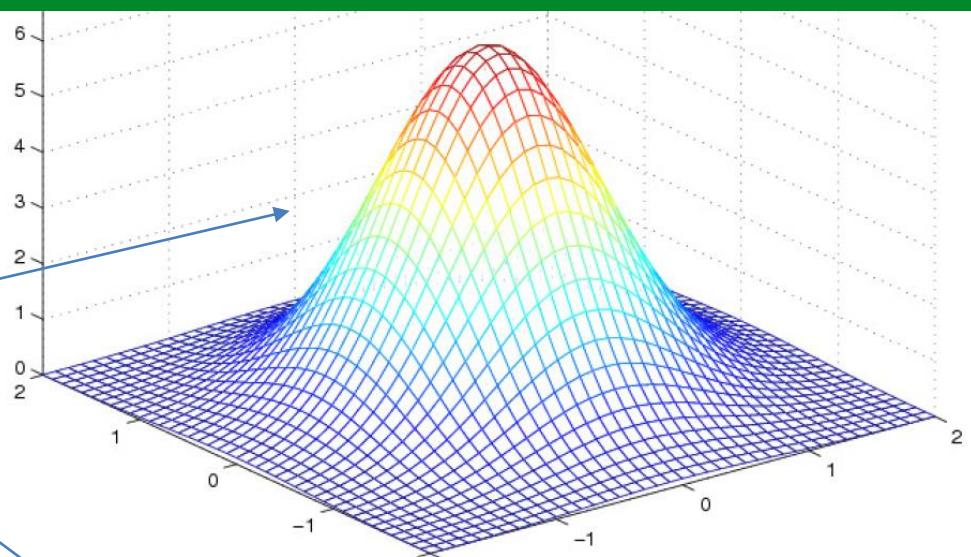
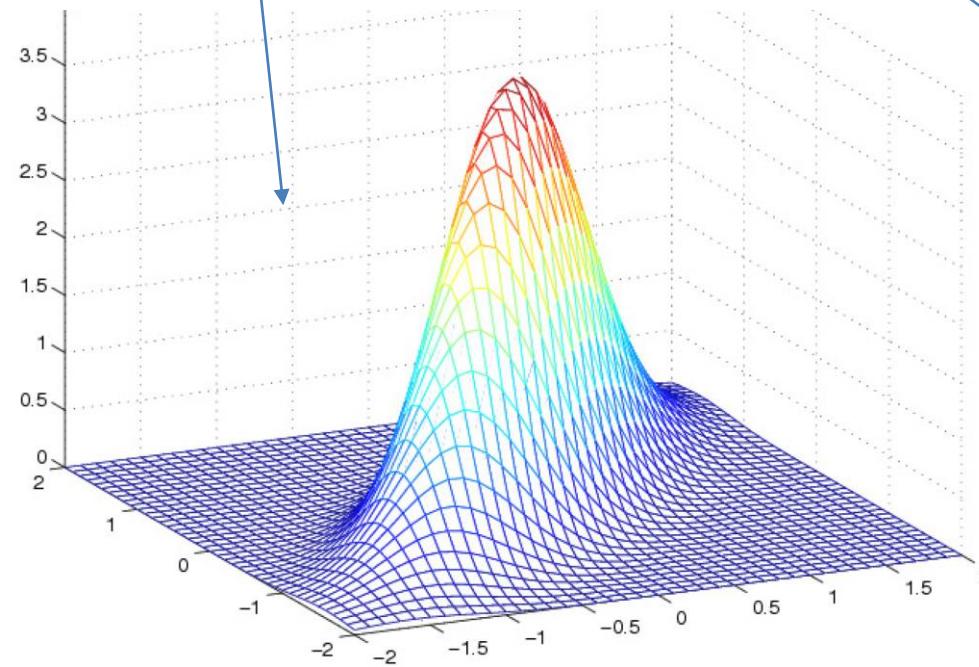


**EJEMPLOS DE DENSIDADES
NORMALES BIDIMENSIONALES**

$$\mu = \begin{bmatrix} 0 \\ 0 \end{bmatrix}; \Sigma = \begin{bmatrix} 1 & 0,8 \\ 0,8 & 1 \end{bmatrix}$$

$$\mu = \begin{bmatrix} 0 \\ 0 \end{bmatrix}; \Sigma = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

$$\mu = \begin{bmatrix} 0 \\ 0 \end{bmatrix}; \Sigma = \begin{bmatrix} 1 & -0,8 \\ -0,8 & 1 \end{bmatrix}$$



RECONOCIMIENTO DE OBJETOS

TECNICAS BÁSICAS DE CLASIFICACIÓN

- Análisis discriminante
 - ✓ Clasificación basada en el Teorema de Bayes
 - ✓ Clasificación basada en distribución normal multivariante
 - ✓ Clasificador QDA: análisis discriminante cuadrático
 - ✓ Clasificador LDA: análisis discriminante lineal
 - ✓ Casos particulares LDA: clasificadores mínima distancia

RECORDANDO: ANÁLISIS DISCRIMINANTE - Clasificación basada en distribución normal multivariante

Instancia a clasificar: $x = (X_1, \dots, X_p)$

1. Teorema de Bayes: Clasificador Decisor Máximo A Posteriori:

$$d_k(x) = \Pr(X=x | Y=k) \Pr(Y=k) = f_k(x) \pi_k$$

2. Estimación de la verosimilitud o probabilidad a posteriori de x en la clase k , $f_k(x)$, asumiendo que los datos de entrenamiento X siguen una distribución normal multivariante (p -dimensional) para cada clase del problema:

$$X \sim N(\mu_k, \Sigma_k) \rightarrow f_k(x) = \frac{1}{(2\pi)^{p/2} |\Sigma_k|^{1/2}} \exp \left[-\frac{1}{2}(x - \mu_k)^T (\Sigma_k)^{-1} (x - \mu_k) \right]$$

3. A partir del conjunto de datos de entrenamiento, diseño de una función de decisión para cada clase del problema:

$$d_k(x) = f_k(x) \pi_k \Rightarrow d_k(x) = \log[f_k(x) \pi_k] = \log[f_k(x)] + \log[\pi_k] \text{ con } f_k(x) = N(\mu_k, \Sigma_k)$$

donde $x = (X_1, \dots, X_p)$ representa los valores de los predictores de cualquier muestra a clasificar

4. Criterio de clasificación: evaluar cada función de decisión para la observación a clasificar dada por $X = x$ y asociarla a la clase cuya función de decisión es máxima.

$$Y(x) = i \quad \text{si} \quad d_i(x) > d_j(x) \quad \forall j \neq i$$

K clases ; Instancia a clasificar: $x = (X_1, \dots, X_p)$; Datos de entrenamiento: $X = \begin{bmatrix} X_1 \\ \vdots \\ X_n \end{bmatrix} = \begin{bmatrix} X_{11} & \cdots & X_{1p} \\ \vdots & \ddots & \vdots \\ X_{n1} & \cdots & X_{np} \end{bmatrix}$; $Y = \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix}$

CLASIFICADOR QDA: ANÁLISIS DISCRIMINANTE CUADRÁTICO

$$d_k(x) = \log[f_k(x)] + \log[\pi_k] \text{ con } f_k(x) = N(\mu_k, \Sigma_k) = \frac{1}{(2\pi)^{p/2} |\Sigma_k|^{1/2}} \exp\left[-\frac{1}{2}(x - \mu_k)^T (\Sigma_k)^{-1} (x - \mu_k)\right]$$

$$d_k(x) = \log\left[\frac{1}{(2\pi)^{p/2} |\Sigma_k|^{1/2}} \exp\left[-\frac{1}{2}(x - \mu_k)^T (\Sigma_k)^{-1} (x - \mu_k)\right]\right] + \log[\pi_k] =$$

$$= -\frac{1}{2}(x - \mu_k)^T (\Sigma_k)^{-1} (x - \mu_k) + \log\left[\frac{1}{(2\pi)^{p/2} |\Sigma_k|^{1/2}}\right] + \log[\pi_k] =$$

$$= -\frac{1}{2}(x - \mu_k)^T (\Sigma_k)^{-1} (x - \mu_k) - \frac{p}{2} \log 2\pi - \frac{1}{2} \log |\Sigma_k| + \log[\pi_k]$$



$$d_k(x) = -\frac{1}{2}(x - \mu_k)^T (\Sigma_k)^{-1} (x - \mu_k) - \frac{1}{2} \log |\Sigma_k| + \log[\pi_k]$$



$$Y(x) = i \quad si \quad d_i(x) > d_j(x) \quad \forall \quad j \neq i$$

CLASIFICADOR QDA: ANÁLISIS DISCRIMINANTE CUADRÁTICO

$$d_k(x) = -\frac{1}{2}(x - \mu_k)^T(\Sigma_k)^{-1}(x - \mu_k) - \frac{1}{2}\log|\Sigma_k| + \log[\pi_k]$$

Para diseñar las funciones de decisión de cada clase, se requiere calcular a partir del conjunto de entrenamiento:

1. **Vector de medias μ_k y matriz de covarianzas Σ_k de la clase k** (calculados sobre las n_k observaciones de la clase k disponibles en el número total n de observaciones de entrenamiento):

$$x = [X_1, X_2, \dots, X_p]^T ; \quad \mu_k = [\mu_1^k, \mu_2^k, \dots, \mu_p^k]^T \quad \text{con } \mu_i^k = E[X_i^k] = \frac{1}{n_k} \sum_{z: y_z=k} X_{zi}$$

$$\Sigma_k = \begin{bmatrix} \sigma_{11}^k & \sigma_{12}^k & \cdots & \sigma_{1p}^k \\ \sigma_{21}^k & \sigma_{22}^k & \cdots & \sigma_{2p}^k \\ \vdots & \vdots & \cdots & \vdots \\ \sigma_{p1}^k & \sigma_{p2}^k & \cdots & \sigma_{pp}^k \end{bmatrix} \quad \text{con } \sigma_{ij}^k = \sigma_{ji}^k = E[(X_i^k - \mu_i^k)(X_j^k - \mu_j^k)] = \frac{1}{n_k - 1} \sum_{z: y_z=k} (X_{zi} - \mu_i^k)(X_{zj} - \mu_j^k)$$

2. **Probabilidad de priori que tiene una muestra de pertenecer a la clase de la clase k :**

- Si no se tiene conocimiento a priori que pueda ser utilizado para estimar esta probabilidad, se suele calcular a partir de la proporción de las muestras de entrenamiento de pertenecer a la clase en cuestión:

$$\pi_k = \frac{n_k}{n}$$

CLASIFICADOR QDA: Observaciones

$$d_k(x) = -\frac{1}{2}(x - \mu_k)^T(\Sigma_k)^{-1}(x - \mu_k) - \frac{1}{2}\log|\Sigma_k| + \log[\pi_k]$$

- Divide el espacio de características en tantas regiones como clases tenga el problema mediante **fronteras de decisión cuadráticas**.
- **Si clases equiprobables** ($\pi_1 = \pi_2 = \dots = \pi_K$) $\Rightarrow d_k(x) = -(x - \mu_k)^T(\Sigma_k)^{-1}(x - \mu_k) - \log|\Sigma_k|$
- **Clasificador que requiere el ajuste de un número elevado de parámetros, especialmente en el caso de tener un alto número de predictores:**

- La matriz de covarianzas de una clase, Σ_k , tiene $\frac{p(p+1)}{2}$ parámetros, por lo que un problema de clasificación con K clases implica el ajuste de $K\frac{p(p+1)}{2}$ parámetros (si $p = 20$, $K = 5$: 1050 parámetros).
 - El clasificador requiere un número elevado de datos de entrenamiento.



CLASIFICADOR LDA: ANÁLISIS DISCRIMINANTE LINEAL

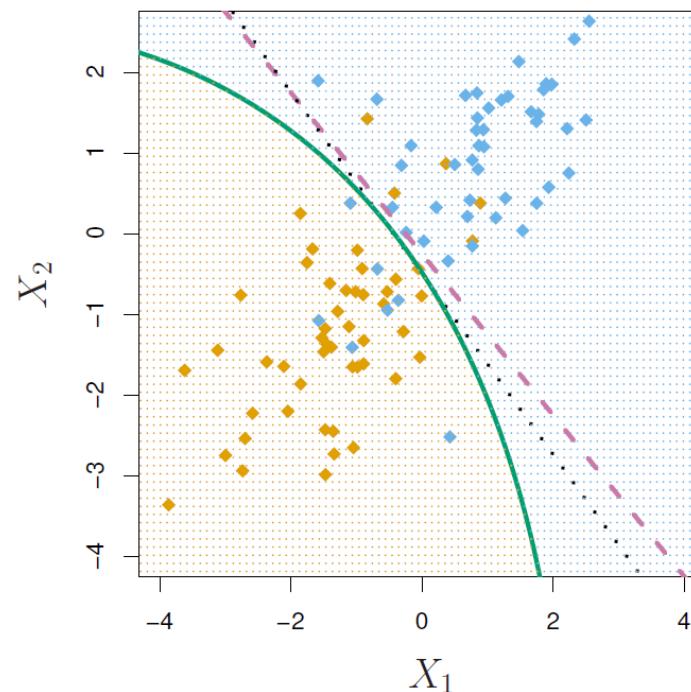
APROXIMACIÓN AL PROBLEMA: *asume que las clases tienen la misma matriz de covarianzas*

- División del espacio de características mediante **fronteras de decisión lineales**.
- LDA es un modelo menos flexible que QDA (implica el ajuste de un número mucho más reducido de parámetros).
- Modelo adecuado cuando el número de datos de entrenamiento es reducido o cuando las clases tienen matrices de covarianzas similares y puede asumirse una única matriz de covarianzas para todas ellas.

FRONTERAS DE DECISIÓN - CLASIFICADOR QDA VS LDA: ejemplo

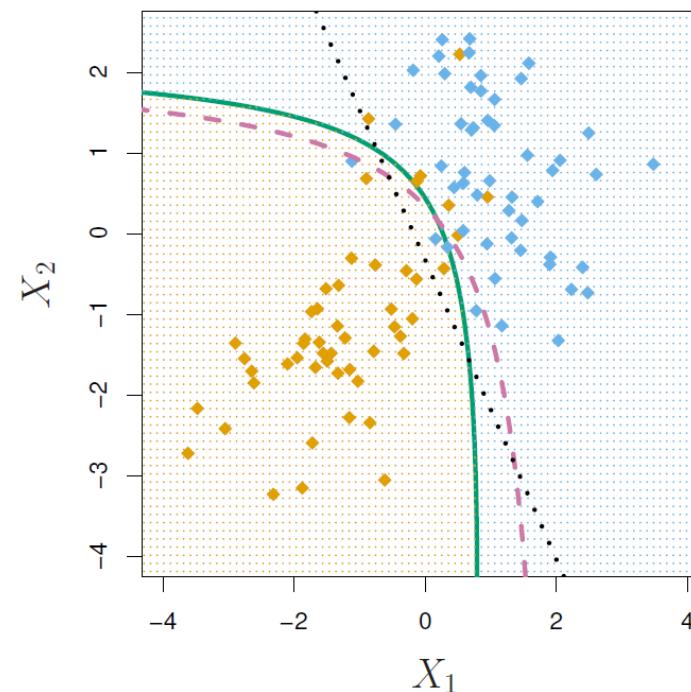
→ Conjunto de datos simulados compuesto por observaciones descritas por dos predictores y correspondiendo a dos clases (los datos de cada clase se han generado con una distribución de probabilidad conocida).

- Línea discontinua morada: frontera de decisión del clasificador Bayesiano (referencia ideal)
- Línea discontinua de puntos: frontera de decisión del clasificador LDA
- Línea continua verde: frontera de decisión del clasificador QDA



Datos generados con $\Sigma_1 = \Sigma_2$

Clasificador LDA se aproxima más al de Bayes



Datos generados con $\Sigma_1 \neq \Sigma_2$

Clasificador QDA se aproxima más al de Bayes

RECONOCIMIENTO DE OBJETOS

TECNICAS BÁSICAS DE CLASIFICACIÓN

- Análisis discriminante
 - ✓ Clasificación basada en el Teorema de Bayes
 - ✓ Clasificación basada en distribución normal multivariante
 - ✓ Clasificador QDA: análisis discriminante cuadrático
 - ✓ Clasificador LDA: análisis discriminante lineal
 - ✓ Casos particulares LDA: clasificadores mínima distancia

K clases ; Instancia a clasificar: $x = (X_1, \dots, X_p)$; Datos de entrenamiento:

$$X = \begin{bmatrix} X_1 \\ \vdots \\ X_n \end{bmatrix} = \begin{bmatrix} X_{11} & \cdots & X_{1p} \\ \vdots & \ddots & \vdots \\ X_{n1} & \cdots & X_{np} \end{bmatrix}; \quad Y = \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix}$$

CLASIFICADOR QDA: $d_k(x) = -\frac{1}{2}(x - \mu_k)^T(\Sigma_k)^{-1}(x - \mu_k) - \frac{1}{2}\log|\Sigma_k| + \log[\pi_k]$

$$x = [X_1, X_2, \dots, X_p]^T$$

CLASIFICADOR LDA – ANÁLISIS DISCRIMINANTE LINEAL

$$\Sigma_1 = \Sigma_2 = \cdots = \Sigma_K = \Sigma$$



$$d_k(x) = -\frac{1}{2}(x - \mu_k)^T(\Sigma)^{-1}(x - \mu_k) + \log[\pi_k]$$

Si conjunto de entrenamiento balanceado en las clases (clases equiprobables)

LDA requiere calcular a partir del conjunto de datos entrenamiento:

$$\Rightarrow \mu_k = [\mu_1^k, \mu_2^k, \dots, \mu_p^k]^T \text{ con } \mu_j^k = E[X_j^k] = \frac{1}{n_k} \sum_{i: y_i=k} X_{ij}$$

$$d_k(x) = - (x - \mu_k)^T(\Sigma)^{-1}(x - \mu_k)$$

$$\Rightarrow \Sigma = \begin{bmatrix} \sigma_{11} & \sigma_{12} & \cdots & \sigma_{1p} \\ \sigma_{21} & \sigma_{22} & \cdots & \sigma_{2p} \\ \vdots & \vdots & \cdots & \vdots \\ \sigma_{p1} & \sigma_{p2} & \cdots & \sigma_{pp} \end{bmatrix} \text{ con } \sigma_{ij} = \sigma_{ji} = \frac{1}{n-K} \sum_{k=1}^K \sum_{z: y_z=k} (X_{zi} - \mu_i^k)(X_{zj} - \mu_j^k)$$

$$\Rightarrow \pi_k = \frac{n_k}{n}$$

Notar que Σ es una estimación de una matriz

de covarianzas común a todas las clases $\Sigma = \frac{1}{n-K} \sum_{k=1}^K (n_k - 1)\Sigma_k$ calculada a partir de la matriz de covarianzas de cada clase, Σ_k :

DOS FORMAS DE APLICACIÓN LDA: K clases ; Instancia a clasificar: $x = (X_1, \dots, X_p)$

1. **A partir de las funciones de decisión cuadráticas:** diseñar K funciones de decisión (una función de decisión d_k para cada clase k del problema) de forma que una observación dada por $X = x$ se asocie a la clase cuya función de decisión es :

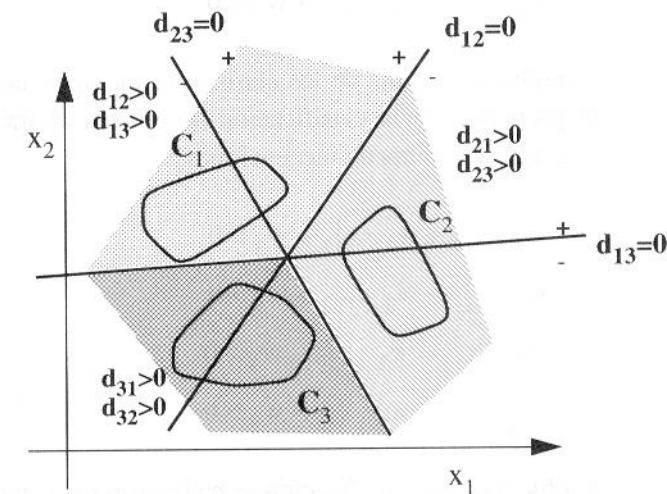
$$d_k(x) = -\frac{1}{2}(x - \mu_k)^T(\Sigma)^{-1}(x - \mu_k) + \log[\pi_k] \longrightarrow Y(x) = i \text{ si } d_i(x) > d_j(x) \quad \forall j \neq i$$

2. **A partir de las fronteras lineales de decisión:** a partir de las funciones de decisión anteriores d_k , determinar $\binom{K}{2} = K(K - 1)/2$ fronteras de decisión para separar las muestras de las clases dos a dos de la siguiente forma:

$$d_{ij}(x) = d_i(x) - d_j(x) = \beta_{ij_0} + \beta_{ij_1}X_1 + \beta_{ij_2}X_2 + \dots + \beta_{ij_p}X_p$$

↓

$$Y(x) = i \text{ si } d_{ij}(x) > 0 \quad \forall j \neq i$$



Ejemplo – Criterio de clasificación para el caso de las 3 clases de la figura:

$$x \in C_1 \text{ si } d_{12} > 0 \text{ y } d_{13} > 0 ; x \in C_2 \text{ si } d_{12} < 0 \text{ y } d_{23} > 0 ; x \in C_3 \text{ si } d_{13} < 0 \text{ y } d_{23} < 0$$

FRONTERAS DE SEPARACIÓN ENTRE CLASES LINEALES :

- Frontera de decisión de las clases $i-j$: HIPERPLANO DADO POR $d_{ij}(x) = 0$
(notar que son los puntos del espacio de predictores que están en la frontera de separación de ambas clases y cumplen que $d_i(x) = d_j(x)$)

RECONOCIMIENTO DE OBJETOS

TECNICAS BÁSICAS DE CLASIFICACIÓN

- Análisis discriminante
 - ✓ Clasificación basada en el Teorema de Bayes
 - ✓ Clasificación basada en distribución normal multivariante
 - ✓ Clasificador QDA: análisis discriminante cuadrático
 - ✓ Clasificador LDA: análisis discriminante lineal
 - ✓ Casos particulares LDA: clasificadores mínima distancia

CLASIFICADOR LDA – ANÁLISIS DISCRIMINANTE LINEAL

$$\Sigma_1 = \Sigma_2 = \dots = \Sigma_K = \Sigma \quad \rightarrow \quad d_k(x) = -\frac{1}{2}(x - \mu_k)^T(\Sigma)^{-1}(x - \mu_k) + \log[\pi_k]$$

$$x = [X_1, X_2, \dots, X_p]^T ; \quad \mu_k = [\mu_1^k, \mu_2^k, \dots, \mu_p^k]^T ; \quad \Sigma = \begin{bmatrix} \sigma_{11} & \sigma_{12} & \cdots & \sigma_{1p} \\ \sigma_{21} & \sigma_{22} & \cdots & \sigma_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{p1} & \sigma_{p2} & \cdots & \sigma_{pp} \end{bmatrix} = \frac{1}{n-K} \sum_{k=1}^K (n_k - 1) \Sigma_k$$

→ CLASIFICADOR MÍNIMA DISTANCIA DE MAHALANOBIS

- Asume clases equiprobables (conjunto de entrenamiento balanceado en las clases): $\pi_1 = \pi_2 = \dots = \pi_K$

$$d_k(x) = - (x - \mu_k)^T(\Sigma)^{-1}(x - \mu_k) = -D_M^2(x, \mu_k)$$

- Este clasificador asigna una observación descrita por x a la clase cuyo vector promedio μ_i esté a distancia de Mahalanobis mínima.
- El criterio es, por tanto, medir la distancia a cada prototipo de las clases y clasificar la instancia cuyo vector prototipo esté más cerca según la Distancia de Mahalanobis (clasificador mediante el prototipo más próximo).

CLASIFICADOR LDA – ANÁLISIS DISCRIMINANTE LINEAL

$$\Sigma_1 = \Sigma_2 = \dots = \Sigma_K = \Sigma \quad \rightarrow \quad d_k(x) = -\frac{1}{2}(x - \mu_k)^T(\Sigma)^{-1}(x - \mu_k) + \log[\pi_k]$$

$$x = [X_1, X_2, \dots, X_p]^T ; \quad \mu_k = [\mu_1^k, \mu_2^k, \dots, \mu_p^k]^T ; \quad \Sigma = \begin{bmatrix} \sigma_{11} & \sigma_{12} & \dots & \sigma_{1p} \\ \sigma_{21} & \sigma_{22} & \dots & \sigma_{2p} \\ \vdots & \vdots & \dots & \vdots \\ \sigma_{p1} & \sigma_{p2} & \dots & \sigma_{pp} \end{bmatrix} = \frac{1}{n-K} \sum_{k=1}^K (n_k - 1) \Sigma_k$$

→ CLASIFICADOR MÍNIMA DISTANCIA EUCLIDEA. Asunciones:

- Clases equiprobables (conjunto de entrenamiento balanceado en las clases): $\pi_1 = \pi_2 = \dots = \pi_K$
- Las variables de los predictores X_i son estadísticamente independientes, no están correladas: $\sigma_{ij} = 0 \quad \forall i \neq j$
- Las varianzas de cada variable predictora X_i son iguales: $\sigma_{ii} = \sigma_i^2 = \sigma^2 \quad \forall i = 1, 2, \dots, p$

$$\Sigma = \begin{bmatrix} \sigma_{11} & \sigma_{12} & \dots & \sigma_{1p} \\ \sigma_{21} & \sigma_{22} & \dots & \sigma_{2p} \\ \vdots & \vdots & \dots & \vdots \\ \sigma_{p1} & \sigma_{p2} & \dots & \sigma_{pp} \end{bmatrix} = \begin{bmatrix} \sigma^2 & 0 & \dots & 0 \\ 0 & \sigma^2 & \dots & 0 \\ \vdots & \vdots & \dots & \vdots \\ 0 & 0 & \dots & \sigma^2 \end{bmatrix} = \sigma^2 \begin{bmatrix} 1 & 0 & \dots & 0 \\ 0 & 1 & \dots & 0 \\ \vdots & \vdots & \dots & \vdots \\ 0 & 0 & \dots & 1 \end{bmatrix} = \sigma^2 I$$

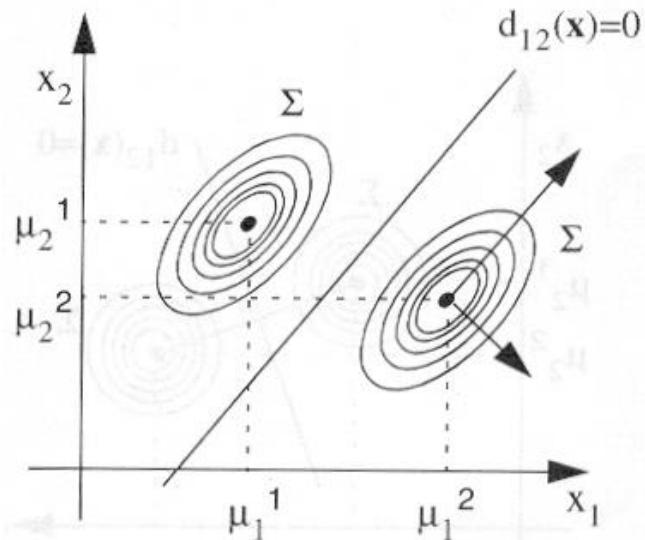
$$d_k(x) = -(x - \mu_k)^T(\Sigma)^{-1}(x - \mu_k) = -\frac{1}{\sigma^2}(x - \mu_k)^T(x - \mu_k) \quad \rightarrow \quad d_k(x) = -(x - \mu_k)^T(x - \mu_k) = -D_E^2(x, \mu_k)$$

- Este clasificador asigna una observación descrita por x a la clase cuyo vector promedio μ_i esté a distancia Euclídea mínima (clasificador mediante el prototipo más próximo según distancia Euclídea).

CASOS PARTICULARES LDA: Clasificadores Mínima Distancia Mahalanobis, Mínima Distancia Euclídea

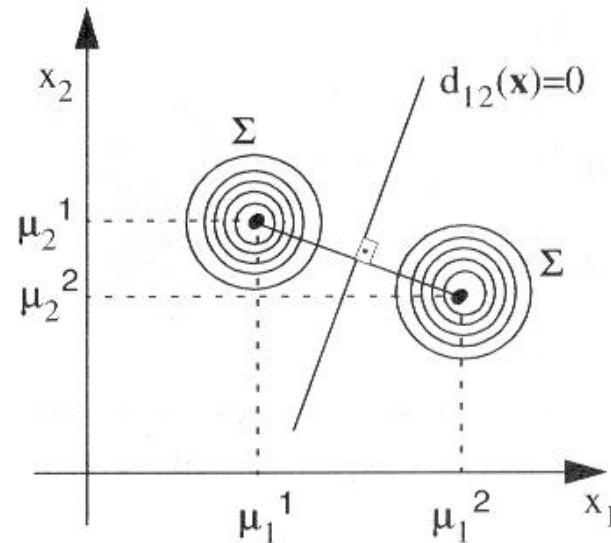
Clasificador de Mínima Distancia de Mahalanobis:

- ⇒ El lugar geométrico de aquellos puntos equidistantes del centro (vector de medias) son hiperelipsoides.
- ⇒ Caso de aplicación:
 - Clases equiprobables.
 - Matrices de covarianzas de las clases similares.
 - Las variables presentan cierta dependencia lineal o correlación entre ellas y/o tienen distinta varianza.



Clasificador de Mínima Distancia Euclídea

- ⇒ El lugar geométrico de aquellos puntos equidistantes del centro (vector de medias) son hiperesferas.
- ⇒ Casos de aplicación:
 - Clases equiprobables.
 - Matrices de covarianzas de las clases similares.
 - Las variables son independientes (no están correladas) y presentan una varianza similar.



PARÉNTESIS – NOCIONES MATEMÁTICAS: DISTANCIA DE MAHALANOBIS

- Sea $X = [X_1, X_2, \dots, X_p]^T$ un vector aleatorio p -dimensional ($p \times 1$). Se dispone de un conjunto de n muestras de X , cuyo vector de medias y matriz de covarianzas son μ y Σ , respectivamente:

$$\rightarrow \text{Vector de medias: } \mu = [\mu_1, \mu_2, \dots, \mu_p]^T, \text{ donde } \mu_i = E[X_i] = \frac{1}{n} \sum_{z=1}^n X_{zi}$$

$$\rightarrow \text{Matriz de covarianzas: } \Sigma = \begin{bmatrix} \sigma_{11} & \sigma_{12} & \cdots & \sigma_{1p} \\ \sigma_{21} & \sigma_{22} & \cdots & \sigma_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{p1} & \sigma_{p2} & \cdots & \sigma_{pp} \end{bmatrix}, \text{ donde } \sigma_{ij} = Cov(X_i, X_j) = \frac{1}{n-1} \sum_{z=1}^n (X_{zi} - \mu_i)(X_{zj} - \mu_j)$$

□ DISTANCIA DE MAHALANOBIS DE X a μ :

$$D_M(X, \mu) = \sqrt{(X - \mu)^T (\Sigma)^{-1} (X - \mu)} \quad \rightarrow \quad D_M^2(X, \mu) = (X - \mu)^T (\Sigma)^{-1} (X - \mu)$$

La distancia de Mahalanobis D_M es adimensional, es una distancia normalizada por la dispersión de los datos (dada por la matriz de covarianzas):

- D_M tiene en cuenta las diferentes variabilidades (varianzas de las variables).
- D_M tiene en cuenta las correlaciones entre las variables (covarianzas de las variables).

PARÉNTESIS – NOCIONES MATEMÁTICAS: DISTANCIA DE MAHALANOBIS

EJEMPLOS SIGNIFICADO DISTANCIA MAHALANOBIS SUPONIENDO

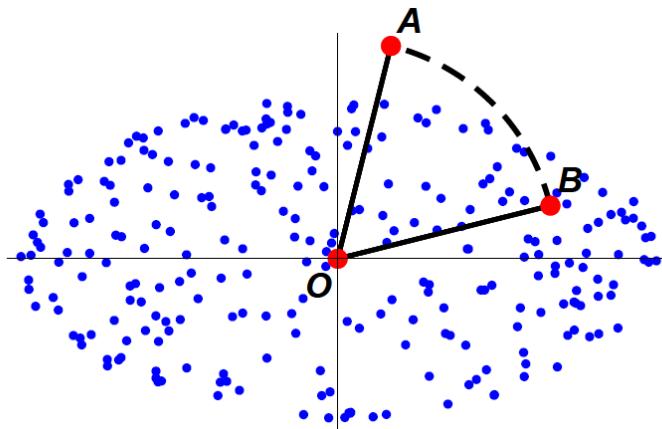
UN CONJUNTO DE n DATOS DESCritos POR $X = (X_1, X_2)$

$$\rightarrow X = \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} X_{11} & X_{12} \\ \vdots & \vdots \\ X_{n1} & X_{n2} \end{bmatrix}$$

- **EJEMPLO 1:** variables no correlacionadas, de distinta varianza

$$\Sigma = \begin{bmatrix} \sigma_1^2 & \sigma_{12} \\ \sigma_{21} & \sigma_2^2 \end{bmatrix} = \begin{bmatrix} 10 & 0 \\ 0 & 2,5 \end{bmatrix}$$

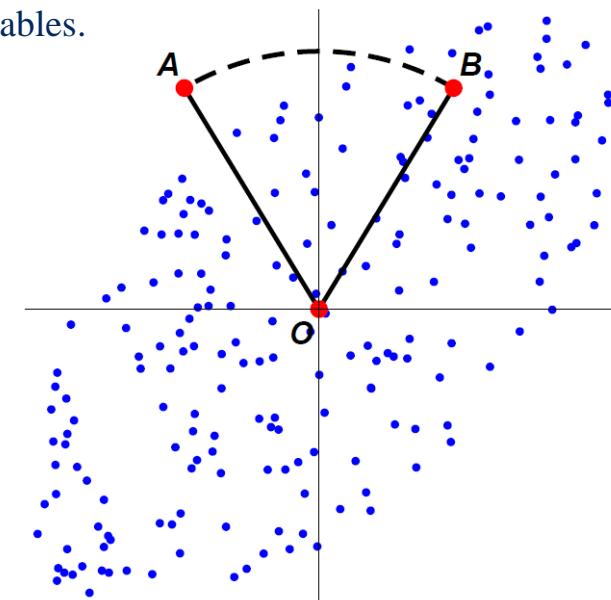
- $D_E(O, A) = D_E(O, B)$, $D_M(O, A) > D_M(O, B)$
- D_M tiene en cuenta las diferentes variabilidades (varianzas de las variables).



- **EJEMPLO 2:** variables correlacionadas, misma varianza

$$\Sigma = \begin{bmatrix} \sigma_1^2 & \sigma_{12} \\ \sigma_{21} & \sigma_2^2 \end{bmatrix} = \begin{bmatrix} 6 & 4 \\ 4 & 6 \end{bmatrix}$$

- $D_E(O, A) = D_E(O, B)$; $D_M(O, A) > D_M(O, B)$
- D_M tiene en cuenta la correlación entre las variables.



- **EJEMPLO 3:** dado el conjunto de datos, compuesto por 5 observaciones de dos variables X_1 y X_2 , comparar los valores de distancia Euclídea y de Mahalanobis entre los puntos O-A y O-B, donde O es el punto medio del conjunto de datos, A (3, 5) y B (6, 7)

$$O = \begin{bmatrix} 3 \\ 3.6 \end{bmatrix} ; \Sigma = \begin{bmatrix} 2.5 & 3.125 \\ 3.125 & 3.925 \end{bmatrix}$$

$$D_E(O, A) = 1.40 ; D_M(O, A) = 10.22$$

$$D_E(O, B) = 4.53 ; D_M(O, B) = 3.18$$

%% CÓDIGO MATLAB:

```
% Datos X:
datos = [1 ,1 ; 2 , 2.5 ; 3 , 3.5 ; 4,5 ; 5 , 6];

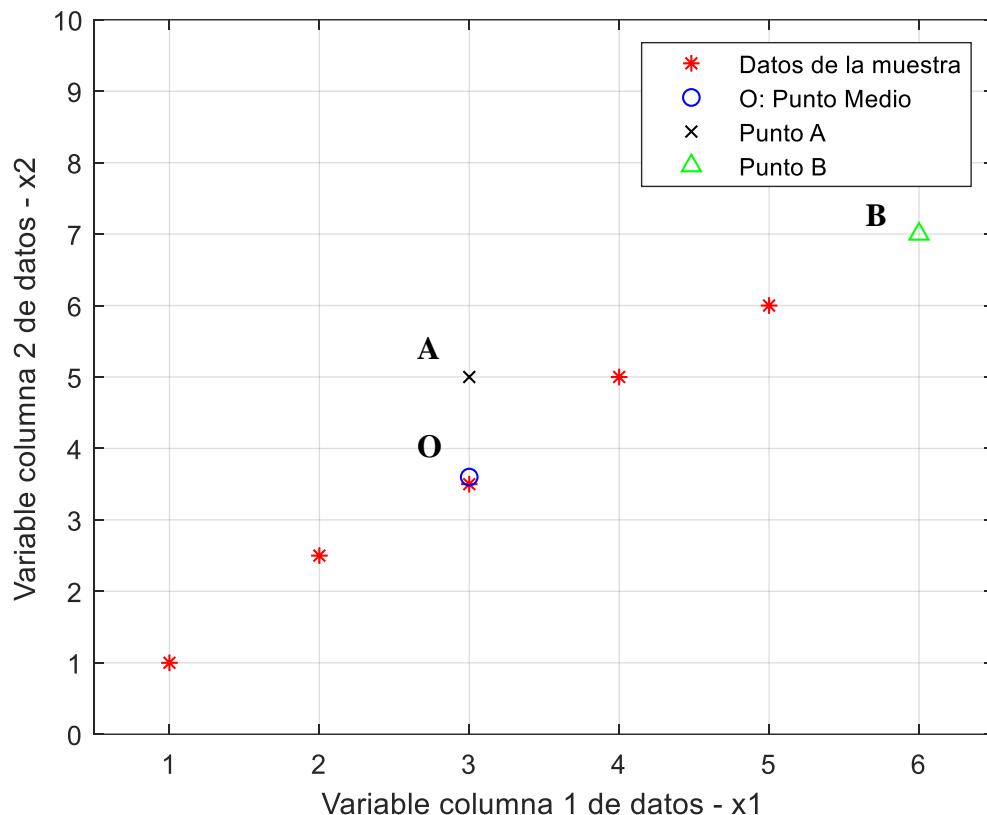
% Vector de medias (Punto O):
O = mean(datos)';

% Definición de X=[x1;x2] simbólico
x1 = sym('x1','real');
x2 = sym('x2','real');
X = [x1 ; x2];

% DE y MD al cuadrado de X al punto O
dE2 = expand((X-O)'*(X - O))
MCov = cov(datos);
dM2 = expand((X-O)'*inv(MCov)*(X - O))

% Obtención de distancias solicitadas
x1 = 3; x2 = 5; dE_OA = sqrt(eval(dE2)); dM_OA = sqrt(eval(dM2));
x1 = 6; x2 = 7; dE_OB = sqrt(eval(dE2)); dM_OB = sqrt(eval(dM2));
```

$$X = \begin{bmatrix} 1 & 1 \\ 2 & 2.5 \\ 3 & 3.5 \\ 4 & 5 \\ 5 & 6 \end{bmatrix} ; A = \begin{bmatrix} 3 \\ 5 \end{bmatrix} ; B = \begin{bmatrix} 6 \\ 7 \end{bmatrix}$$



- **EJEMPLO 4:** dado el conjunto de datos, compuesto por 2000 observaciones de dos clases (1000 observaciones de cada clase) descritas por tres variables X_1 , X_2 y X_3 . Diseña una clasificador LDA.

→ Clases equiprobables (conjunto de entrenamiento balanceado en las clases):

$$\pi_1 = \pi_2$$

→ Matrices de covarianzas de las clases similares.

$$\Sigma_1 \approx \Sigma_2$$

→ Las variables presentan cierta dependencia lineal o correlación entre ellas.



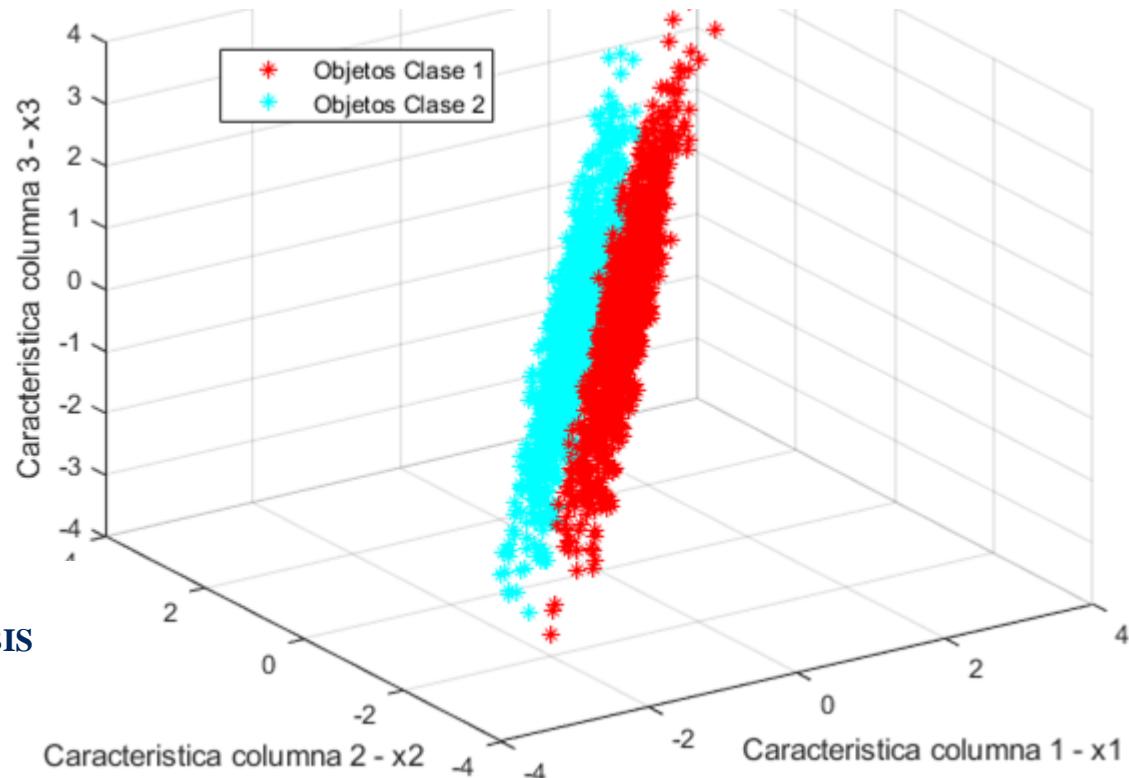
LDA: MÍNIMA DISTANCIA DE MAHALANOBIS

$$x = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \quad d_k = -D_M^2(x, \mu_k)$$

$$d_1(x) = -(x - \mu_1)^T (\Sigma)^{-1} (x - \mu_1)$$

$$d_2(x) = -(x - \mu_2)^T (\Sigma)^{-1} (x - \mu_2)$$

$$x \in C_1 \text{ si } d_1 > d_2 ; \quad x \in C_2 \text{ si } d_1 < d_2$$



$$d_{12}(x) = d_1(x) - d_2(x) = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3$$

$$x \in C_1 \text{ si } d_{12} > 0 ; \quad x \in C_2 \text{ si } d_{12} < 0$$

- **EJEMPLO 4:** dado el conjunto de datos, compuesto por 2000 observaciones de dos clases (1000 observaciones de cada clase) descritas por tres variables X_1 , X_2 y X_3 . Diseña una clasificador LDA.

$$X = \begin{bmatrix} X_1 \\ \vdots \\ X_{2000} \end{bmatrix} = \begin{bmatrix} X_{11} & X_{21} & X_{31} \\ \vdots & \vdots & \vdots \\ X_{12000} & X_{22000} & X_{32000} \end{bmatrix}; \quad Y = \begin{bmatrix} y_1 \\ \vdots \\ y_{2000} \end{bmatrix} \quad Y = \{C_1, C_2\} = \{1, 2\}$$

→ Obtener el vector de medias y matriz covarianzas de cada clase:

$$X \rightarrow X_{C1} = X_1 \quad y \quad X_{C2} = X_2 \quad \rightarrow \quad \mu_1, \Sigma_1 \quad y \quad \mu_2, \Sigma_2$$

$$\mu_1 = \begin{bmatrix} 0,27 \\ -0,12 \\ 0,13 \end{bmatrix} \quad \mu_2 = \begin{bmatrix} -0,27 \\ 0,12 \\ -0,13 \end{bmatrix} \quad \Sigma_1 = \begin{bmatrix} 0,93 & 0,95 & 0,91 \\ 0,95 & 0,99 & 0,94 \\ 0,91 & 0,94 & 1,02 \end{bmatrix} \quad \Sigma_2 = \begin{bmatrix} 0,92 & 0,94 & 0,83 \\ 0,94 & 0,98 & 0,86 \\ 0,83 & 0,86 & 0,96 \end{bmatrix}$$

→ Obtener la matriz de covarianzas común para cada clase:

$$\Sigma = \frac{1}{n-K} \sum_{k=1}^K (n_k - 1) \Sigma_k \Rightarrow \Sigma = \begin{bmatrix} 0,93 & 0,94 & 0,87 \\ 0,94 & 0,99 & 0,90 \\ 0,87 & 0,90 & 0,99 \end{bmatrix}$$

→ Obtener la función de decisión para cada clase: $d_k(x) = -(x - \mu_k)^T (\Sigma)^{-1} (x - \mu_k)$

$$x = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \quad d_1(x) = A_0 + A_1 x_1^2 + A_2 x_2^2 + A_3 x_3^2 + A_4 x_1 x_2 + A_5 x_1 x_3 + A_6 x_2 x_3 + A_7 x_1 + A_8 x_2 + A_9 x_3$$

$$d_2(x) = B_0 + B_1 x_1^2 + B_2 x_2^2 + B_3 x_3^2 + B_4 x_1 x_2 + B_5 x_1 x_3 + B_6 x_2 x_3 + B_7 x_1 + B_8 x_2 + B_9 x_3$$

- **EJEMPLO 4:** dado el conjunto de datos, compuesto por 2000 observaciones de dos clases (1000 observaciones de cada clase) descritas por tres variables X_1, X_2 y X_3 . Diseña una clasificador LDA.

$$X = \begin{bmatrix} x_1 \\ \vdots \\ x_{2000} \end{bmatrix} = \begin{bmatrix} X_{11} & X_{21} & X_{31} \\ \vdots & \vdots & \vdots \\ X_{1_{2000}} & X_{2_{2000}} & X_{3_{2000}} \end{bmatrix}; \quad Y = \begin{bmatrix} y_1 \\ \vdots \\ y_{2000} \end{bmatrix} \quad Y = \{C_1, C_2\} = \{1, 2\}$$

→ Obtener la función discriminante entre las dos clases (frontera de decisión):

$$x = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \quad d_{12}(x) = d_1(x) - d_2(x) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3$$

→ Ejemplo de cálculo de coeficientes $\beta_0, \beta_1, \beta_2, \beta_3$ en Matlab (en el ejemplo: $d_{12}(x) = Ax_1 + Bx_2 + Cx_3 + D$)

```
x1 = sym('x1','real');
x2 = sym('x2','real');
x3 = sym('x3','real');
X = [x1; x2 ; x3];

% Asumiendo que ya se han calculado los vectores
% prototipo de cada clase (vectores columna) y la
% matriz covarianza común: M1, M2, mCov

d1 = expand(-(X-M1) * inv(mCov) * (X - M1));
d2 = expand(-(X-M2) * inv(mCov) * (X - M2));
```

% Función de decisión que separa las muestras de

% las clases:

d12 = d1 - d2;

% Calculo de coeicientes: d12 = A*x1+B*x2+C*x3+D

```
x1 = 0; x2 = 0; x3 = 0; D = eval(d12);
x1 = 1; x2 = 0; x3 = 0; A = eval(d12)-D;
x1 = 0; x2 = 1; x3 = 0; B = eval(d12)-D;
x1 = 0; x2 = 0; x3 = 1; C = eval(d12)-D;
```

- **EJEMPLO 4:** dado el conjunto de datos, compuesto por 2000 observaciones de dos clases (1000 observaciones de cada clase) descritas por tres variables X_1, X_2 y X_3 . Diseña una clasificador LDA.

→ Representación frontera de separación: $d_{12}(x)=0$ con Matlab:

```
%% LDA: FRONTERA DE SEPARACIÓN LINEAL
% (HIPERPLANO)
% 2 DIMENSIONES: LÍNEA RECTA
% 3 DIMENSIONES: PLANO
% d12 = 0

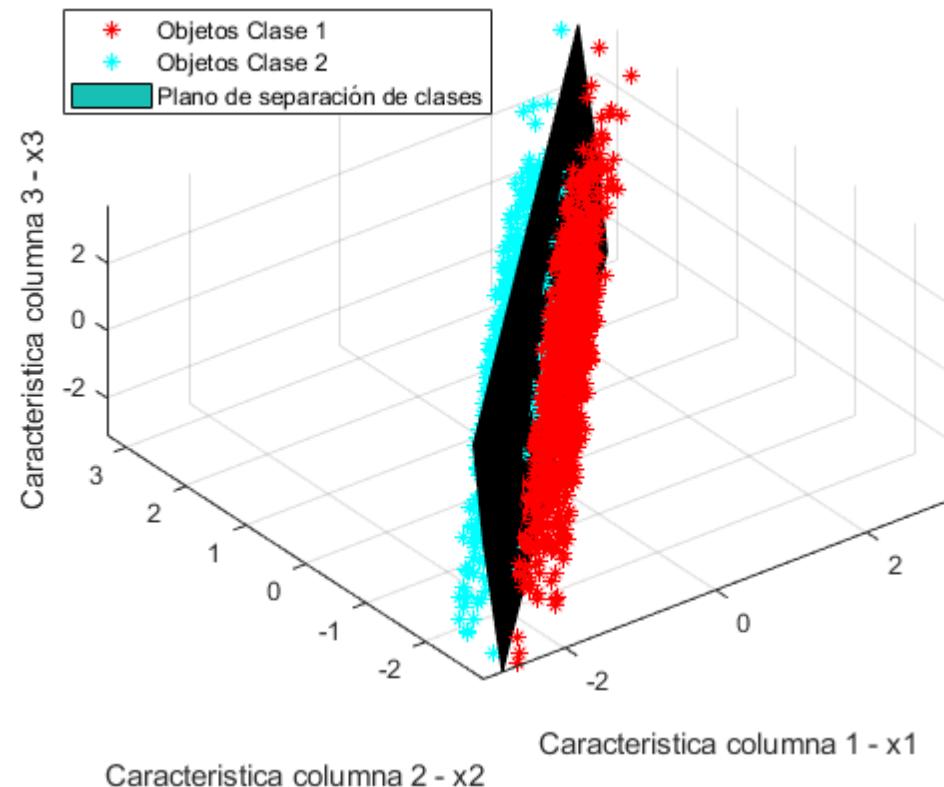
%% 1.- Discretizamos espacio x1-x2 con meshgrid
% Debemos establecer los valores mínimos
% y máximos, así como el paso de discretización

[x1Plano, x2Plano] =
meshgrid(x1min:paso1:x1max,x2min:paso2:x2max);

%% 2.- Determinamos x3, sabiendo que:
% A x1 + Bx2 + Cx3 + D = 0

x3Plano = -(A*x1Plano + B*x2Plano + D) / (C+eps);

%% 3.- Representamos con surf
surf(x1Plano,x2Plano, x3Plano)
```



Ejercicios de ejemplo: tres ejercicios con conjunto de datos muy reducidos y básicos, pensados para resolverlos de forma teórica y refrendar los resultados de forma práctica con Matlab.

- **Ejercicio 1:** Teniendo en cuenta la muestra de la tabla, diseñar un Clasificador de Mínima Distancia Euclídea.

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 1

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

CLASE 2

- **Ejercicio 2:** Teniendo en cuenta la muestra de la tabla, diseñar un Clasificador de Mínima Distancia Mahalanobis estimando una única matriz de covarianzas para ambas clases (no considerar el desbalanceo de las clases, el criterio de clasificación debe ser considerar únicamente la distancia de 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

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

$$X = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}; \pi_1 = \pi_2 = \pi_3; \mu_1 = \begin{bmatrix} 0 \\ 3 \end{bmatrix}; \mu_2 = \begin{bmatrix} 5 \\ 2 \end{bmatrix}; \mu_3 = \begin{bmatrix} 1 \\ 0 \end{bmatrix}; \Sigma_1 = \Sigma_2 = \Sigma_3 = \Sigma = \begin{bmatrix} 1/2 & 0 \\ 0 & 1/4 \end{bmatrix}; \Sigma^{-1} = \begin{bmatrix} 2 & 0 \\ 0 & 4 \end{bmatrix}$$

Ejercicios de ejemplo: tres ejercicios con conjunto de datos muy reducidos y básicos, pensados para resolverlos de forma teórica y refrendar los resultados de forma práctica con Matlab.

□ **Ejercicio 1:** Teniendo en cuenta la muestra de la tabla, diseñar 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

$$x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}, \mu_k = \begin{bmatrix} \mu_1^k \\ \mu_2^k \end{bmatrix} \rightarrow \mu_1 = \begin{bmatrix} \mu_1^1 \\ \mu_2^1 \end{bmatrix} = \begin{bmatrix} 2,5 \\ 2,5 \end{bmatrix}; \mu_2 = \begin{bmatrix} \mu_1^2 \\ \mu_2^2 \end{bmatrix} = \begin{bmatrix} 5,5 \\ 6 \end{bmatrix}$$

Funciones de decisión: $d_k(x) = -D_E^2(x, \mu_k) = -(x - \mu_k)^T(x - \mu_k)$

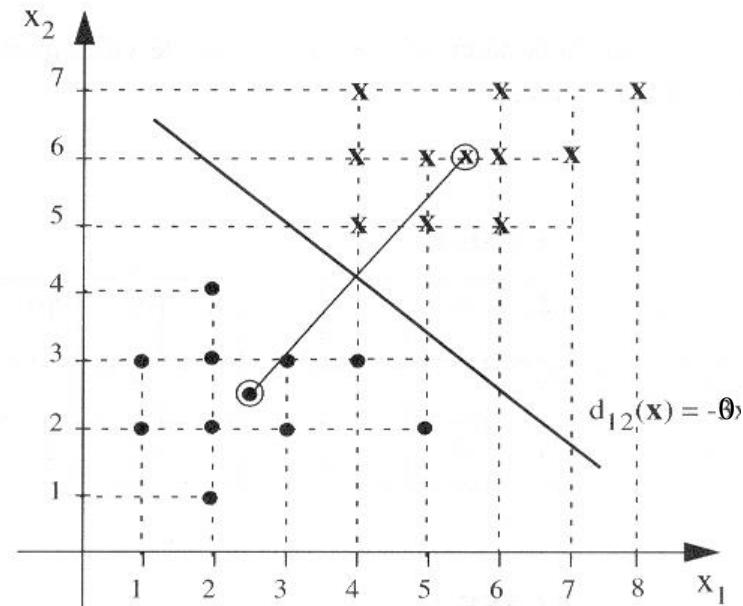
$$\begin{aligned} d_1(x) &= -[x_1 - 2.5 \quad x_2 - 2.5] \begin{bmatrix} x_1 - 2.5 \\ x_2 - 2.5 \end{bmatrix} = -[(x_1 - 2.5)^2 + (x_2 - 2.5)^2] = \\ &= -x_1^2 - x_2^2 + 5x_1 + 5x_2 - 12.5 \end{aligned}$$

$$d_2(x) = -[(x_1 - 5.5)^2 + (x_2 - 6)^2] = -x_1^2 - x_2^2 + 11x_1 + 12x_2 - 66.25$$

Función discriminante: $d_{12}(x) = d_1(x) - d_2(x) = -6x_1 - 7x_2 + 53.75$

Frontera de decisión: $d_{12}(x) = 0 \rightarrow -6x_1 - 7x_2 + 53.75 = 0$

Criterio de clasificación: $x \in C_1 \text{ si } d_{12}(x) > 0 (\equiv d_1(x) > d_2(x)) ; x \in C_2 \text{ en caso contrario}$



Ejercicios de ejemplo: tres ejercicios con conjunto de datos muy reducidos y básicos, pensados para resolverlos de forma teórica y refrendar los resultados de forma práctica con Matlab.

- **Ejercicio 2:** Teniendo en cuenta la muestra de la tabla, diseñar un Clasificador de Mínima Distancia Mahalanobis estimando una única matriz de covarianzas para ambas clases (no considerar el desbalanceo de las clases, el criterio de clasificación debe ser considerar únicamente la distancia de 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

$$x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}, \quad \mu_k = \begin{bmatrix} \mu_1^k \\ \mu_2^k \end{bmatrix} \rightarrow \mu_1 = \begin{bmatrix} \mu_1^1 \\ \mu_2^1 \end{bmatrix} = \begin{bmatrix} 3 \\ 2 \end{bmatrix}; \quad \mu_2 = \begin{bmatrix} \mu_1^2 \\ \mu_2^2 \end{bmatrix} = \begin{bmatrix} 6 \\ 2 \end{bmatrix}$$

$$\Sigma = \begin{bmatrix} \sigma_{11} & \sigma_{12} \\ \sigma_{21} & \sigma_{22} \end{bmatrix} \text{ con } \sigma_{ij} = \sigma_{ji} = \frac{1}{n-K} \sum_{k=1}^K \sum_{z: y_z=k} (X_{zi} - \mu_i^k)(X_{zj} - \mu_j^k)$$

$$\Sigma = \frac{1}{5} \left[\sum_{z=1}^4 \begin{bmatrix} (x_{z1}^1 - 3)^2 & (x_{z1}^1 - 3)(x_{z2}^1 - 2) \\ (x_{z2}^1 - 2)(x_{z1}^1 - 3) & (x_{z2}^1 - 2)^2 \end{bmatrix} + \sum_{z=1}^3 \begin{bmatrix} (x_{z1}^2 - 6)^2 & (x_{z1}^2 - 6)(x_{z2}^2 - 2) \\ (x_{z2}^2 - 2)(x_{z1}^2 - 6) & (x_{z2}^2 - 2)^2 \end{bmatrix} \right] = \frac{1}{5} \begin{bmatrix} 4 & 2 \\ 2 & 4 \end{bmatrix}$$

$$\Sigma^{-1} = \frac{1}{3} \begin{bmatrix} 5 & -5/2 \\ -5/2 & 5 \end{bmatrix}$$

□ **Ejercicio 2:** Teniendo en cuenta la muestra de la tabla, diseñar un Clasificador de Mínima Distancia Mahalanobis estimando una única matriz de covarianzas para ambas clases (no considerar el desbalanceo de las clases, el criterio de clasificación debe ser considerar únicamente la distancia de Mahalanobis):

$$x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}, \mu_k = \begin{bmatrix} \mu_1^k \\ \mu_2^k \end{bmatrix} \rightarrow \mu_1 = \begin{bmatrix} \mu_1^1 \\ \mu_2^1 \end{bmatrix} = \begin{bmatrix} 3 \\ 2 \end{bmatrix}; \mu_2 = \begin{bmatrix} \mu_1^2 \\ \mu_2^2 \end{bmatrix} = \begin{bmatrix} 6 \\ 2 \end{bmatrix}; \Sigma = \frac{1}{5} \begin{bmatrix} 4 & 2 \\ 2 & 4 \end{bmatrix} \rightarrow \Sigma^{-1} = \frac{1}{3} \begin{bmatrix} 5 & -5/2 \\ -5/2 & 5 \end{bmatrix}$$

Funciones de decisión: $d_k(x) = -D_M^2(x, \mu_k) = -(x - \mu_k)^T \Sigma^{-1} (x - \mu_k)$

$$d_1(x) = -\frac{1}{3} [x_1 - 3 \quad x_2 - 2] \begin{bmatrix} 5 & -2,5 \\ -2,5 & 5 \end{bmatrix} \begin{bmatrix} x_1 - 3 \\ x_2 - 2 \end{bmatrix} = \frac{20x_1}{3} + \frac{5x_2}{3} + \frac{5x_1x_2}{3} - \frac{5x_1^2}{3} - \frac{5x_2^2}{3} - \frac{35}{3}$$

$$d_2(x) = \frac{50x_1}{3} - \frac{10x_2}{3} + \frac{5x_1x_2}{3} - \frac{5x_1^2}{3} - \frac{5x_2^2}{3} - \frac{140}{3}$$

Función discriminante: $d_{12}(x) = d_1(x) - d_2(x)$

$$d_{12}(x) = d_1(x) - d_2(x) = -10x_1 + 5x_2 + 35$$

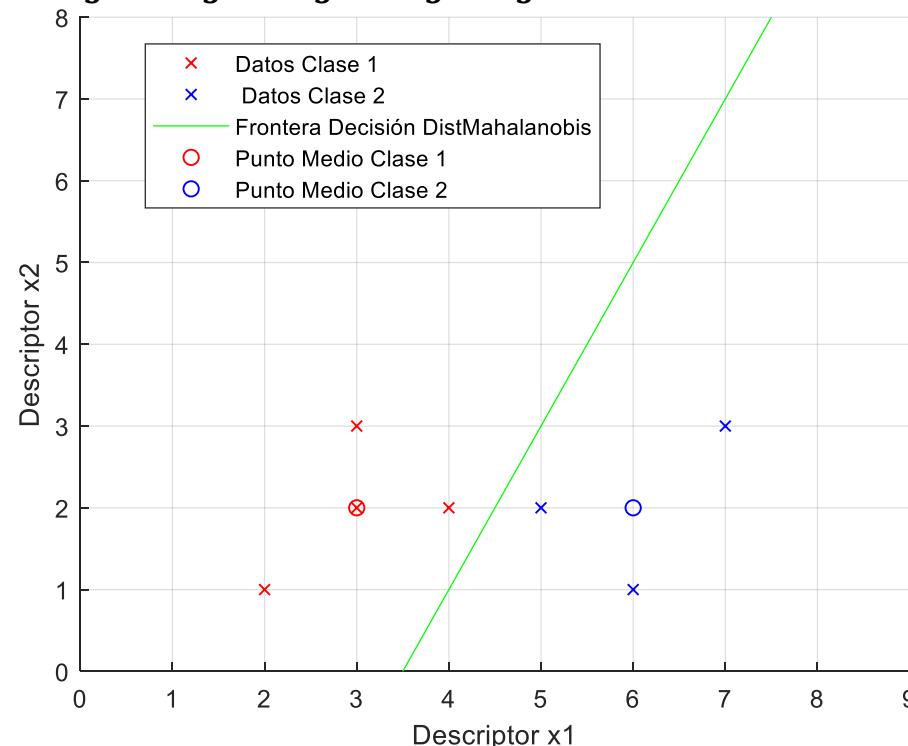
Frontera de decisión: $d_{12}(x) = 0$

$$-10x_1 + 5x_2 + 35 = 0$$

Criterio de clasificación:

$$x \in C_1 \text{ si } d_{12}(x) > 0 (\equiv d_1(x) > d_2(x))$$

$x \in C_2$ en caso contrario



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

$$X = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}; \boldsymbol{\pi}_1 = \boldsymbol{\pi}_2 = \boldsymbol{\pi}_3; \boldsymbol{\mu}_1 = \begin{bmatrix} 0 \\ 3 \end{bmatrix}; \boldsymbol{\mu}_2 = \begin{bmatrix} 5 \\ 2 \end{bmatrix}; \boldsymbol{\mu}_3 = \begin{bmatrix} 1 \\ 0 \end{bmatrix}; \Sigma_1 = \Sigma_2 = \Sigma_3 = \Sigma = \begin{bmatrix} 1/2 & 0 \\ 0 & 1/4 \end{bmatrix}; \Sigma^{-1} = \begin{bmatrix} 2 & 0 \\ 0 & 4 \end{bmatrix}$$

Funciones de decisión:

$$d_1(x) = -2x_1^2 - 4x_2^2 + 24x_2 + 36$$

$$d_2(x) = -2x_1^2 - 4x_2^2 + 20x_1 + 16x_2 - 66$$

$$d_3(x) = -2x_1^2 - 4x_2^2 + 4x_1 - 2$$

Funciones discriminantes entre clases:

$$d_{12}(x) = d_1(x) - d_2(x) = -20x_1 + 8x_2 + 30$$

$$d_{13}(x) = d_1(x) - d_3(x) = -4x_1 + 24x_2 - 34$$

$$d_{23}(x) = d_2(x) - d_3(x) = 16x_1 + 16x_2 - 64$$

Fronteras de decisión:

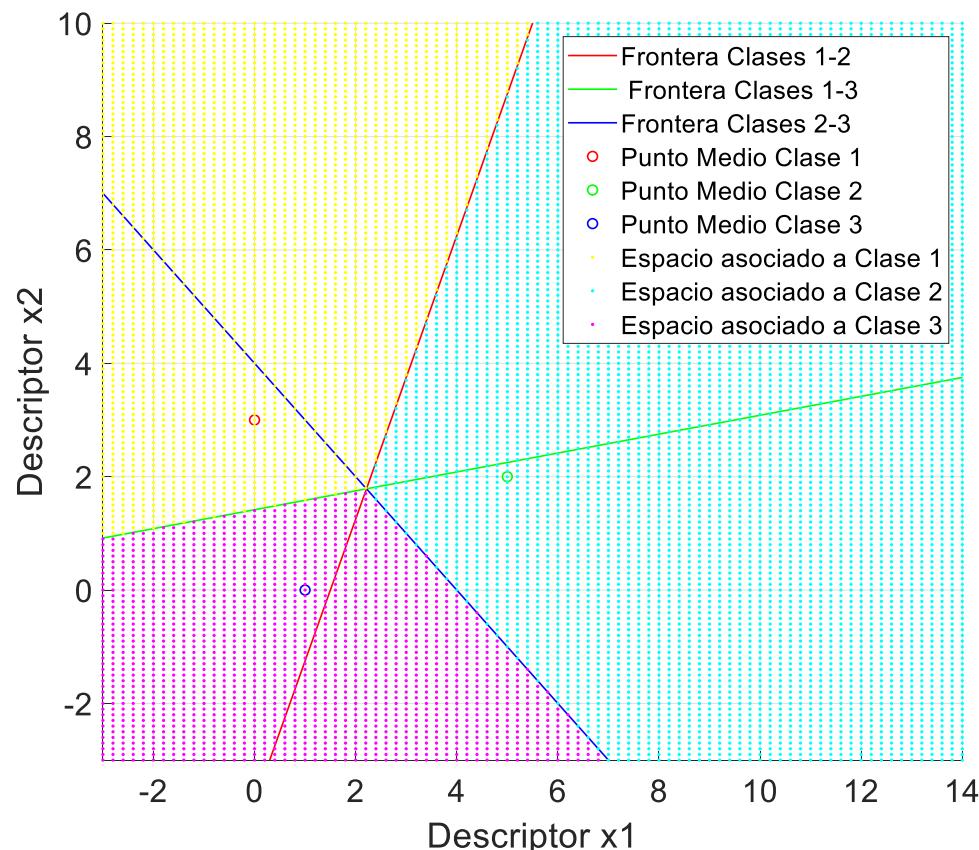
$$d_{12} = 0 \rightarrow -10x_1 + 4x_2 + 15 = 0$$

$$d_{13} = 0 \rightarrow -2x_1 + 12x_2 - 17 = 0$$

$$d_{23} = 0 \rightarrow x_1 + x_2 - 4 = 0$$

Criterio de clasificación: $x \in C_1$ si $d_{12}(x) > 0$ y $d_{13}(x) > 0$

$x \in C_2$ si $d_{12}(x) < 0$ y $d_{23}(x) > 0$; $x \in C_3$ si $d_{13}(x) < 0$ y $d_{23}(x) < 0$



BIBLIOGRAFÍA PRINCIPAL

- James G., Witten D.,Hastie T. y Tibshirani R (2017). "An Introduction to Statistical Learning, with applications in R", Springer, Recurso libre: <http://faculty.marshall.usc.edu/gareth-james/ISL/index.html>

Otras referencias consultadas y fuente de figuras:

- "VISIÓN POR COMPUTADOR" (Paraninfo, 1999), González Jiménez, J.
- "Aprendizaje automático para el análisis de datos", Grado en Estadística y Empresa, Ricardo Aler, Universidad Carlos III de Madrid. OpenCourseWare: <http://ocw.uc3m.es/ingenieria-informatica/aprendizaje-automatico-para-el-analisis-de-datos>
- "Some of the figures in this presentation are taken from "An Introduction to Statistical Learning, with applications in R" (Springer, 2013) with permission from the authors: G. James, D. Witten, T. Hastie and R. Tibshirani "

RECONOCIMIENTO DE OBJETOS

- INTRODUCCIÓN
 - Clasificación: enfoque basado en la teoría de la Decisión
- TECNICAS BÁSICAS DE CLASIFICACIÓN
 - Análisis discriminante
 - K-vecinos más cercanos
- DESARROLLO DE SISTEMAS DE RECONOCIMIENTO DE OBJETOS: ASPECTOS PRÁCTICOS
 - Análisis y pre-procesamiento de datos
 - Selección de atributos
 - Evaluación de modelos

RECONOCIMIENTO DE OBJETOS

- INTRODUCCIÓN
 - Clasificación: enfoque basado en la teoría de la Decisión
- TECNICAS BÁSICAS DE CLASIFICACIÓN
 - Análisis discriminante
 - K-vecinos más cercanos
- DESARROLLO DE SISTEMAS DE RECONOCIMIENTO DE OBJETOS: ASPECTOS PRÁCTICOS
 - Análisis y pre-procesamiento de datos
 - Selección de atributos
 - Evaluación de modelos

RECONOCIMIENTO DE OBJETOS

□ TECNICAS BÁSICAS DE CLASIFICACIÓN

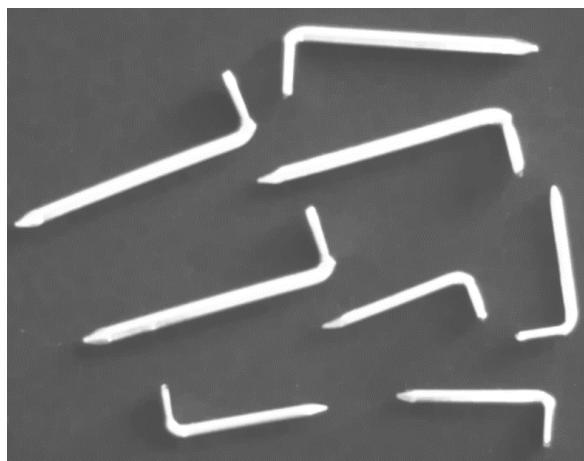
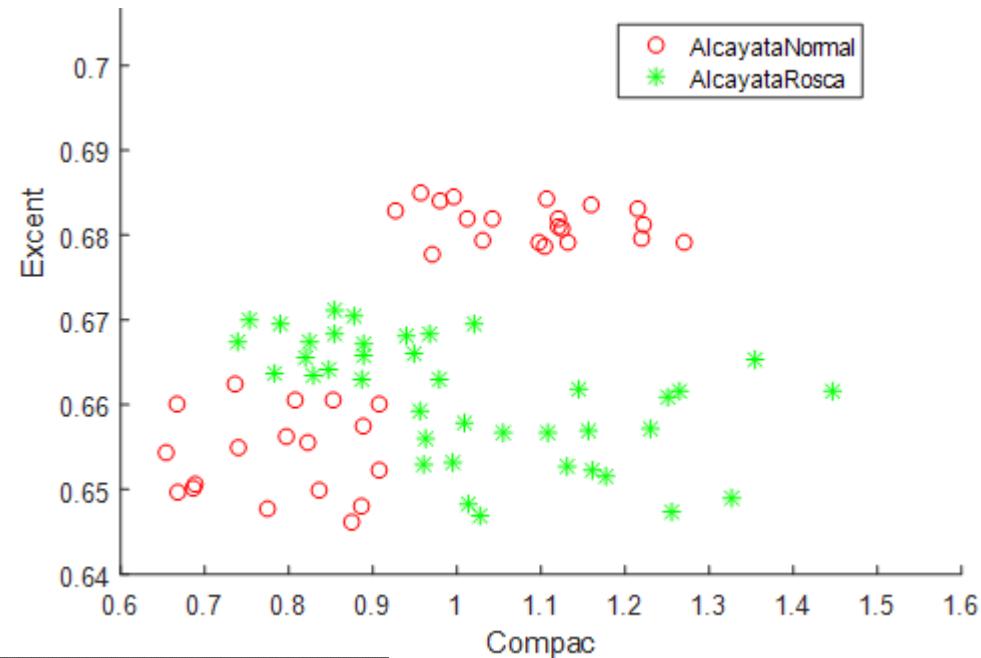
- K-vecinos más cercanos
 - Clasificador K-NN
 - Selección de instancias
 - Medidas de similitud/disimilitud
 - Regresión K-NN

RECONOCIMIENTO DE OBJETOS

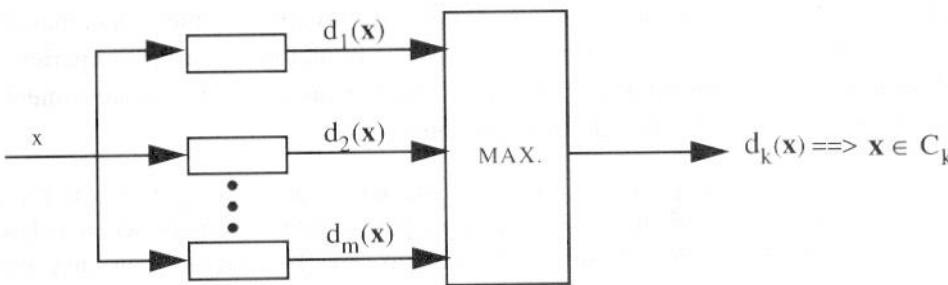
□ TECNICAS BÁSICAS DE CLASIFICACIÓN

- K-vecinos más cercanos
 - Clasificador K-NN
 - Selección de instancias
 - Medidas de similitud/disimilitud
 - Regresión K-NN

CLASIFICADOR KNN – “K-VECINOS MÁS PRÓXIMOS” (K – NEAREST NEIGHBOURS): EJEMPLO DE APLICACIÓN



➤ PLANTEAMIENTO PROBLEMA DE CLASIFICACIÓN BASADO EN TEORÍA DE DECISIÓN:



- Se diseña una función de decisión para cada clase del problema.
- Estas funciones de decisión se evalúan para una muestra descrita por un vector de atributos x .
- La muestra se asigna a la clase C_k cuya función de decisión sea mayor.

➤ CLASIFICADOR DE BAYES:

- Asigna una observación dada por x a la clase más probable.

- Función de decisión asociada a la clase j (la variable de respuesta Y tiene el valor j):

$$d_j(x) = P(Y=j | X=x) - \text{probabilidad que una muestra descrita por } x \text{ } (X=x) \text{ sea de la clase } j \text{ } (Y=j)$$

➤ CLASIFICADOR K-NN:

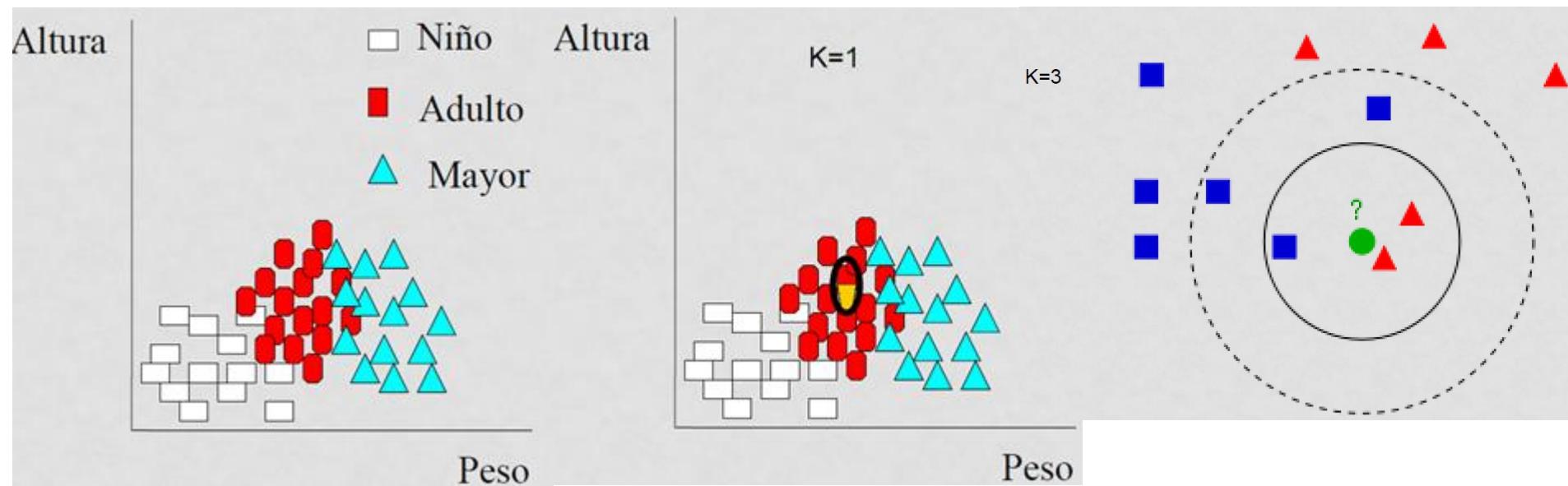
Dado un entero positivo K y una observación de test $X=x_0$:

1. El clasificador calcula N_0 : conjunto de K muestras del conjunto de entrenamiento que están más cerca de x_0 (cercanía medida en términos de distancia entre los puntos que representan las muestras en el espacio de características).
2. El clasificador estima la probabilidad condicional de una clase como la fracción de puntos de N_0 que son de la clase en cuestión. \Rightarrow Probabilidad que una muestra dada por x_0 sea de la clase j :

$$d_j(x_0) = P(Y=j | X=x_0) = \frac{1}{K} \sum_{i \in N_0} I(y_i, j) \quad \text{con} \quad I(m, n) = \begin{cases} 1 & \text{si } m = n \\ 0 & \text{si } m \neq n \end{cases}$$

□ CLASIFICADOR K-NN. Planteamiento

1. Dado un conjunto de datos de entrenamiento (observaciones o muestras descritas por los correspondientes valores de los predictores, de clase conocida) y dada una muestra de test cuya clase es desconocida, se buscan las **"K" muestras de entrenamiento más parecidas a la de test.**
2. La clase predicha para la instancia de test es la clase más numerosa de las clases a las que pertenecen las "K" muestras de entrenamiento más cercanas.



“K MÁS PARECIDOS”

- MEDIDAS DE SIMILITUD (SEMEJANZA) / DISIMILITUD (DESEMEJANZA) (*SIMILARITY / DISSIMILARITY*):

1. SIMILITUD (Ejemplo: correlación, coseno)

- Medida numérica del parecido de dos muestras de datos
- Valores más altos indican muestras más parecidas
- Generalmente su máximo valor se sitúa en 1

2. DISIMILITUD (Ejemplo: distancia)

- Medida numérica de cómo de diferentes son dos muestras de datos
- Valores más bajos indican muestras más parecidas
- El mínimo valor es generalmente 0, el valor máximo puede diferir dependiendo de la métrica.

IMPLEMENTACIÓN KNN PARA ESTABLECER EL SUBCONJUNTO K DE INSTANCIAS MÁS PARECIDAS A LA INSTANCIA DE TEST:

- MEDIDAS DE DISTANCIAS, entre las que se incluyen medidas de similitud transformadas a medidas de disimilitud (Ejemplo: 1-coseno; 1-correlación)

CARACTERÍSTICAS GENERALES

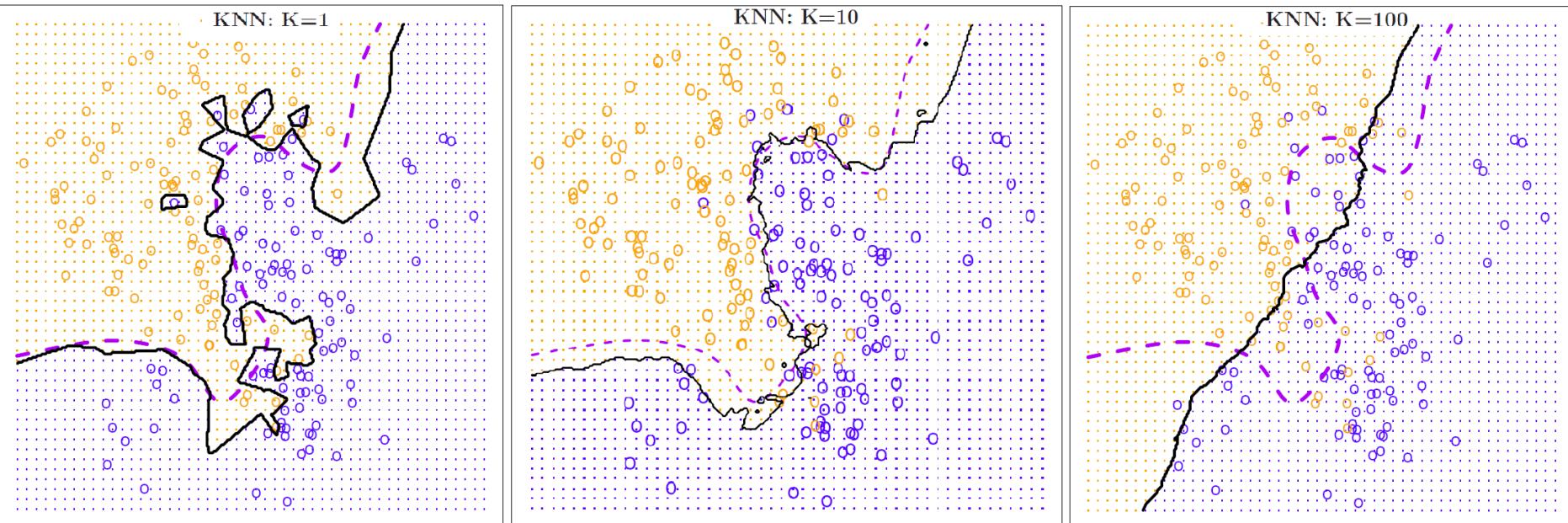
- ❑ **Algoritmo “perezoso” (*lazy*):**
 - K-NN no genera un modelo fruto del aprendizaje con datos de entrenamiento, sino *que el aprendizaje sucede en el mismo momento en el que se prueban los datos de test.*
 - Durante el entrenamiento, sólo “guardan” las instancias, no se construye ningún modelo.
 - La clasificación se hace cuando llega la instancia de test.
- ❑ **Es no paramétrico:** no se hacen suposiciones sobre la distribución que siguen los datos (como, por ejemplo, hacen clasificadores basados en distribuciones normales); asume que el mejor modelo de los datos son los propios datos.
- ❑ **Es local:** la clase de un dato depende sólo de los k vecinos mas cercanos (no se construye un modelo global).
- ❑ **Hiperparámetro del modelo:** valor de K

Observación:

- **Parámetros de un modelo:** son las variables que se estiman durante el proceso de entrenamiento con los conjuntos de datos (coeficientes en una regresión lineal, pesos en una red neuronal, vectores de soporte en una máquina de vector soporte).
- **Hiperparámetros de un modelo:** son parámetros que se configuran antes del entrenamiento del modelo y no forman parte del modelo como tal; generalmente, sus valores óptimos no se conocen a priori, para establecerlos se deben utilizar reglas genéricas, valores que han funcionado en problemas similares o ajustarlos mediante prueba y error (mediante validación cruzada o, si es demasiado costoso en tiempo, utilizando un único conjunto de validación).

SELECCIÓN DEL VALOR DE K: CONSIDERACIONES

- **K = 1:** las instancias que son ruido (o solape entre clases) tienen mucha influencia. K bajos conduce a modelos muy flexibles ⇒ riesgo de sobreaprendizaje.
- **K > 1:** se consideran mas vecinos y las instancias de ruido pierden influencia
- **K muy altos:** se pierde la idea de localidad ⇒ modelo muy poco flexible (pérdida de precisión en la clasificación). Para evitar que los vecinos lejanos tengan mucha influencia, se puede hacer que cada vecino vote de manera inversamente proporcional a la distancia (así, cuanto más lejos, tiene menos peso en la votación)
- **Ejemplo:** conjunto de datos simulados de 100 observaciones de dos clases. Se muestran las fronteras de decisión del clasificador KNN para tres valores de K (línea continua) y se comparan con la que genera el clasificador bayesiano ideal (línea discontinua).

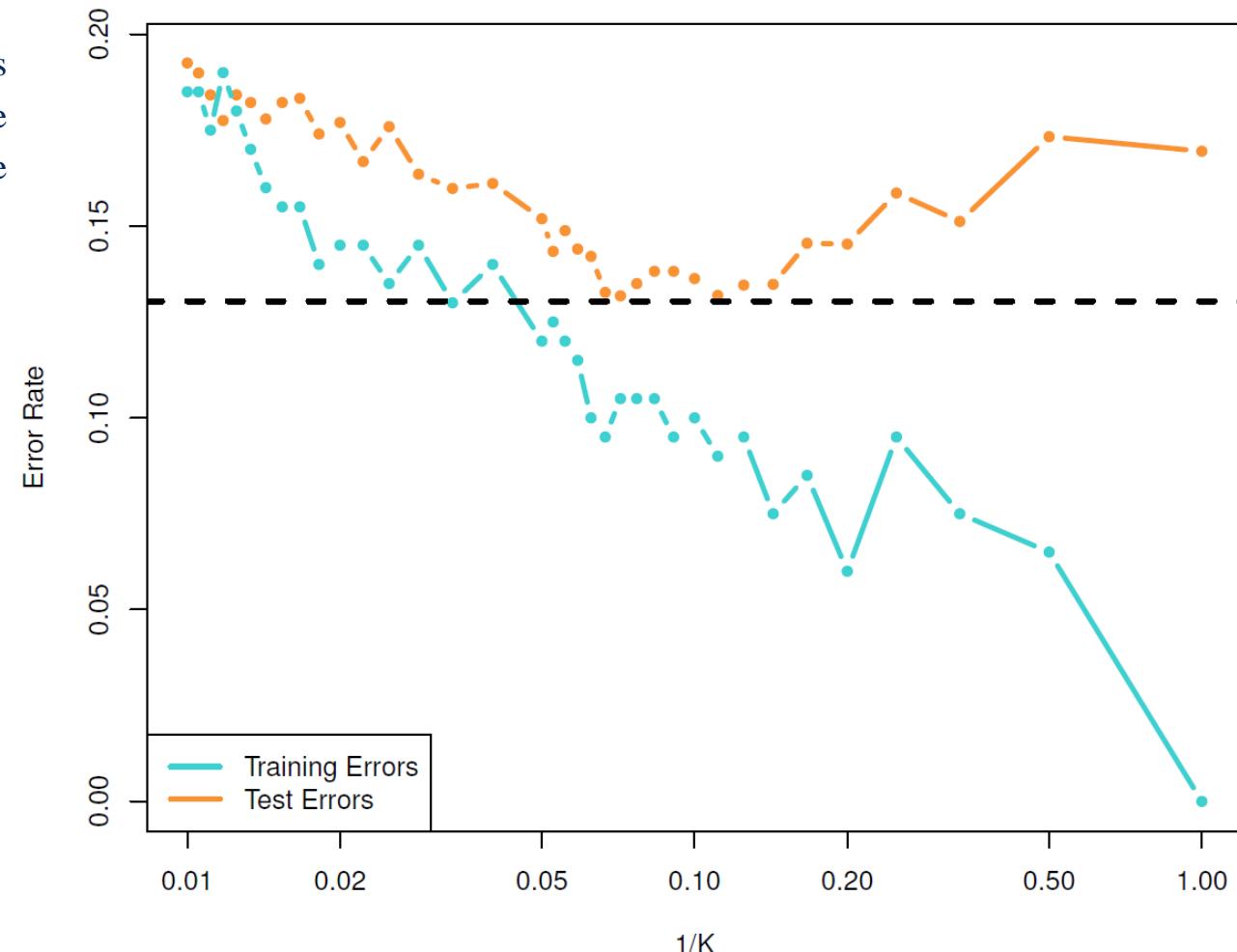


- Ejemplo anterior.** evaluación de clasificador KNN sobre un conjunto de test de 5000 observaciones, utilizando el conjunto de entrenamiento de 200 observaciones. La línea discontinua muestra el error del Clasificador de Bayes.

→ Para $K = 100$ ($1/K = 0.01$) : altos errores en ambos conjuntos de entrenamiento y test (contorno de decisión cercano al lineal).

→ A medida que disminuye K (aumenta $1/K$), el modelo es cada vez más flexible y error en el entrenamiento baja, siendo 0 para $K = 1$. Sin embargo el error de test para este valor de K es muy elevado → se produce sobreaprendizaje:

- Curva de Error en Test con forma de U característica del sobreaprendizaje.

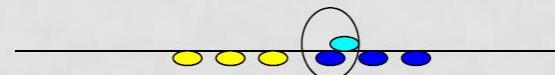


→ El mínimo error en test se produce para $K = 10$; para valores menores, el error en test tiende a aumentar → los modelos son tan flexibles que sobreajustan.

CARACTERÍSTICAS GENERALES

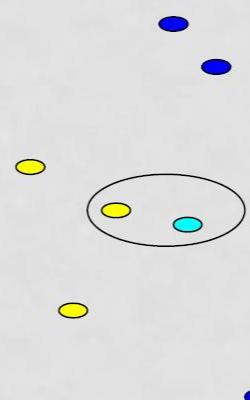
- **LIMITACIÓN:** Muy sensible a atributos irrelevantes

0 atributos irrelevantes



Con el atributo relevante, se clasifica bien

Atributo irrelevante



Atributo irrelevante

1 atributo irrelevante

Con el atributo irrelevante, se clasifica mal (las distancias cambian)

Atributo relevante

Vecino mas cercano k=1

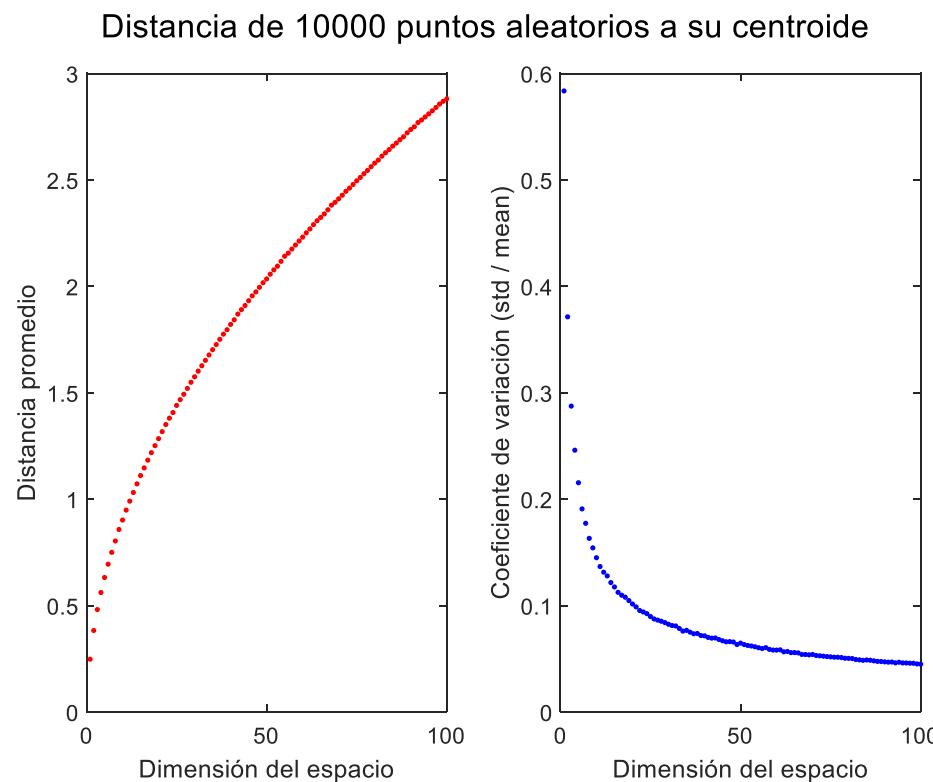
⇒ **SOLUCIÓN.** PREPROCESAMIENTO DE DATOS: aplicar técnicas de selección de atributos

CARACTERÍSTICAS GENERALES

- **LIMITACIÓN:** Muy sensible al aumento de la dimensión del problema (maldición de la dimensionalidad)

Observación: La maldición de la dimensión:

- ❖ La distancia media entre los datos aumenta con el número de dimensiones
- ❖ La variabilidad de la distancia disminuye exponencialmente con el número de dimensiones. Este es el verdadero problema:
 - Cuando hay un gran número de atributos, los datos están todos a casi la misma distancia. Es decir, no hay variabilidad entre sus distancias y es más difícil saber qué puntos están más cerca de otros.

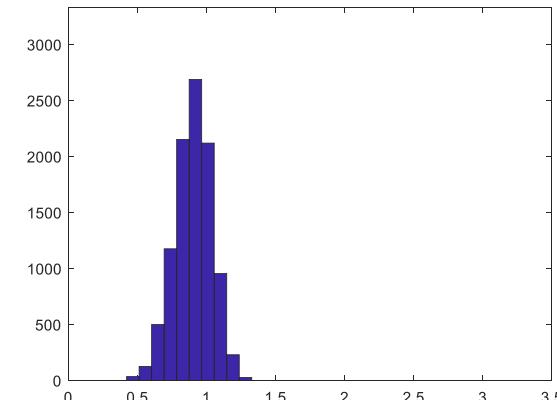
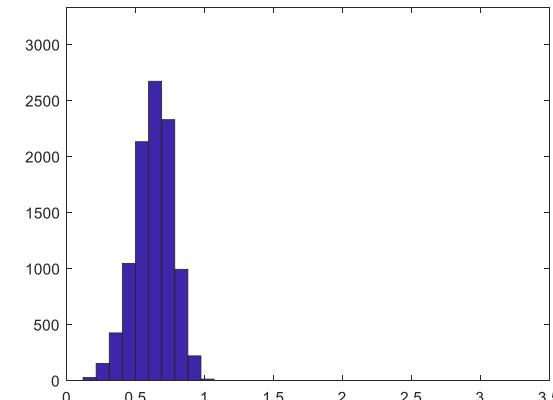
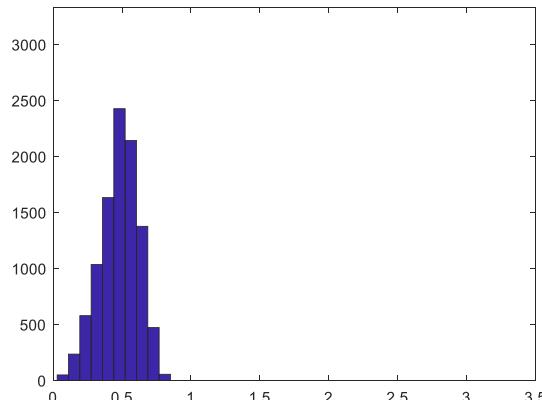


- ⇒ **SOLUCIÓN:** Aplicar selección de atributos (si se tiene la posibilidad, aumentar exponencialmente la cantidad de datos también puede ser una buena opción) para paliar trabajar con una dimensión alta.
 - A más dimensiones, más datos se necesitan para llenar el espacio. Cuando el número de dimensiones es muy alto, el espacio está casi vacío. El aumento de datos debe ser exponencial con el número de dimensiones, para contrarrestar el efecto exponencial de la pérdida de variabilidad de las distancias.

Ejemplo Maldición de la Dimensionalidad

Histograma - Distancia de 10000 puntos aleatorios a su centroide

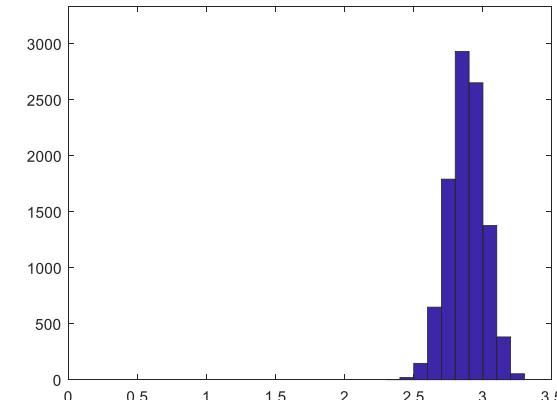
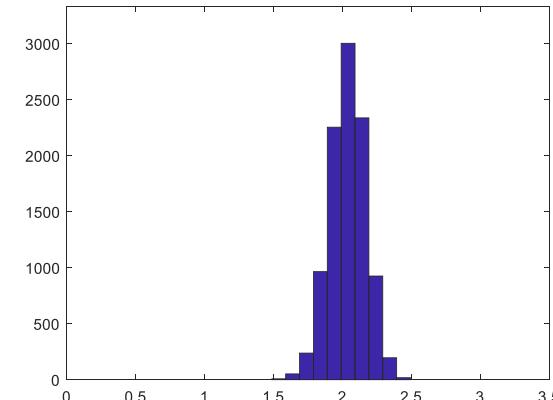
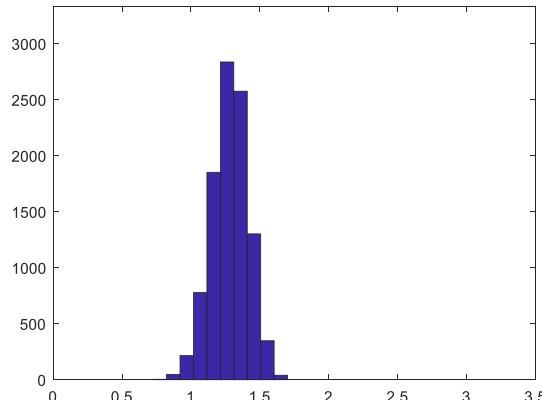
Dim: 3 ; Media: 0.48098 ; Desv: 0.1374 ; C.V.: 0.28566 Dim: 5 ; Media: 0.62906 ; Desv: 0.13678 ; C.V.: 0.21744 Dim: 10 ; Media: 0.9025 ; Desv: 0.13266 ; C.V.: 0.14699



Dim: 20 ; Media: 1.2832 ; Desv: 0.13071 ; C.V.: 0.10187

Dim: 50 ; Media: 2.0377 ; Desv: 0.12942 ; C.V.: 0.063513

Dim: 100 ; Media: 2.8816 ; Desv: 0.12916 ; C.V.: 0.044821



CARACTERÍSTICAS GENERALES

- **LIMITACIÓN:** Lento, si hay muchos datos de entrenamiento (en almacenamiento y en tiempo)

⇒ **SOLUCIÓN:**

- ✓ PREPROCESAMIENTO DE DATOS – **SELECCIÓN DE INSTANCIAS:**
 - *Eliminación de instancias superfluas*

- **LIMITACIÓN:** Muy sensible al ruido o instancias “engañosas” o de solape entre clases.

⇒ **SOLUCIÓN:**

- ✓ ALUSTE DEL HIPERPARÁMETRO DEL NÚMERO DE VECINOS
- ✓ PREPROCESAMIENTO DE DATOS – **SELECCIÓN DE INSTANCIAS:**
 - *Eliminación de instancias “engañosas”*

RECONOCIMIENTO DE OBJETOS

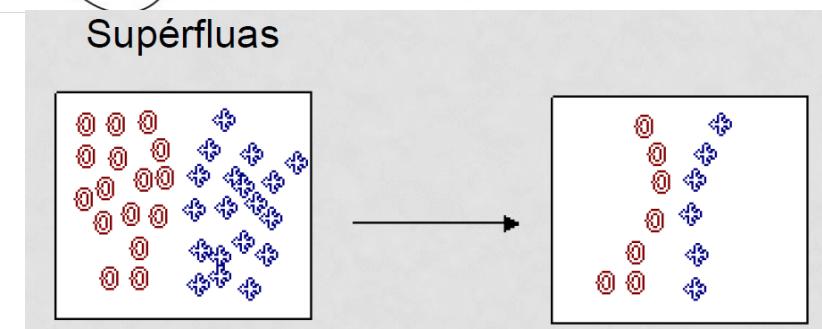
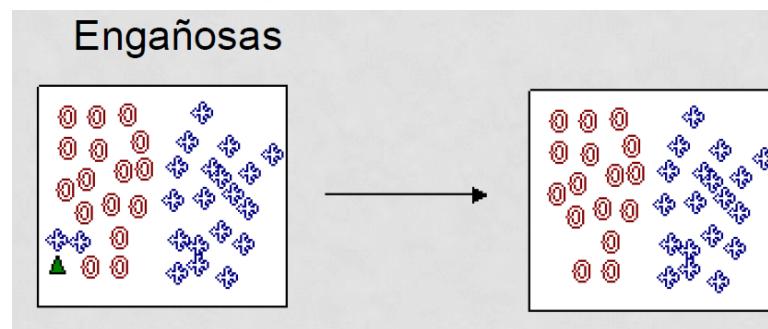
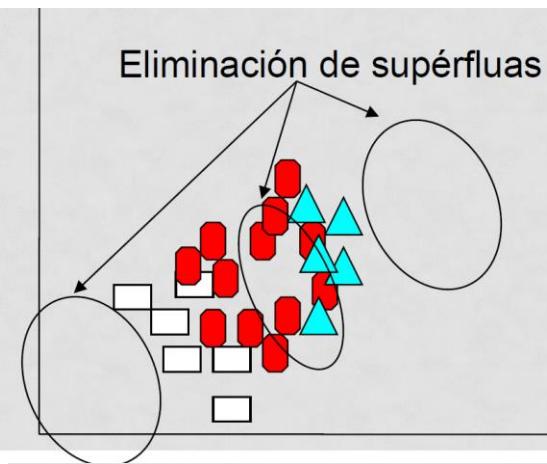
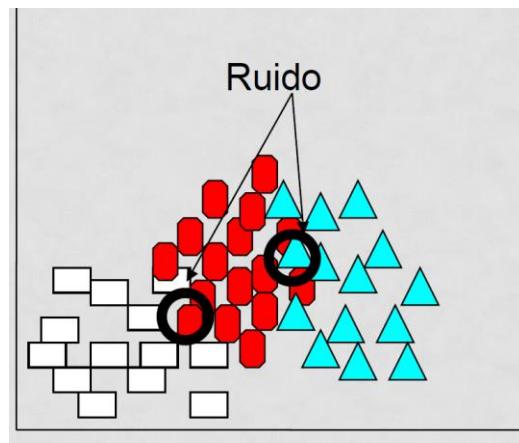
□ TECNICAS BÁSICAS DE CLASIFICACIÓN

- K-vecinos más cercanos
 - Clasificador K-NN
 - Selección de instancias
 - Medidas de similitud/disimilitud
 - Regresión K-NN

LIMITACIÓN K-NN: sensible a instancias “ruidosas” y lento en caso de conjuntos de datos muy grandes

SOLUCIÓN: SELECCIÓN DE INSTANCIAS EN UNA FASE DE PREPROCESO

- **ELIMINACIÓN DE INSTANCIAS “RUIDOSAS”:** instancias engañosas o de solape entre clases que confunden al clasificador – si se eliminan mejorará el porcentaje de aciertos en la clasificación, fundamentalmente para valores bajos de K.
- **ELIMINACIÓN DE INSTANCIAS SUPÉRFICIALES:** instancias que no aumentan el porcentaje de aciertos en la clasificación – si se eliminan se decrementará el tiempo de clasificación.

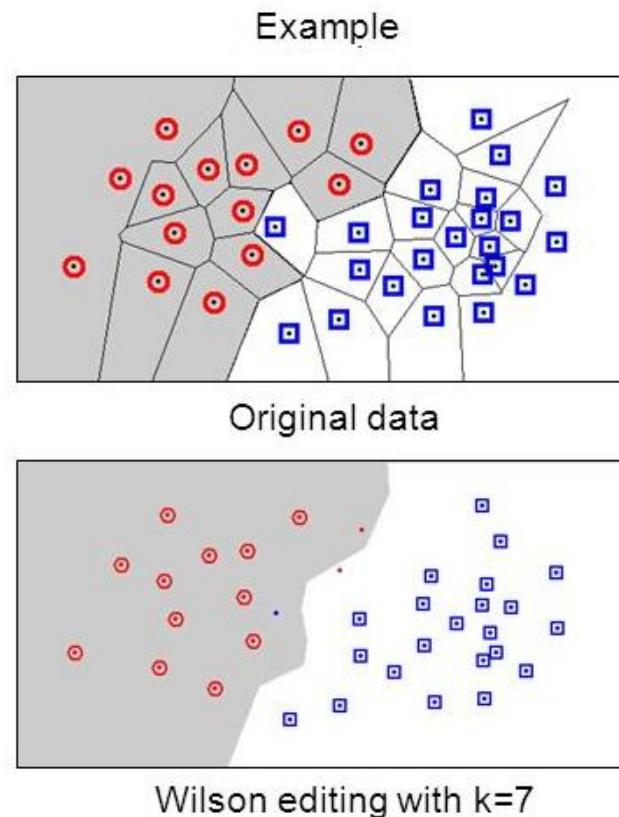


SELECCIÓN DE INSTANCIAS EN UNA FASE DE PREPROCESO

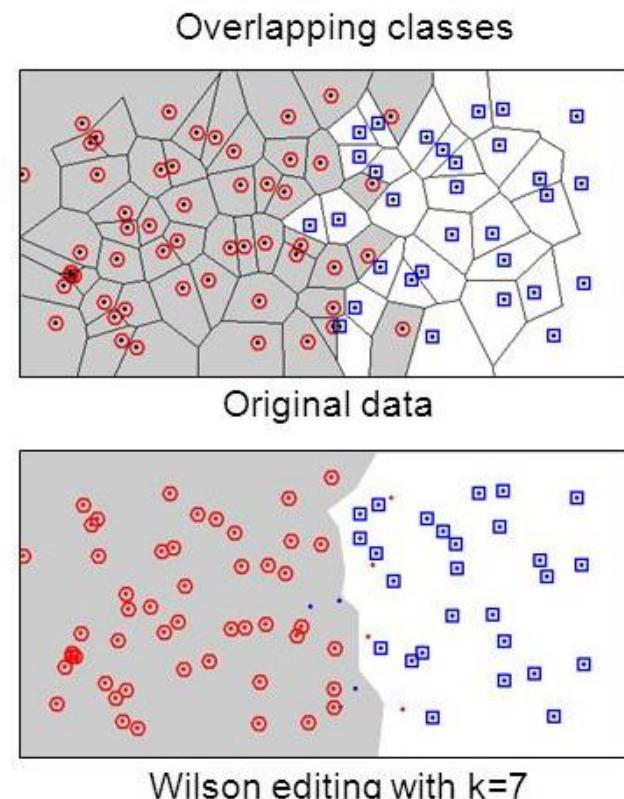
TÉCNICAS DE ELIMINACIÓN DE INSTANCIAS RUIDOSAS: ALGORITMOS DE EDICIÓN (EDITING)

- **WILSON EDITING:** Elimina una instancia x_i si es clasificada incorrectamente por sus K vecinos

(Wilson, D.L.: Asymptotic properties of nearest neighbor rules using edited data sets. IEEE Trans. on Systems, Man and Cybernetics 2, 408–421, 1972)



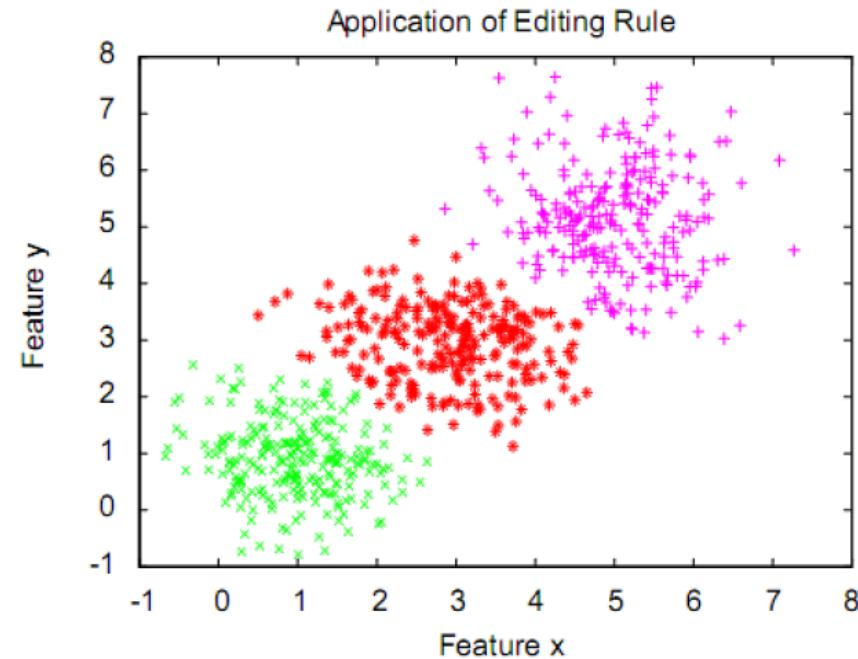
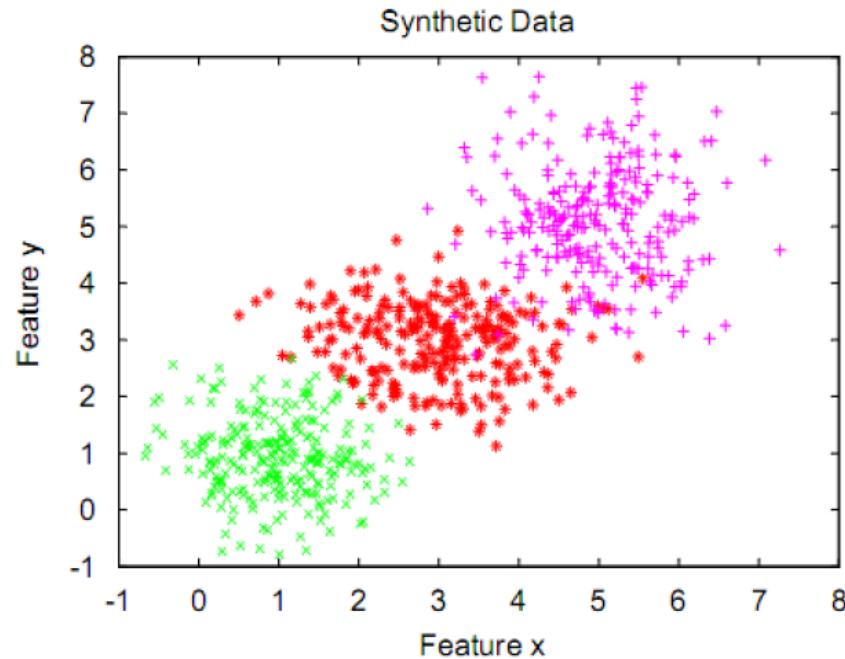
- **REPEATED WILSON EDITING:** repite *Wilson Editing* hasta que no se eliminan instancias



SELECCIÓN DE INSTANCIAS EN UNA FASE DE PREPROCESO

TÉCNICAS DE ELIMINACIÓN DE INSTANCIAS RUIDOSAS: ALGORITMOS DE EDICIÓN (EDITING)

- ❑ **WILSON EDITING:** otro ejemplo



- **Elimina pocos datos; funciona bien si no hay demasiado “ruido”**
 - Elimina bien excepciones, ruido aislado, en el interior de una clase
 - Elimina puntos en las fronteras (suaviza las fronteras)

SELECCIÓN DE INSTANCIAS EN UNA FASE DE PREPROCESO

TÉCNICAS DE ELIMINACIÓN DE INSTANCIAS SUPERFLUAS: ALGORITMOS DE CONDENSACIÓN (CONDENSED NEAREST NEIGHBOURS – CNN)

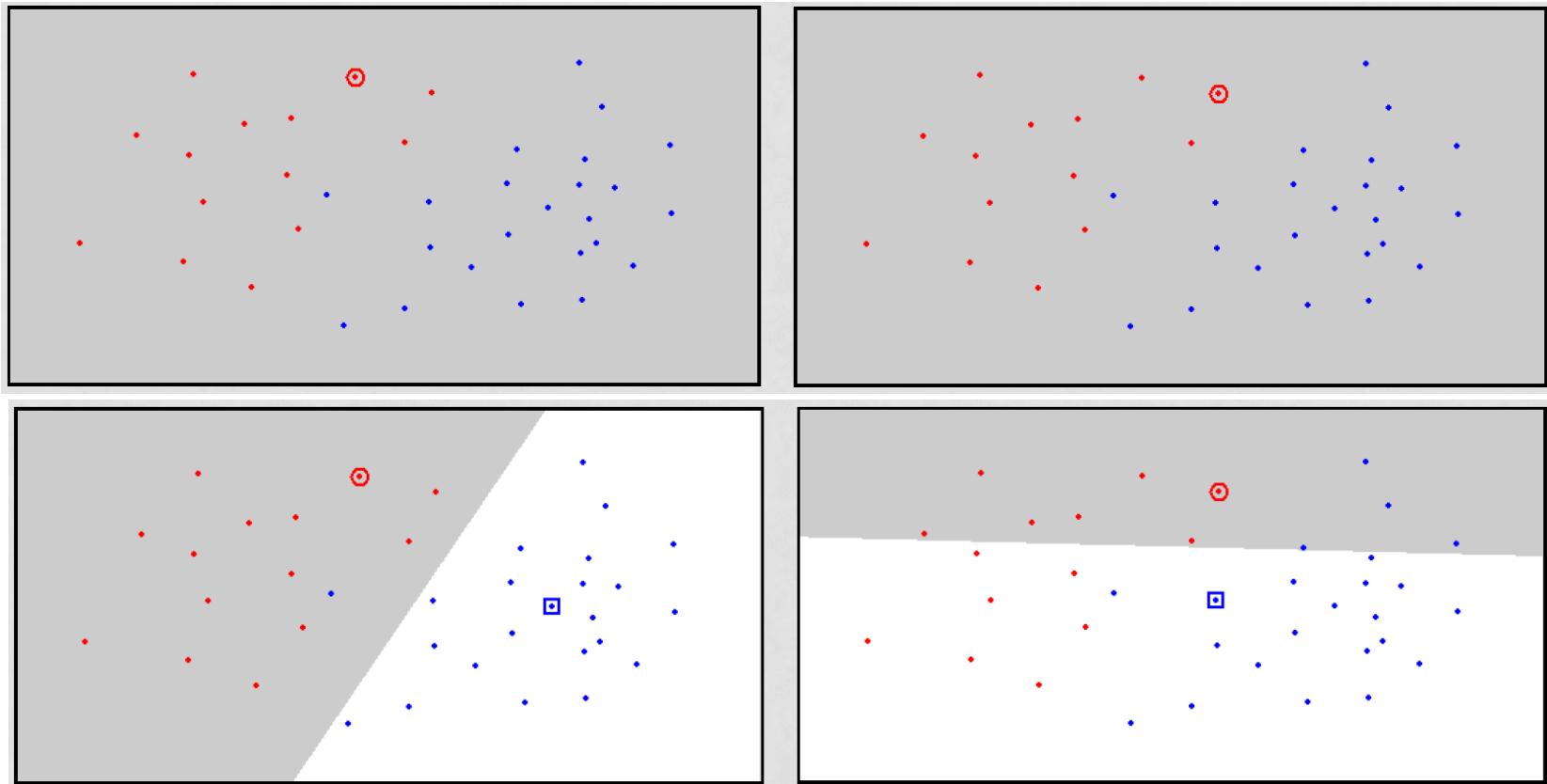
- **CNN – ALGORITMO DE HART** (Hart, P. E. "The Condensed Nearest Neighbor Rule". IEEE Transactions on Information Theory 18, 515–516, 1968)
 - **Objetivo:** reducir el número de instancias, eliminando las superfluas.
 - **Idea general:** generar un “almacén” de instancias críticas; se recorren las instancias del conjunto de datos original y, si esa instancia se clasifica bien con las que ya hay en el “almacén”, no pasa a formar parte de él – sólo se mueven al “almacén” aquellas instancias que no se clasifican bien con las instancias ya almacenadas.
 1. Inicializar el “almacén” incorporando una instancia X_1 del conjunto original (se mueve) .
 2. Incorporar al “almacén” otra instancia X_i que NO se clasifique bien con las instancias del “almacén”.
 3. Repetir 2 hasta que no se muevan más instancias del conjunto de instancias restantes de fuera al “almacén”.
 4. Utilizar para clasificar el conjunto de instancias del “almacén” en lugar del original.

SELECCIÓN DE INSTANCIAS EN UNA FASE DE PREPROCESO

TÉCNICAS DE ELIMINACIÓN DE INSTANCIAS SUPERFLUAS: ALGORITMOS DE CONDENSACIÓN (CONDENSED NEAREST NEIGHBOURS – CNN HART ALGORITHM)

→ **Limitación:** el resultado final depende de la inicialización y orden en el que toman las instancias.

Ejemplo: se ilustra el resultado con dos inicializaciones distintas

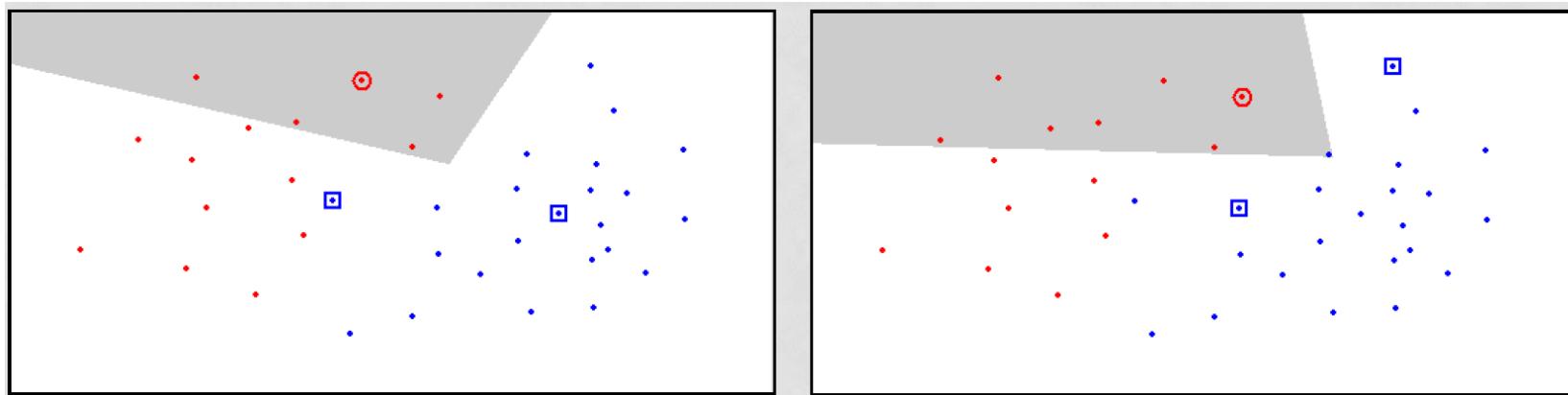
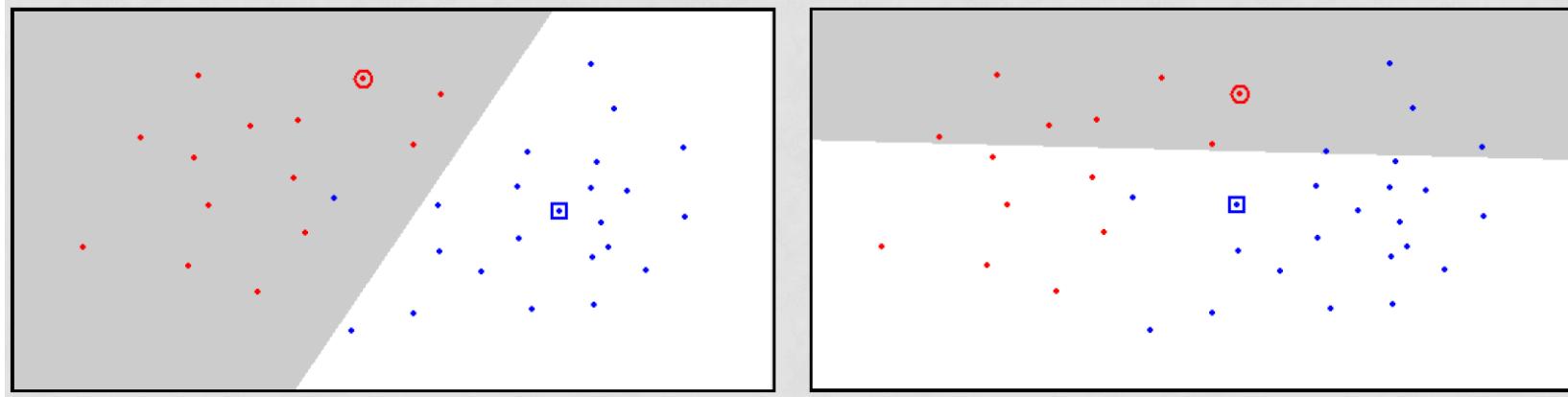


Observación: las imágenes hay que analizarlas en un orden vertical. La imagen inferior representa la incorporación de una instancia al almacén y la partición resultante del espacio de características.

SELECCIÓN DE INSTANCIAS EN UNA FASE DE PREPROCESO

TÉCNICAS DE ELIMINACIÓN DE INSTANCIAS SUPERFLUAS: ALGORITMOS DE CONDENSACIÓN (CONDENSED NEAREST NEIGHBOURS – CNN HART ALGORITHM)

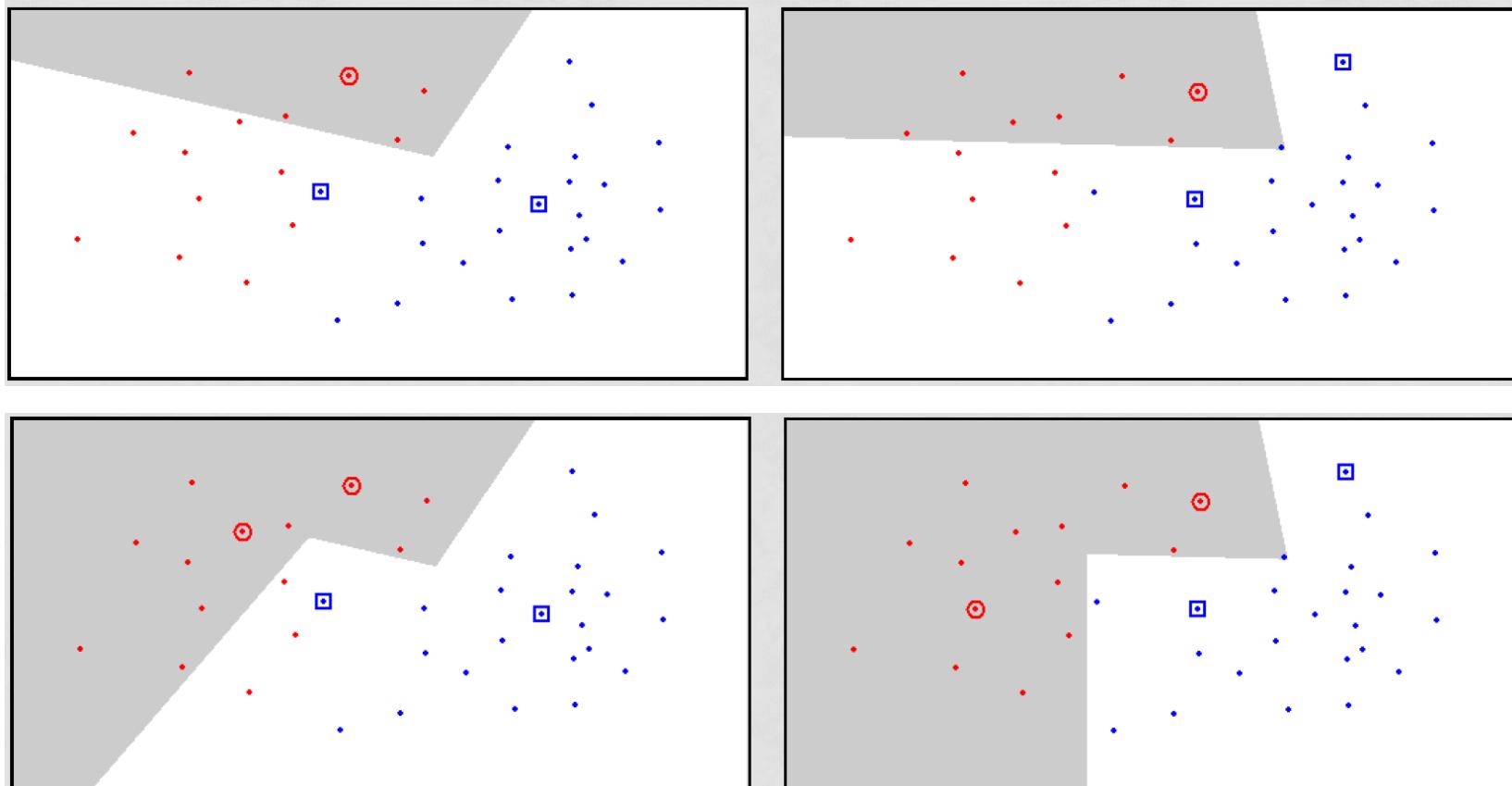
Ejemplo: las imágenes superiores son las imágenes resultantes del paso anterior



SELECCIÓN DE INSTANCIAS EN UNA FASE DE PREPROCESO

TÉCNICAS DE ELIMINACIÓN DE INSTANCIAS SUPERFLUAS: ALGORITMOS DE CONDENSACIÓN (CONDENSED NEAREST NEIGHBOURS – CNN HART ALGORITHM)

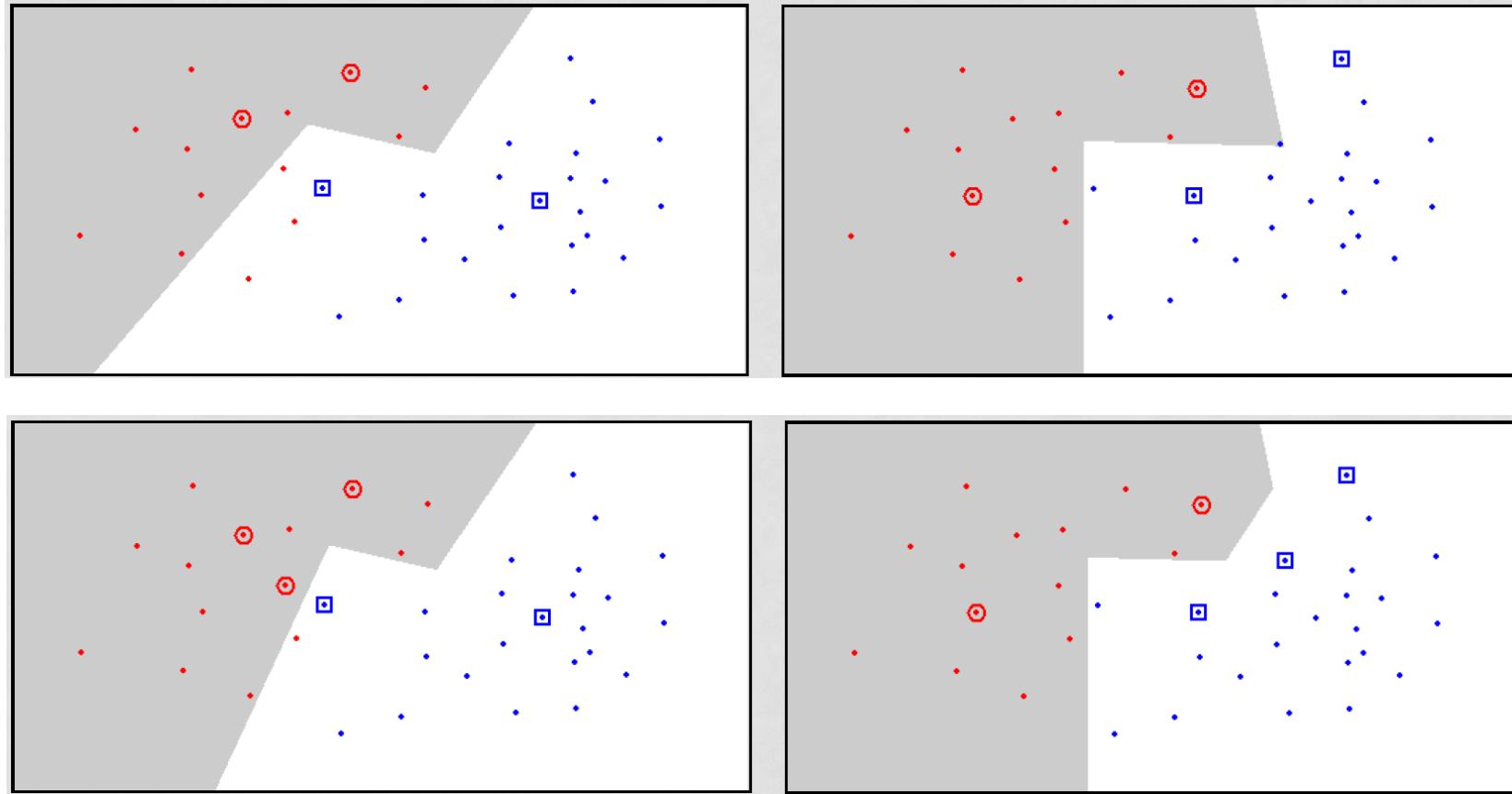
Ejemplo: las imágenes superiores son las imágenes resultantes del paso anterior



SELECCIÓN DE INSTANCIAS EN UNA FASE DE PREPROCESO

TÉCNICAS DE ELIMINACIÓN DE INSTANCIAS SUPERFLUAS: ALGORITMOS DE CONDENSACIÓN (CONDENSED NEAREST NEIGHBOURS – CNN HART ALGORITHM)

Ejemplo: las imágenes superiores son las imágenes resultantes del paso anterior

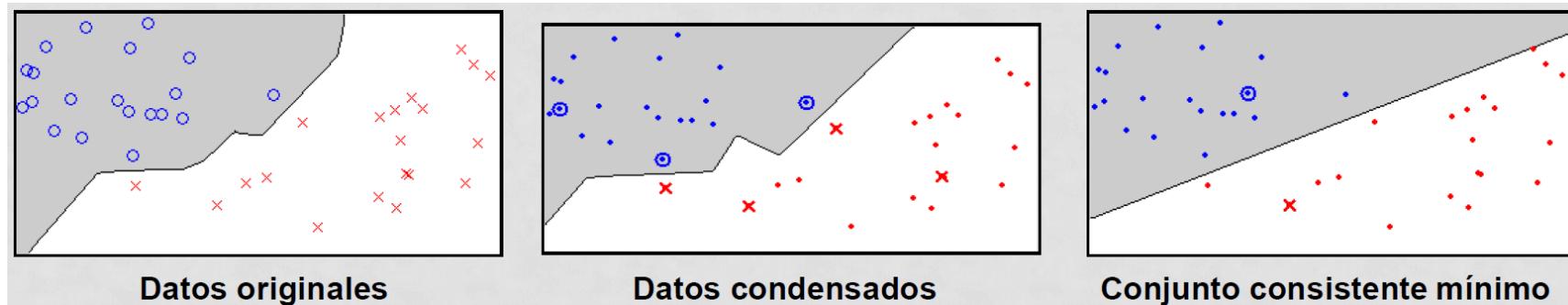


SELECCIÓN DE INSTANCIAS EN UNA FASE DE PREPROCESO

TÉCNICAS DE ELIMINACIÓN DE INSTANCIAS SUPERFLUAS: ALGORITMOS DE CONDENSACIÓN *(CONDENSED NEAREST NEIGHBOURS – CNN HART ALGORITHM)*

□ CNN – ALGORITMO DE HART - Resumiendo:

- Elimina todas aquellas instancias no críticas para la clasificación (reduce mucho la necesidad de almacenamiento).
 - Tiende a conservar aquellas instancias con ruido (puesto que son mal clasificadas por las instancias en el “almacén”).
 - Depende del orden en el que se toman las instancias
 - No hay garantía de que se genera un conjunto de datos consistente mínimo



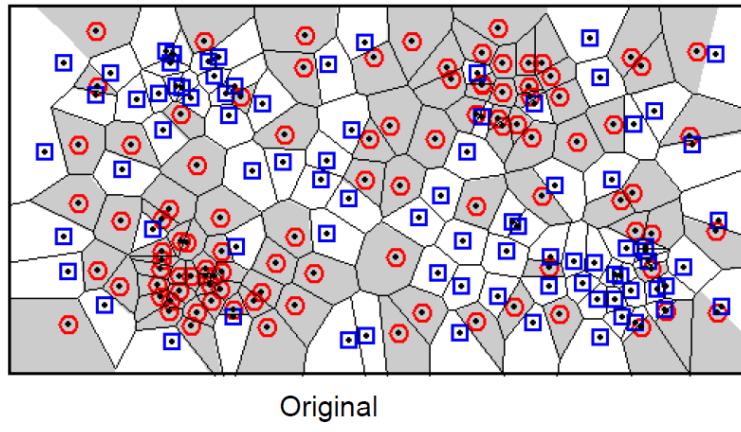
- **ALTERNATIVA A CNN: RNN – Reduced Nearest Neighbour rule:** parte de un almacén con todos los datos. El proceso consiste en ir eliminando datos del almacén: se eliminan si, al quitarlos, no provoca ningún error de clasificación. De esta forma, se almacenan las instancias críticas/necesarias para una clasificación correcta. El proceso favorece la eliminación de instancias ruidosas, puesto que no contribuyen a clasificar correctamente otras instancias.

ALGORITMOS DE EDICIÓN Y DE CONDENSACIÓN: Resumiendo

- **EDICIÓN:** elimina datos “ruidosos” de solape entre clases y suaviza las fronteras, pero mantiene la mayor parte de los datos (mejora la capacidad de generalización pero no mejora la eficiencia).
- **CONDENSACIÓN:** elimina gran cantidad de datos superfluos, contemplando en su análisis los posibles datos “ruidosos”.

ALGORITMOS HÍBRIDOS: 1.- EDITAR ; 2.- CONDENSAR

Ejemplos de algoritmos híbridos avanzados:



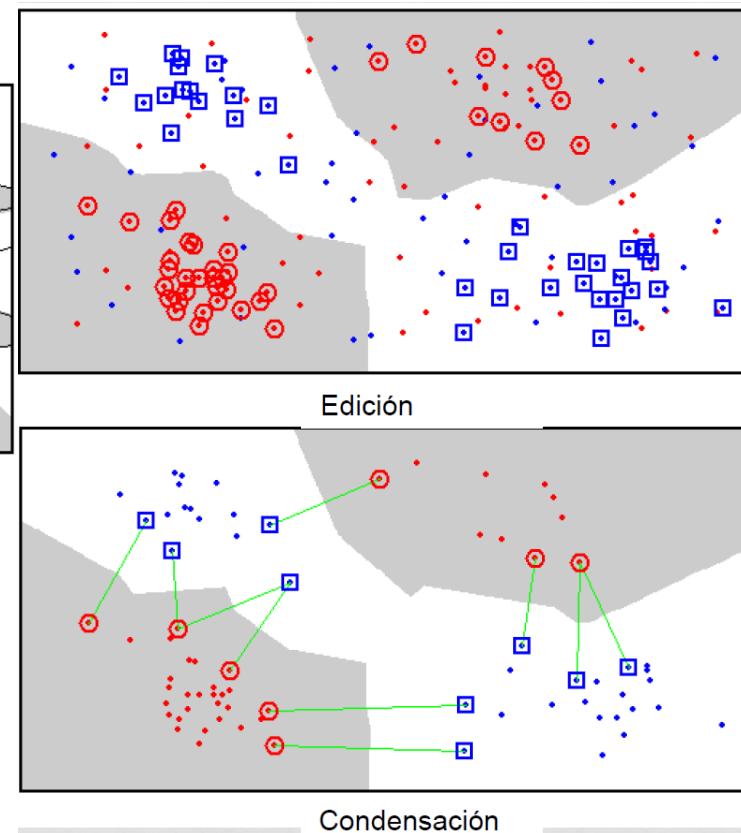
Original

RT3:

D. Randall Wilson, Tony R. Martinez: Instance Pruning Techniques. ICML 403-411, 1997.

Iterative case filtering (ICF):

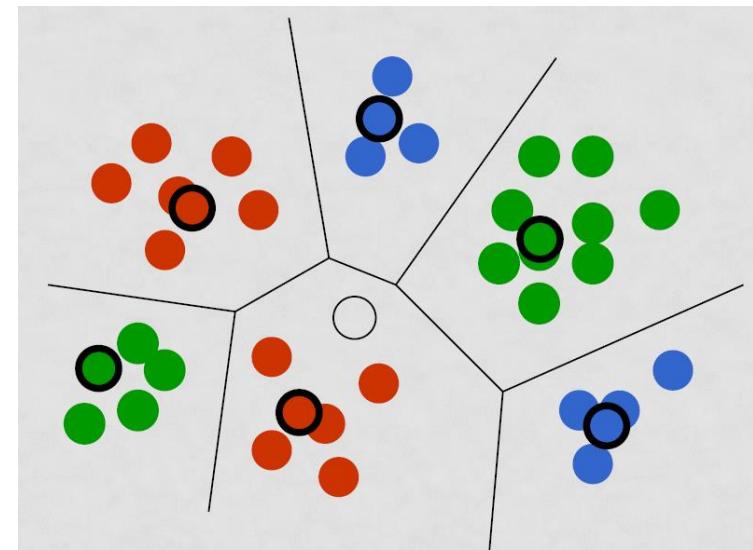
Henry Brighton, Chris Mellish: Advances in Instance Selection for Instance-Based Learning Algorithms. Data Min. Knowl. Discov. 6(2): 153-172, 2002.



Condensación

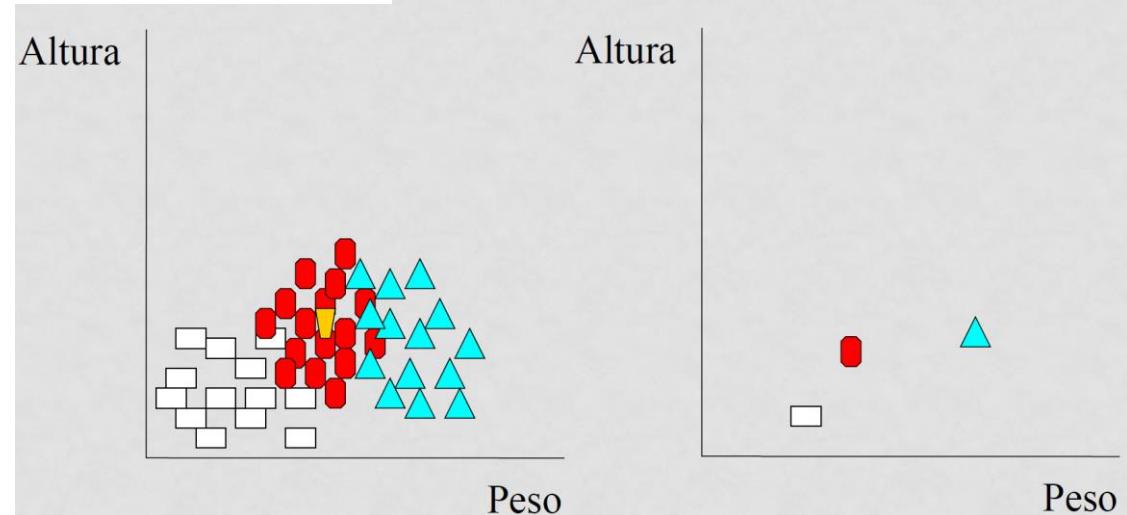
□ CLASIFICACIÓN BASADA EN PROTOTIPOS:

- Cada prototipo tiene una etiqueta
- Se clasifica según la clase del prototipo más cercano (o según sus regiones de Voronoi)
- Mejora la eficiencia en espacio (sólo se guardan unos pocos prototipos) y en tiempo (se computan muchas menos distancias cuando llega el dato de test)



→ Algoritmo de posicionamiento de prototipos: LVQ - Learning Vector Quantization

Kohonen T. Learning Vector Quantization. In: Self-Organizing Maps. Springer Series in Information Sciences, vol 30. Springer, Berlin, Heidelberg (1995 ; 2001)



RECONOCIMIENTO DE OBJETOS

□ TECNICAS BÁSICAS DE CLASIFICACIÓN

- K-vecinos más cercanos
 - Clasificador K-NN
 - Selección de instancias
 - Medidas de similitud/disimilitud
 - Regresión K-NN

CLASIFICADOR KNN – CUESTIONES DE IMPLEMENTACIÓN

PLANTEAMIENTO DE CLASIFICACIÓN:

Vector de predictores p -dimensional $\rightarrow X = (X_1, X_2, \dots, X_p)$; K clases $\rightarrow Y = \{C_1, C_2, \dots, C_K\} = \{1, 2, \dots, K\}$

Conjunto de entrenamiento $\rightarrow n$ instancias dadas por:

$$X_{train} = \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} X_{11} & \cdots & X_{1p} \\ \vdots & \ddots & \vdots \\ X_{n1} & \cdots & X_{np} \end{bmatrix}; Y_{train} = \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix}$$

Objetivo:

\rightarrow Predecir la clase de una instancia de test dada por su vector de predictores: $x_0 = (X_{01}, X_{02}, \dots, X_{0p}) \rightarrow y_0 = ???$

- **CLASIFICADOR KNN:** Decidir la clase más numerosa de las *K instancias más parecidas* del conjunto de muestras de entrenamiento a la instancia de test
- **HAY QUE CALCULAR CUÁLES SON LAS K INSTANCIAS MÁS PARECIDAS DEL CONJUNTO DE ENTRANAMIENTO:** en la práctica, cuantificando una métrica de distancia de cada instancia del conjunto de entrenamiento a la instancia de test.
 - \rightarrow MEDIDAS DE SIMILITUD (SEMEJANZA) / DISIMILITUD (DESEMEJANZA) ENTRE INSTANCIAS DESCRITAS POR UN VECTOR DE ATRIBUTOS:
 1. SIMILITUD (correlación, coseno)
 2. DISIMILITUD (medidas de distancia)

CUESTIONES ABIERTAS:

KNN: PREDICCIÓN BASADA EN MEDIR DISTANCIAS ENTRE PUNTOS p -DIMENSIONALES
(observaciones descritas por p predictores, espacio de predictores de p dimensiones)

□ **¿ CÓMO MEDIR DISTANCIAS ENTRE PUNTOS DEFINIDOS POR PREDICTORES QUE PUEDAN TENER RANGOS DE VARIACIÓN Y/O DISPERSIÓN MUY DIFERENTES ?**

- Homogeneización de predictores, para que todos ellos tengan el mismo rango de variación.
- Estandarización de predictores, para que todos ellos tengan el mismo peso en el cálculo de la distancia, independientemente de su media y dispersión.

□ **¿ CÓMO MEDIR DISTANCIAS ENTRE PUNTOS DEFINIDOS POR PREDICTORES DE DISTINTA NATURALEZA?**

- Los predictores pueden tener naturaleza cuantitativa (numérica) o cualitativa (nominal u ordinal). Hay que establecer una función de distancia adecuada para cada caso.
- Si vector de atributos mixtos:
 - Calcular distancias según la tipología de los atributos y promediar de forma ponderada.
 - Transformar todos los atributos a naturaleza numérica.

MEDIDA DE DISTANCIA ENTRE MUESTRAS p -DIMENSIONALES DEFINIDAS POR $X = (X_1, X_2, \dots, X_p)$

- ❖ NOTACIÓN: sean x_i y x_j dos muestras de un conjunto de datos $X \rightarrow$ Evaluación de la distancia entre esas muestras (distancia de dos puntos en el espacio p -dimensional definido por las variables que componen el vector de atributos de las muestras)

$$X = \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} X_{11} & \cdots & X_{1p} \\ \vdots & \ddots & \vdots \\ X_{n1} & \cdots & X_{np} \end{bmatrix} \rightarrow \begin{array}{l} x_i = (x_{i1}, x_{i2}, \dots, x_{ip}) \\ x_j = (x_{j1}, x_{j2}, \dots, x_{jp}) \end{array} \rightarrow d(x_i, x_j) = d_{ij} = ???$$

- ❖ De forma general, una función D es una métrica de distancia si, dados 3 puntos p_1, p_2 y p_3 , se verifica:

1. $D(p_1, p_2) \geq 0$ $[D(p_1, p_2) = 0 \text{ si } p_1 = p_2]$ - Definida positiva
2. $D(p_1, p_2) = D(p_2, p_1)$ - Simetría
3. $D(p_1, p_3) \leq D(p_1, p_2) + D(p_2, p_3)$ - Desigualdad triangular

- ❖ A CONTINUACIÓN, DEFINICIÓN DE MÉTRICAS DE DISTANCIA EN EL CASO DE TRABAJAR CON VECTORES DE ATRIBUTOS COMPUESTOS POR:
 - Atributos de naturaleza cuantitativa: numéricos
 - Atributos de naturaleza cualitativa: nominales o categóricos (binarios o de más de dos estados)
 - Atributos de naturaleza cualitativa: ordinales

$$X = (X_1, X_2, \dots, X_p) \rightarrow X = \begin{bmatrix} X_1 \\ \vdots \\ X_n \end{bmatrix} = \begin{bmatrix} X_{11} & \cdots & X_{1p} \\ \vdots & \ddots & \vdots \\ X_{n1} & \cdots & X_{np} \end{bmatrix} \rightarrow \begin{array}{l} x_i = (x_{i1}, x_{i2}, \dots, x_{ip}) \\ x_j = (x_{j1}, x_{j2}, \dots, x_{jp}) \end{array} \rightarrow d(x_i, x_j) = d_{ij} = ???$$

□ **ATRIBUTOS NUMÉRICOS: MÉTRICA DE DISTANCIA**

DISTANCIA DE MINKOWSKI (Norma L_h): $d(x_i, x_j) = d_{ij} = \sqrt[h]{(|X_{i1} - X_{j1}|^h + |X_{i2} - X_{j2}|^h + \dots + |X_{ip} - X_{jp}|^h)}$

- h , entero positivo. Casos particulares:

→ **DISTANCIA DE MANHATAN (CITY BLOCK, Norma L_1):** $h = 1$

$$d(x_i, x_j) = d_{ij} = |X_{i1} - X_{j1}| + |X_{i2} - X_{j2}| + \dots + |X_{ip} - X_{jp}|$$

- **DISTANCIA HAMMING** - caso particular cuando los atributos son binarios (número de bits diferentes en dos vectores binarios)

→ **DISTANCIA EUCLIDEA (Norma L_2):** $h = 2$

$$d(x_i, x_j) = d_{ij} = \sqrt{(|X_{i1} - X_{j1}|^2 + |X_{i2} - X_{j2}|^2 + \dots + |X_{ip} - X_{jp}|^2)}$$

→ **DISTANCIA DE CHEBYSHEV (Norma suprema L_∞ , expresa la diferencia máxima en cualquiera de las componentes o atributos del vector):** $h \rightarrow \infty$

$$d(x_i, x_j) = d_{ij} = \lim_{h \rightarrow \infty} \left(\sum_{f=1}^p |X_{if} - X_{jf}|^h \right)^{1/h} = \max_f |X_{if} - X_{jf}| = \max(|X_{i1} - X_{j1}|, |X_{i2} - X_{j2}|, \dots, |X_{ip} - X_{jp}|)$$

ATRIBUTOS NUMÉRICOS: MÉTRICAS DE DISTANCIA - EJEMPLO

→ **DISTANCIA DE MANHATAN:** $d(x_i, x_j) = |X_{i1} - X_{j1}| + |X_{i2} - X_{j2}| + \dots + |X_{ip} - X_{jp}|$

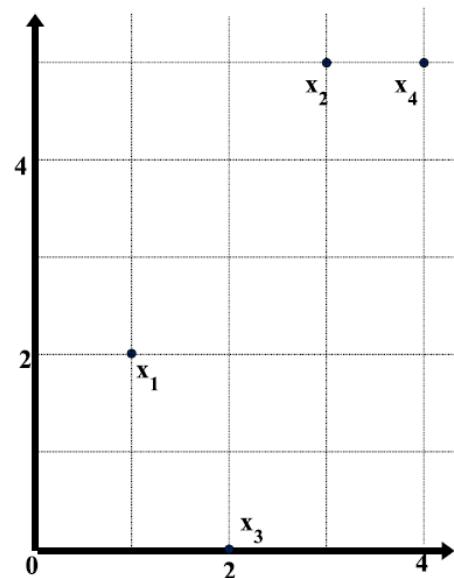
→ **DISTANCIA EUCLIDEA:** $d(x_i, x_j) = \sqrt{(|X_{i1} - X_{j1}|^2 + |X_{i2} - X_{j2}|^2 + \dots + |X_{ip} - X_{jp}|^2)}$

→ **DISTANCIA DE CHEBYSHEV**
(SUPREMUM):

$$d(x_i, x_j) = \max_f |X_{if} - X_{jf}|$$

point	attribute 1	attribute 2
x1	1	2
x2	3	5
x3	2	0
x4	4	5

Ejemplo: 4 puntos descritos por dos atributos



Manhattan (L_1)

L	x1	x2	x3	x4
x1	0			
x2	5	0		
x3	3	6	0	
x4	6	1	7	0

Euclidean (L_2)

L2	x1	x2	x3	x4
x1	0			
x2	3.61	0		
x3	2.24	5.1	0	
x4	4.24	1	5.39	0

Supremum

L_∞	x1	x2	x3	x4
x1	0			
x2	3	0		
x3	2	5	0	
x4	3	1	5	0

□ **ATRIBUTOS NOMINALES O CATEGÓRICOS: MEDIDA DE DISTANCIA**

Dado un espacio definido por $X = (X_1, X_2, \dots, X_p)$ con X_f nominal $\forall f \text{ de } 1 \text{ a } p$. Determinar la distancia entre dos puntos de este espacio: $x_i = (x_{i1}, x_{i2}, \dots, x_{ip})$; $x_j = (x_{j1}, x_{j2}, \dots, x_{jp}) \rightarrow d(x_i, x_j) = d_{ij} = ???$

1. **ATRIBUTOS BINARIOS:** $X = (X_1, X_2, \dots, X_p)$ con $X_f \in \{0,1\}$, atributos binarios $\forall f \text{ de } 1 \text{ a } p$

→ Distancia para atributos simétricos:

$$d_{ij} = \frac{\text{Número de discrepancias}}{\text{Número de atributos}} = \frac{M_{10} + M_{01}}{M_{11} + M_{10} + M_{01} + M_{00}}$$

→ Distancia para atributos asimétricos

$$d(x_i, x_j) = 1 - \text{SimJaccard}(x_i, x_j) = \frac{\text{Número de discrepancias}}{\text{Número de atributos sin coincidencias en 00}} = \frac{M_{10} + M_{01}}{M_{11} + M_{10} + M_{01}}$$

Tabla de contingencia para datos binarios

		x_j		Total
		1	0	
x_i	1	M_{11}	M_{10}	$M_{11} + M_{10}$
	0	M_{01}	M_{00}	$M_{01} + M_{00}$
Total		$M_{11} + M_{01}$	$M_{10} + M_{00}$	p

Coeficiente de Jaccard (medida de similitud):

$$\text{SimJaccard}(x_i, x_j) = \frac{\text{Número de coincidencias 11}}{\text{Número de atributos sin coincidencias en 00}} = \frac{M_{11}}{M_{11} + M_{10} + M_{01}}$$

□ **ATRIBUTOS NOMINALES: MEDIDA DE DISTANCIA**

Dado un espacio definido por $X = (X_1, X_2, \dots, X_p)$ con X_f nominal $\forall f$ de 1 a p . Determinar la distancia entre dos puntos de este espacio: $x_i = (x_{i1}, x_{i2}, \dots, x_{ip})$; $x_j = (x_{j1}, x_{j2}, \dots, x_{jp}) \rightarrow d(x_i, x_j) = d_{ij} = ???$

2. **ATRIBUTOS NOMINALES DE MÁS DE DOS ESTADOS:**

$$\rightarrow \textbf{Opción 1: } d(x_i, x_j) = \frac{\text{Nº de discrepancias}}{\text{Nº de atributos}} = \frac{\text{Nº de atributos} - \text{Nº de coincidencias}}{\text{Número de atributos}} = \frac{p - m}{p}$$

→ **Opción 2:** transformar los datos a forma binaria, creando un atributo binario por cada uno de los estados categóricos posibles de cada atributo

□ **ATRIBUTOS ORDINALES:** se transforman a naturaleza numérica para computar la distancia entre muestras.

Suponiendo X_f un atributo ordinal con M_f valores.

1. Establecer una escala de valores numérica para cada atributo, un ranking: $\text{Ranking}_X_f = \{1, 2, \dots, M_f\}$
2. Reemplazar los valores de las muestras de cada atributo por su ranking: $X_{if} \rightarrow R_{if} \in \{1, 2, \dots, M_f\}$
3. Normalizar el rango valores a $[0, 1]$: $R_{if} \rightarrow X_{if}^* = \frac{R_{if}-1}{M_f-1} \in [0, 1]$

MEDIDA DE DISTANCIA ENTRE MUESTRAS DE DATOS DEFINIDOS POR ATRIBUTOS MIXTOS

- Un conjunto de datos puede estar definido por atributos de tipo numérico (de intervalo y de razón), ordinal, nominal binario (simétrico o asimétrico) y nominal de más de dos estados.
- Para medir la distancia entre dos puntos en ese espacio de atributos mixtos – Posibilidades:
 - Tratar los datos de forma conjunta, transformar todos los atributos a naturaleza numérica y normalizar/estandarizar sus datos.
 - Tratar los datos en su naturaleza original y promediar de forma ponderada las distancias de cada atributo individual o las distancias evaluadas para cada conjuntos de atributos de la misma tipología.

Ejemplo de tratamiento de tratamiento de distancias entre atributos de forma individual:

$$x_i = (x_{i1}, x_{i2}, \dots, x_{ip}); x_j = (x_{j1}, x_{j2}, \dots, x_{jp}) \rightarrow d(x_i, x_j) = \frac{\sum_{f=1}^p \delta_f * d(x_{if}, x_{jf})}{\sum_{f=1}^p \delta_f}$$

- Si X_f es nominal: $d(x_{if}, x_{jf}) = 0$ si $x_{if} = x_{jf}$ y $d(x_{if}, x_{jf}) = 1$ si $x_{if} \neq x_{jf}$
- Si X_f es numérico: evaluar distancia estandarizada (es habitual considerar la distancia Euclidea dividida por la desviación estándar del atributo)
- Si x_f es ordinal (o numérico de razón): transformar a numérico y evaluar distancia estandarizada.
- δ_f : pesos de ponderación a la distancia de cada atributo

MEDIDA DE SIMILITUD POR COSENO

- Hasta ahora, se han medido distancias entre dos observaciones de datos que se han tratado como puntos en el espacio de atributos (MEDIDA DE DISIMILITUD: distancia entre puntos)
 - En este apartado se introduce una medida de similitud basada en tratar las observaciones como vectores en ese espacio de atributos: MEDIDA DE SIMILITUD POR COSENO (medida basada en el ángulo que forman esos vectores)
- MEDIDA DE SIMILITUD POR COSENO DEL ÁNGULO QUE FORMAN LOS VECTORES ASOCIADOS A DOS OBSERVACIONES DE DATOS**

$$x_i = (x_{i1}, x_{i2}, \dots, x_{ip}); \quad x_j = (x_{j1}, x_{j2}, \dots, x_{jp}) \rightarrow \cos(x_i, x_j) = (x_i \cdot x_j) / (\|x_i\| * \|x_j\|)$$

donde:

$$\|x_i\| = \sqrt{(x_i \cdot x_i)} ; \|x_j\| = \sqrt{(x_j \cdot x_j)}$$

- \bullet : denota el producto escalar
- $\| \ |$: denota el módulo o longitud del vector

- Ejemplo de aplicación: vectores de atributos definidos en términos de frecuencia de aparición o conteo de determinada información o magnitud

MEDIDA DE SIMILITUD POR COSENO: EJEMPLO

- Un documento puede ser descrito por miles de atributos, cada uno registrando la frecuencia de aparición en el documento de una determinada palabra (palabras clave) o frase

	team	coach	baseball	soccer	penalty	score	win	loss
Doc1	5	0	0	2	0	0	2	0
Doc2	3	0	2	1	0	0	3	0
Doc3	0	7	0	1	0	0	3	0
Doc4	0	1	0	1	2	2	0	3

→ Encontrar la similitud entre los documentos 1 y 2 $\cos(x_i, x_j) = (x_i \cdot x_j) / (\|x_i\| * \|x_j\|)$

$$d_1 = (5, 0, 3, 0, 2, 0, 0, 2, 0, 0)$$

$$d_2 = (3, 0, 2, 0, 1, 1, 0, 1, 0, 1)$$

$$d_1 \cdot d_2 = 5*3+0*0+3*2+0*0+2*1+0*1+0*1+2*1+0*0+0*1 = 25$$

$$\|d_1\| = (5^2 + 0^2 + 3^2 + 0^2 + 2^2 + 0^2 + 0^2 + 1^2 + 1^2 + 0^2)^{0.5} = (42)^{0.5} = 6.481$$

$$\|d_2\| = (3^2 + 0^2 + 2^2 + 0^2 + 1^2 + 1^2 + 0^2 + 1^2 + 0^2 + 1^2)^{0.5} = (17)^{0.5} = 4.12$$

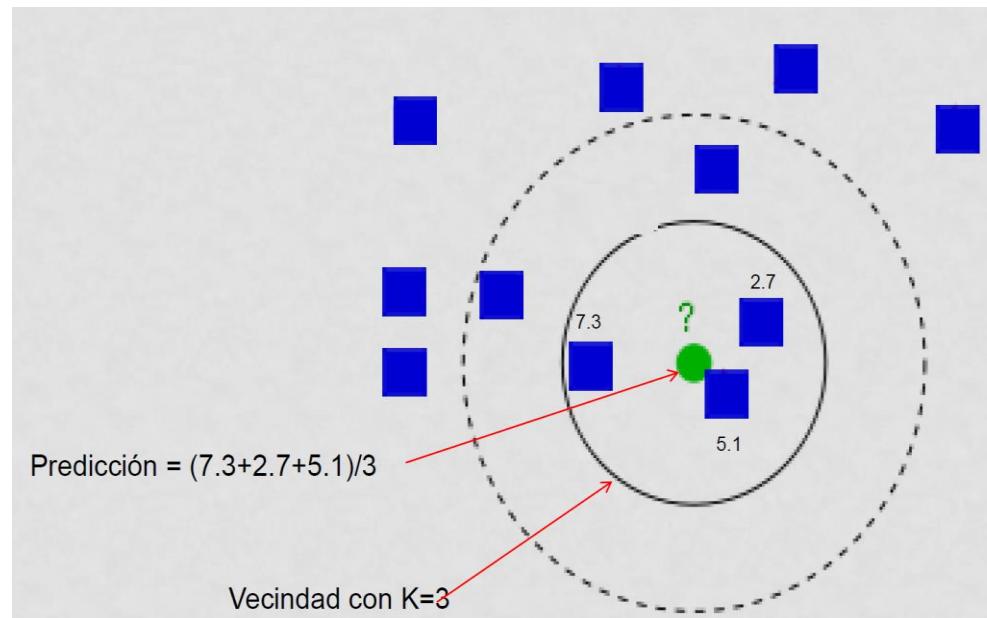
$$\cos(d_1, d_2) = 0.94$$

RECONOCIMIENTO DE OBJETOS

□ TECNICAS BÁSICAS DE CLASIFICACIÓN

- K-vecinos más cercanos
 - Clasificador K-NN
 - Selección de instancias
 - Medidas de similitud/disimilitud
 - Regresión K-NN

- ❑ **KNN para clasificación:** predice la clase de un instancia como la clase mayoritaria de entre los k vecinos mas cercanos de entre los datos de entrenamiento (la respuesta o salida del problema es de naturaleza cualitativa o discreta, la clase se refiere a uno de sus posibles valores).
- ❑ **KNN para regresión:** predice la respuesta o salida (de naturaleza numérica cuantitativa o continua) de una instancia como la media de las respuestas de los k vecinos mas cercanos de los datos de entrenamiento.



□ PROBLEMAS DE REGRESIÓN EN APRENDIZAJE AUTOMÁTICO

- *Predictores, o variables de entrada:* $X = (X_1, X_2, \dots, X_p)$
- *Repuesta o variable de salida:* Y

Problema de regresión: problemas con una respuesta Y cuantitativa..

Dado un conjunto de datos compuesto por n *observaciones*: $\{ (x_1, y_1), (x_2, y_2), \dots, (x_n, y_n) \}$

$x_i = (X_{i1}, X_{i2}, \dots, X_{ip}) \in Y_i$: *valores de las variable de entrada y salida para la observación i*

X_{ij} : *valor de la variable X_j para la observación i con $i = 1, \dots, n$ y $j = 1, \dots, p$*

$$X = \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} X_{11} & \cdots & X_{1p} \\ \vdots & \ddots & \vdots \\ X_{n1} & \cdots & X_{np} \end{bmatrix} \quad Y = \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix}$$

y asumiendo existe una función f que relaciona las variables entrada-salida $X-Y$: $Y = f(X) + \varepsilon$

- **OBJETIVO:** encontrar una estimación de la función f que establece la relación $X-Y$ a partir de las observaciones entrada-salida disponibles.

$$\hat{Y} = \hat{f}(X)$$

- \hat{f} representa nuestra estimación sobre f ,
- \hat{Y} representa la predicción resultante de Y

REGRESIÓN KNN:

- Método no paramétrico, no se hace ninguna suposición explícita sobre la forma funcional de f , se busca la estimación de f que se acerque lo más posible a los puntos del conjunto de datos de entrenamiento.

Dado un entero positivo K y una observación de test $X=x_0$:

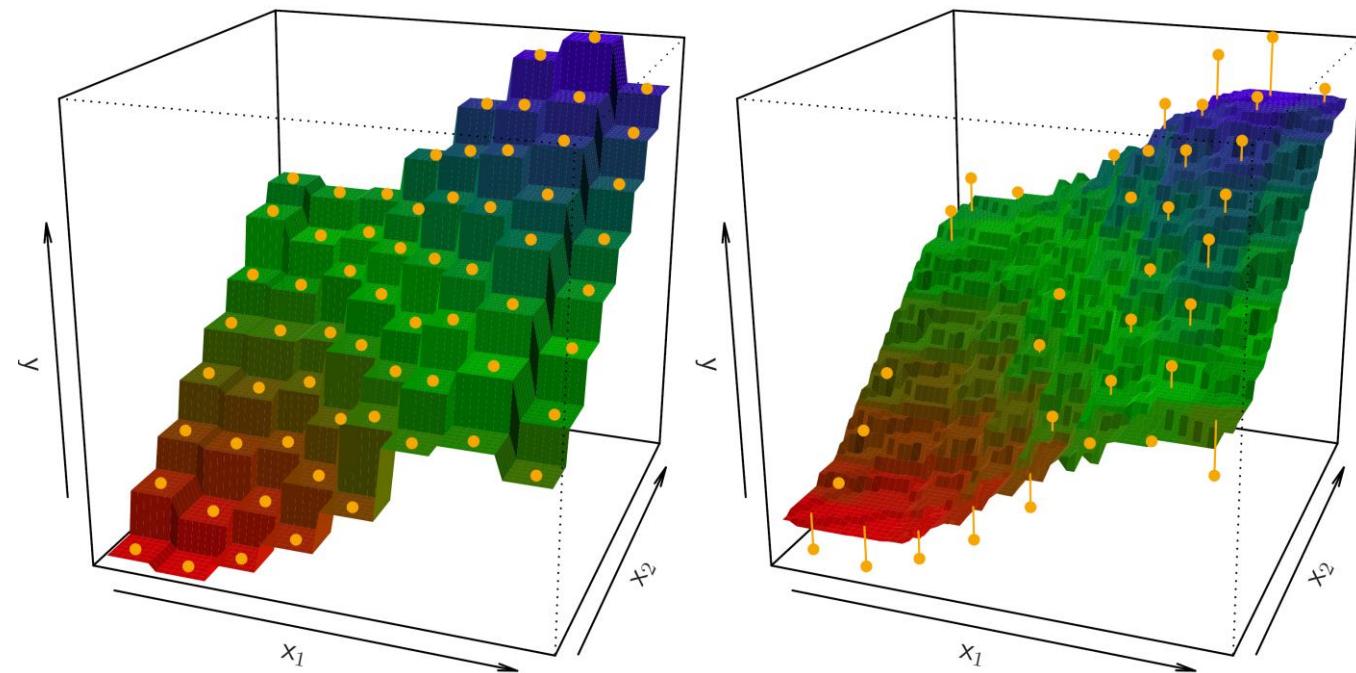
1. La regresión KNN calcula N_0 : conjunto de K muestras del conjunto de entrenamiento que están más cerca de x_0 .
2. Estima la salida de la observación dada por x_0 mediante el promedio de las salidas de las muestras N_0

$$\hat{y} = \hat{f}(X = x_0) = \frac{1}{K} \sum_{x_i \in N_0} y_i$$

- ❖ Para que las instancias más lejanas tengan menos importancia, se puede hacer una media ponderada por $1/d$

REGRESIÓN KNN. Ejemplo: 64 muestras de dos predictores y salida (puntos naranjas).

- La figura de la izquierda muestra el ajuste de un KNN con $K = 1$ (ajuste abrupto dado por funciones escalón). La figura de la derecha muestra el ajuste de un KNN con $K=9$ (ajuste más suave a los datos).



- ❖ **Valores pequeños de K** proporcionan modelos muy flexibles, con gran capacidad de ajuste a los datos de entrenamiento → Riesgo de sobreaprendizaje: la predicción en una región dada depende exclusivamente de un número pequeño de observación (tan solo una en el caso de $K=1$).
- ❖ **Valores más elevados de K** proporcionan modelos que se ajustan de forma más suave a los datos (favorecen la generalización del modelo). En estos casos, la predicción en un regióñ es el promedio de los K puntos más cercanos al punto bajo consideración.
- ❖ **Ajuste del valor de K:** analizando los errores en las predicciones de modelos KNN con diferentes valore de K en un conjunto de validación.

BIBLIOGRAFÍA PRINCIPAL

- James G., Witten D.,Hastie T. y Tibshirani R (2017). "An Introduction to Statistical Learning, with applications in R", Springer, Recurso libre: <http://faculty.marshall.usc.edu/gareth-james/ISL/index.html>
- "Análisis de datos", Grado en Ingeniería Informática, Jesús García Herrero, Ricardo Aler Mur, Julia Sidorova, Universidad Carlos III de Madrid. OpenCourseWare: <http://ocw.uc3m.es/ingenieria-informatica/analisis-de-datos>

Otras referencias consultadas:

- "Data types and similarities", Introduction to Data Mining Course (CS 591.03), Abdullah Mueen, University Of New Mexico. Online-available at: https://www.cs.unm.edu/~mueen/Teaching/CS591/Lectures/2_Data.pdf
- "Aprendizaje automático para el análisis de datos", Grado en Estadística y Empresa, Ricardo Aler, Universidad Carlos III de Madrid. OpenCourseWare: <http://ocw.uc3m.es/ingenieria-informatica/aprendizaje-automatico-para-el-analisis-de-datos>
- "Some of the figures in this presentation are taken from "An Introduction to Statistical Learning, with applications in R" (Springer, 2013) with permission from the authors: G. James, D. Witten, T. Hastie and R. Tibshirani "

RECONOCIMIENTO DE OBJETOS

- INTRODUCCIÓN
 - Clasificación: enfoque basado en la teoría de la Decisión
- TECNICAS BÁSICAS DE CLASIFICACIÓN
 - Análisis discriminante
 - K-vecinos más cercanos
- DESARROLLO DE SISTEMAS DE RECONOCIMIENTO DE OBJETOS: ASPECTOS PRÁCTICOS
 - Análisis y pre-procesamiento de datos
 - Selección de atributos
 - Evaluación de modelos

RECONOCIMIENTO DE OBJETOS

- INTRODUCCIÓN
 - Clasificación: enfoque basado en la teoría de la Decisión
- TECNICAS BÁSICAS DE CLASIFICACIÓN
 - Análisis discriminante
 - K-vecinos más cercanos
- DESARROLLO DE SISTEMAS DE RECONOCIMIENTO DE OBJETOS: ASPECTOS PRÁCTICOS
 - Análisis y pre-procesamiento de datos
 - Selección de atributos
 - Evaluación de modelos

RECONOCIMIENTO DE OBJETOS

□ DESARROLLO DE SISTEMAS DE RECONOCIMIENTO DE OBJETOS: ASPECTOS PRÁCTICOS

- Análisis y pre-procesamiento de datos
 - Predictores de distinta naturaleza
 - Estandarización / normalización
 - Valores anómalos

RECONOCIMIENTO DE OBJETOS

- DESARROLLO DE SISTEMAS DE RECONOCIMIENTO DE OBJETOS: ASPECTOS PRÁCTICOS
 - Análisis y pre-procesamiento de datos
 - Predictores de distinta naturaleza
 - Estandarización / normalización
 - Valores anómalos

TIPOS DE ATRIBUTOS SEGÚN LA ESCALA DE MEDIDA

□ ATRIBUTO NUMÉRICO (atributos cuantitativos, indican cantidad – valor real o entero)

1. DE INTERVALO

- Los valores tienen orden, valores más altos indican más cantidad de la propiedad que se cuantifica.
- Escala de medida basada en unidades de igual tamaño.
- El punto o valor cero no indica la ausencia de la propiedad, no es el origen absoluto de la escala. De esta forma, podemos referenciar los valores por medio de la suma o resta de intervalos, pero no tiene sentido la proporción o razón entre ellos.
- Ejemplo: temperatura en °C (por ejemplo, respecto a 4° y 8°, podemos decir que 8° es una temperatura 4 unidades mayor que 4°, pero no es el doble de temperatura – notar que el cero absoluto de temperatura es 0 Kelvin = -273°C).

2. DE RAZÓN

- Los datos tienen todas las propiedades de los datos de intervalo, pero ahora, la proporción entre ellos tiene sentido.
- De esta forma, el valor cero de la escala indica la ausencia de la propiedad a medir..
- Ejemplo: temperatura en K (ahora 8K si es el doble de temperatura que 4K), conteos, salario, tiempo en realizar una tarea, medidas de altura, longitud...

TIPOS DE ATRIBUTOS SEGÚN LA ESCALA DE MEDIDA

ATRIBUTO CUALITATIVO (indica categorías, estados o nombres de cosas):

1. NOMINAL O CATEGÓRICO (no importa el orden de sus posibles valores)

- En esta escala carecen de sentido el orden de las etiquetas, así como la comparación y las operaciones aritméticas. La única finalidad de este tipo de datos es clasificar a las observaciones.
- Ejemplos: color del pelo, género, estado civil...

❖ ATRIBUTO BINARIO: caso particular de atributo nominal que únicamente tiene dos estados (0 y 1)

- **Atributo binario simétrico:** los dos estados tienen el mismo peso o importancia (no importa qué estado se codifique con 0 o 1). Ejemplo: género.
- **Atributo binario asimétrico:** uno de los estados es más relevante (por convención, se le asigna el valor 1). Ejemplo: test médico (positivo vs negativo).

2. ORDINAL (el orden de sus posibles valores tiene significado)

- Muestran las propiedades de los datos nominales, pero además tiene sentido el orden (o jerarquía)
- En esta variable sigue sin tener sentido las operaciones aritméticas.
- Los valores indican un orden (ranking) pero la magnitud entre ellos no es conocida.
- Ejemplos: tamaño (pequeño, mediano, grande), nivel de satisfacción (en escala de 1 a 5)...

¿COMO TRATAR PREDICTORES DE DISTINTO TIPO, NUMÉRICOS / CUALITATIVOS?

NORMA GENERAL: Trabajar con todos los predictores numéricos transformando los predictores cualitativos

Si atributo cualitativo nominal o categórico:

- Por cada predictor categórico se crean tantas variables “ficticias” (*dummy variables*) como posibles valores pueden tomar (cada variable está asociada con un posible valor del predictor categórico).
- Estas variables pueden tomar únicamente dos posibles valores numéricos, 0 o 1, activándose el valor 1 cuando el valor del predictor es el de la variable en cuestión.

Si atributo cualitativo ordinal:

Supongamos un conjunto de datos X donde cada observación está descrita por p atributos (X_1, X_2, \dots, X_p) . Sea X_f un atributo ordinal con M_f valores.

$$X = \begin{bmatrix} X_1 \\ \vdots \\ X_n \end{bmatrix} = \begin{bmatrix} X_{11} & \cdots & X_{1p} \\ \vdots & \ddots & \vdots \\ X_{n1} & \cdots & X_{np} \end{bmatrix}$$

→ **Procedimiento para la conversión del atributo a naturaleza numérica:**

1. Establecer una escala de valores numérica para cada atributo, un ranking: $\text{Ranking}_X_f = \{1, 2, \dots, M_f\}$
2. Reemplazar los valores de las muestras de cada atributo por su ranking: $X_{if} \rightarrow R_{if} \in \{1, 2, \dots, M_f\}$
3. Normalizar el rango valores a $[0, 1]$: $R_{if} \rightarrow X_{if}^* = \frac{R_{if}-1}{M_f - 1} \in [0, 1]$

¿COMO TRATAR PREDICTORES DE DISTINTO TIPO, NUMÉRICOS / CUALITATIVOS?

NORMA GENERAL: Trabajar con todos los predictores numéricos transformando los predictores cualitativos

Si atributo numérico de razón:

Los valores de este tipo de atributos pueden presentar una escala de variación no lineal con una escala que puede estar distorsionada respecto a las de los atributos numéricos de intervalo (caso general).

Tratamiento:

❖ **Aplicar transformación logarítmica:**

→ Este tipo de atributos suelen ser variables de rápido crecimiento/decrecimiento, con una escala aproximadamente exponencial (ejemplo: crecimiento de una bacteria).

❖ **Considerarlos como atributos ordinales continuos y tratar sus rankings como variable numérica de intervalo:**

→ Discretizar el rango de variación del atributo X_f en M_f intervalos. Cada intervalo puede tener su propia amplitud.

→ Cada valor del atributo se asocia con el nivel del intervalo al que pertenece. $X_{if} \rightarrow R_{if} \in \{1, 2, \dots, M_f\}$

→ Normalizar el rango de valores a $[0, 1]$: $R_{if} \rightarrow X_{if}^* = \frac{R_{if}-1}{M_f-1} \in [0, 1]$

RECONOCIMIENTO DE OBJETOS

- DESARROLLO DE SISTEMAS DE RECONOCIMIENTO DE OBJETOS: ASPECTOS PRÁCTICOS
 - Análisis y pre-procesamiento de datos
 - Predictores de distinta naturaleza
 - Estandarización / normalización
 - Valores anómalos

NECESIDAD DE NORMALIZAR/ESTANDARIZAR LAS OBSERVACIONES: están descritas por vectores de atributos compuestos por variables que pueden tener distinta naturaleza, rangos de variación y/o dispersión diferentes.

□ NORMALIZACIÓN DE PREDICTORES NUMÉRICOS

→ Transformar los valores de los predictores para que todos presenten el mismo rango de variación:

Ejemplo: dado un conjunto de datos, la normalización en el rango [0,1] de cada predictor X_i :

$$X = \begin{bmatrix} X_1 \\ \vdots \\ X_n \end{bmatrix} = \begin{bmatrix} X_{11} & \cdots & X_{1p} \\ \vdots & \ddots & \vdots \\ X_{n1} & \cdots & X_{np} \end{bmatrix} \quad X = (X_1, X_2, \dots, X_p) \rightarrow X^* = (X_1^*, X_2^*, \dots, X_p^*) \text{ variables homogenizadas en } [0, 1]$$
$$X_{zi} \rightarrow X_{zi}^* = \frac{X_{zi} - \min(X_i)}{\max(X_i) - \min(X_i)}$$

- ❖ La normalización sólo realiza un cambio de escala en los valores de los atributos, pero no uniformiza las diferentes dispersiones que pueden presentar.

Este hecho puede provocar distorsiones en las técnicas de aprendizaje que requieren el cálculo de distancias entre las observaciones (los atributos con menos dispersión pierden influencia respecto a los que presentan una mayor dispersión)

→ Solución: *estandarizar predictores*

La estandarización permite que todos los atributos tengan el mismo peso en el cálculo de distancias: los valores de cada atributo se transforman en variables sin unidad variando entorno a su valor medio representativo en unidades relativas a la desviación que presentan (el valor medio y la desviación de cada atributo se calculan en el conjunto de datos disponibles del atributo en cuestión).

□ **ESTANDARIZACIÓN DE PREDICTORES NUMÉRICOS**

→ Obtención de medida estandarizada de cada atributo o Z-Score

Dado un conjunto de datos X compuesto por n observaciones de p variables: $X = \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} X_{11} & \cdots & X_{1p} \\ \vdots & \ddots & \vdots \\ X_{n1} & \cdots & X_{np} \end{bmatrix}$

Se define:

$$\text{Media de cada variable: } M = (\mu_1, \mu_2, \dots, \mu_p) \text{ con } \mu_i = \frac{1}{n} \sum_{z=1}^n X_{zi}$$

$$\text{Desviación estándar de cada variable: } STD = (\sigma_1, \sigma_2, \dots, \sigma_p) \text{ con } \sigma_i = \sqrt{\frac{1}{n-1} \sum_{z=1}^n (X_{zi} - \mu_i)^2}$$

PROCEDIMIENTO DE ESTANDARIZACIÓN: expresar los valores en unidades de la desviación estándar

$$X_{zi} \rightarrow Z_{zi} = \frac{X_{zi} - \mu_i}{\sigma_i}$$

$X = (X_1, X_2, \dots, X_p) \rightarrow Z = (Z_1, Z_2, \dots, Z_p)$ variables estandarizadas o tipificadas con media 0 y desviación típica 1

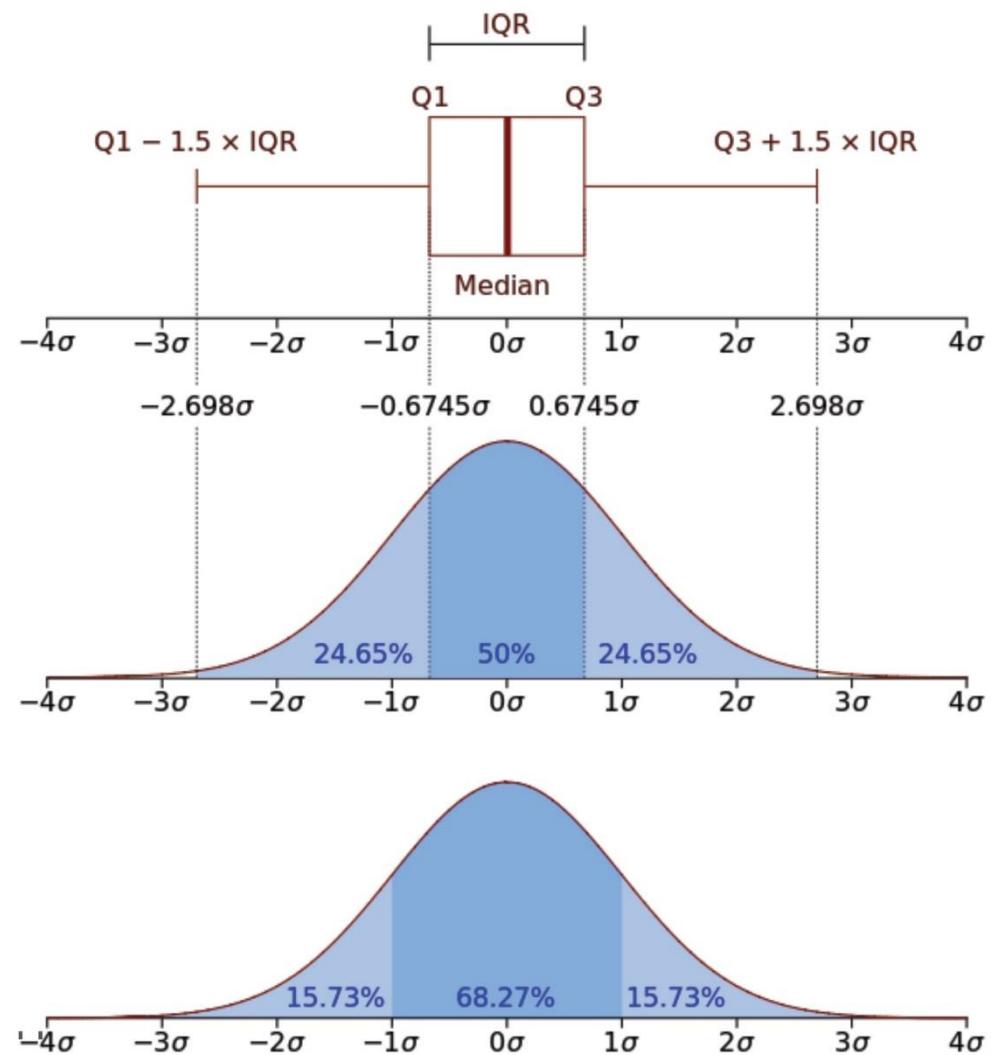
RECONOCIMIENTO DE OBJETOS

- DESARROLLO DE SISTEMAS DE RECONOCIMIENTO DE OBJETOS: ASPECTOS PRÁCTICOS
 - Análisis y pre-procesamiento de datos
 - Predictores de distinta naturaleza
 - Estandarización / normalización
 - Valores anómalos

ELIMINACIÓN DE VALORES ANÓMALOS

CASO UNIDIMENSIONAL

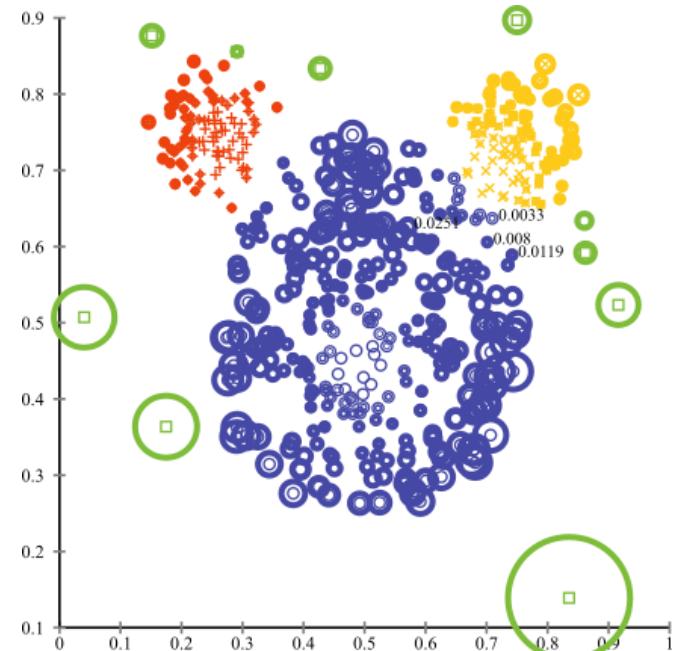
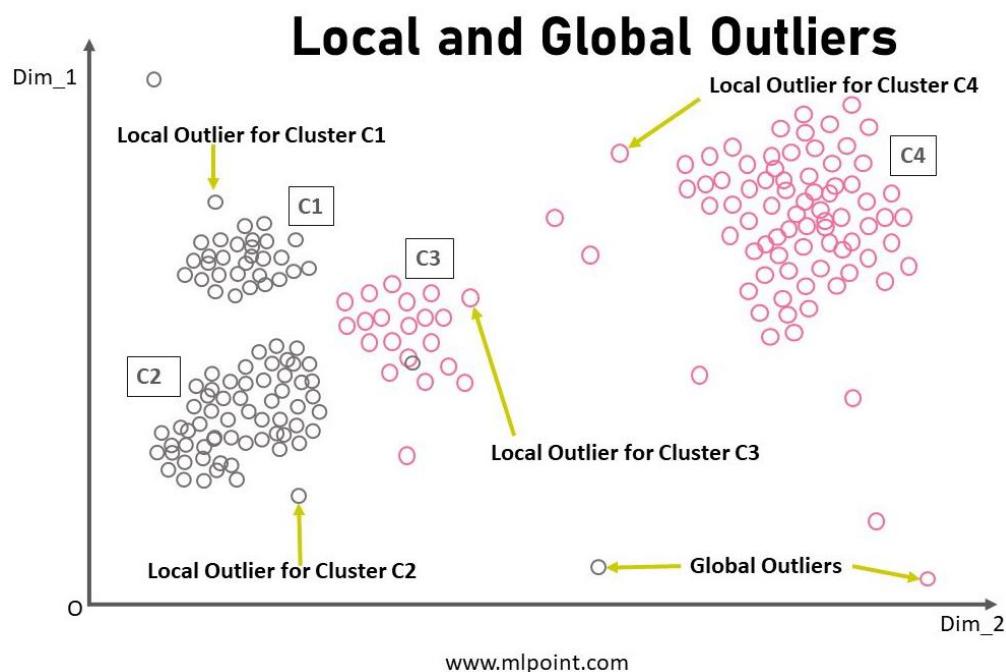
→ Análisis individual de las instancias disponibles de cada atributo.



ELIMINACIÓN DE VALORES ANÓMALOS

□ CASO P-DIMENSIONAL

→ Análisis de las instancias en el espacio definido por sus predictores mediante técnicas de agrupamiento (clustering)



FUENTES CONSULTADAS:

- Introduction to Data Mining Course (CS 591.03), Abdullah Mueen, University Of New Mexico.
 - “Data types and similarities”. Online-available at:
https://www.cs.unm.edu/~mueen/Teaching/CS591/Lectures/2_Data.pdf
 - “Data transformation and dimensionality reduction”. Online-available at:
https://www.cs.unm.edu/~mueen/Teaching/CS591/Lectures/3_Data.pdf
 - “Outlier detection”, Introduction to Data Mining Course”. Online-available at:
https://www.cs.unm.edu/~mueen/Teaching/CS591/Lectures/9_Data.pdf
- “Aprendizaje automático para el análisis de datos”, Grado en Estadística y Empresa, Ricardo Aler, Universidad Carlos III de Madrid. OpenCourseWare: <http://ocw.uc3m.es/ingenieria-informatica/aprendizaje-automatico-para-el-analisis-de-datos>

RECONOCIMIENTO DE OBJETOS

- INTRODUCCIÓN
 - Clasificación: enfoque basado en la teoría de la Decisión
- TECNICAS BÁSICAS DE CLASIFICACIÓN
 - Análisis discriminante
 - K-vecinos más cercanos
- DESARROLLO DE SISTEMAS DE RECONOCIMIENTO DE OBJETOS: ASPECTOS PRÁCTICOS
 - Análisis y pre-procesamiento de datos
 - Selección de atributos
 - Evaluación de modelos

RECONOCIMIENTO DE OBJETOS

- INTRODUCCIÓN
 - Clasificación: enfoque basado en la teoría de la Decisión
- TECNICAS BÁSICAS DE CLASIFICACIÓN
 - Análisis discriminante
 - K-vecinos más cercanos
- DESARROLLO DE SISTEMAS DE RECONOCIMIENTO DE OBJETOS: ASPECTOS PRÁCTICOS
 - Análisis y pre-procesamiento de datos
 - Selección de atributos
 - Evaluación de modelos

METODOLOGÍA Y MÉTRICAS DE EVALUACIÓN DE UN MODELO

¿ Cómo evaluar ?



Metodología de evaluación

¿ Cómo cuantificar el rendimiento de un modelo ?



Métricas de evaluación

ALGUNAS MÉTRICAS DE EVALUACIÓN:

❖ MODELO DE REGRESIÓN:

$$\rightarrow \text{MSE (Mean Squared Error): } MSE = \frac{RSS}{n} = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

$$RSS (\text{Sum of Residual Squares}):$$

$$RSS = \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

$$\rightarrow \text{RSE (Residual Standard Error): } RSE = \sqrt{\frac{1}{n-2} RSS} = \sqrt{\frac{1}{n-2} \sum_{i=1}^n (y_i - \hat{y}_i)^2}$$

❖ MODELO DE CLASIFICACIÓN

→ **Tasa de error (Error Rate)** cometido al aplicar el modelo sobre n observaciones:

$$\text{Error Rate} = \frac{1}{n} \sum_{i=1}^n I(y_i \neq \hat{y}_i) \quad \text{con } I(y_i \neq \hat{y}_i) = \begin{cases} 1 & \text{si } y_i \neq \hat{y}_i \\ 0 & \text{si } y_i = \hat{y}_i \end{cases}$$

→ **TP, TN, FP, FN, Matriz de confusión, Tasa de Aciertos, Sensibilidad, Especificidad, Precisión, Recall, curva ROC y área bajo su curva (clasificación binaria)**

RECONOCIMIENTO DE OBJETOS

- DESARROLLO DE SISTEMAS DE RECONOCIMIENTO DE OBJETOS: ASPECTOS PRÁCTICOS
 - Evaluación de modelos
 - Metodología de evaluación
 - Métricas de evaluación

RECONOCIMIENTO DE OBJETOS

- DESARROLLO DE SISTEMAS DE RECONOCIMIENTO DE OBJETOS: ASPECTOS PRÁCTICOS
 - Evaluación de modelos
 - Metodología de evaluación
 - Métricas de evaluación

EVALUACIÓN DEL MODELO: OBSERVACIONES GENERALES

- Evaluar un modelo significa estimar cuál va a ser su precisión con datos futuros (es decir, no utilizados durante el entrenamiento del modelo).
- El hecho de que un modelo funcione muy bien con los datos de entrenamiento, no significa necesariamente que vaya a hacerlo con datos futuros: El modelo puede haberse *sobreajustado* (*overfitting*) a los datos de entrenamiento.
 - Ejemplo: si a un alumno se le evalúa (examen) con exactamente los mismos problemas con los que aprendió, no se demuestra la capacidad de generalización (tan sólo la capacidad de memorización)
- Por tanto: *un proceso de aprendizaje sobre un conjunto de datos disponibles requiere obtener un modelo y una estimación sobre su precisión futura*

¿Cómo usamos los datos disponibles para el aprendizaje de un modelo y estimar cuál es su rendimiento futuro?

- **Forma básica de hacerlo:** dividir el conjunto de datos disponibles en dos conjuntos, entrenamiento y test.
 - *Conjunto de datos de entrenamiento:* datos utilizados en la fase de aprendizaje del modelo. Suele dividirse en:
 - *Conjunto de entrenamiento:* conjunto de datos específico para el entrenamiento del modelo.
 - *Conjunto de validación:* conjunto de datos reservado para el ajuste de hiperparámetros y validación del modelo entrenado.
 - *Conjunto de datos de test:* datos para realizar la evaluación final del modelo y estimar su rendimiento futuro.

□ PRINCIPIO BÁSICO EN LA EVALUACIÓN DE MODELOS:

- *La evaluación final de un modelo nunca debe realizarse sobre los datos que se utilizan en el aprendizaje del modelo (ni siquiera sobre el conjunto de validación).*

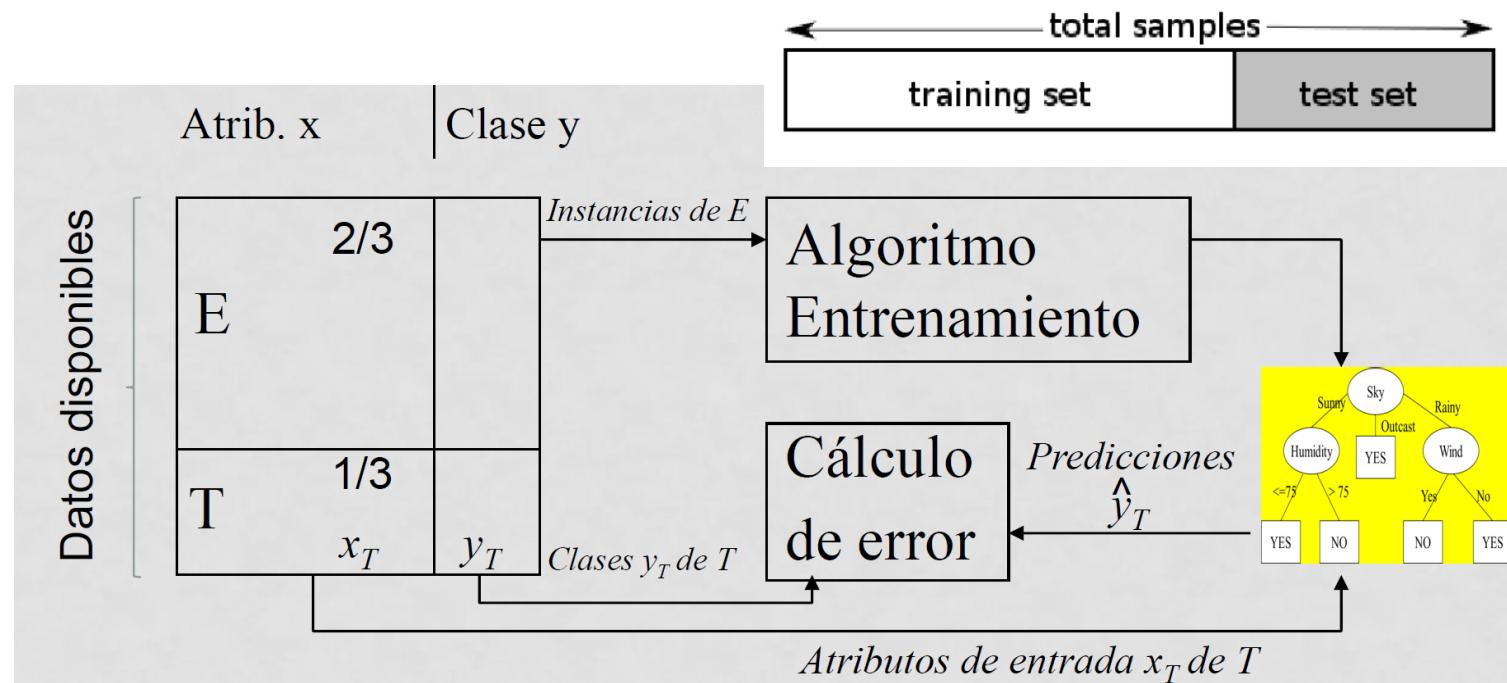
- **Limitación:** número reducido de datos que impidan seleccionar conjuntos representativos de entrenamiento y test
 - *El procedimiento utilizado en la evaluación está condicionado por el número de observaciones disponibles*
- **Diferentes estrategias:** métodos *holdout* y *validación cruzada (crossvalidation)*.

PROCEDIMIENTOS DE EVALUACIÓN: HOLDOUT

□ **Holdout** (evaluación mediante particiones de entrenamiento y test):

→ División de forma aleatoria del conjunto de datos disponibles en dos conjuntos:

- Entrenamiento (E): conjunto utilizado en el entrenamiento del modelo.
- Test (T): conjunto utilizado para estimar el rendimiento del modelo.



→ ***El conjunto de test debe ser lo mas representativo posible.*** Por ejemplo, en problemas de clasificación con clases desbalanceadas, es conveniente que las particiones sean estratificadas por clase y que la proporción entre las clases que existe en el conjunto de datos original, se mantenga en los conjuntos de entrenamiento y test.

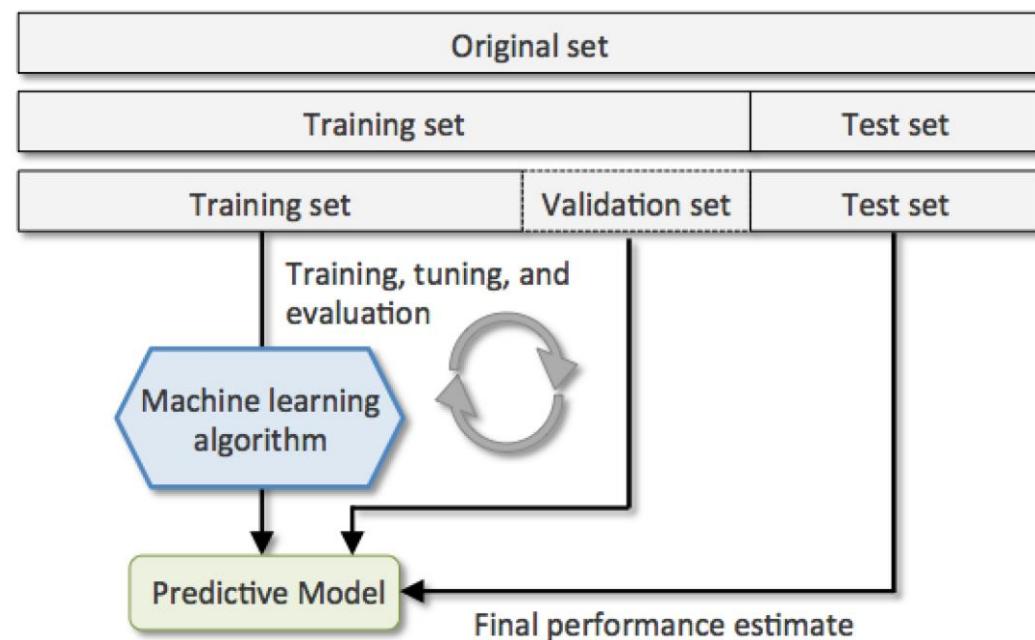
PROCEDIMIENTOS DE EVALUACIÓN: HOLDOUT

Holdout con validación (evaluación mediante particiones de entrenamiento/validación y test):

- Se incorpora un tercer conjunto de datos (conjunto de validación) para ajustar los hiperparámetros o determinados aspectos del modelo con el objetivo de validarlos. Es el método recomendado cuando se dispone de un conjunto de datos amplio y representativo del problema en cuestión.
 - Porcentajes habituales en la división del conjunto de datos en Entrenamiento/Validación/Test: 50/20/30 o 40/20/40.

→ Es importante que:

- ❖ Una vez se entrene y ajuste el modelo en el conjunto de entrenamiento y validación, haya una evaluación completamente independiente en el conjunto de test. De esta forma, es posible realizar una estimación del rendimiento futuro del modelo.
- ❖ El modelo final se entrene utilizando todos los datos del conjunto disponible (entrenamiento/validación/test).



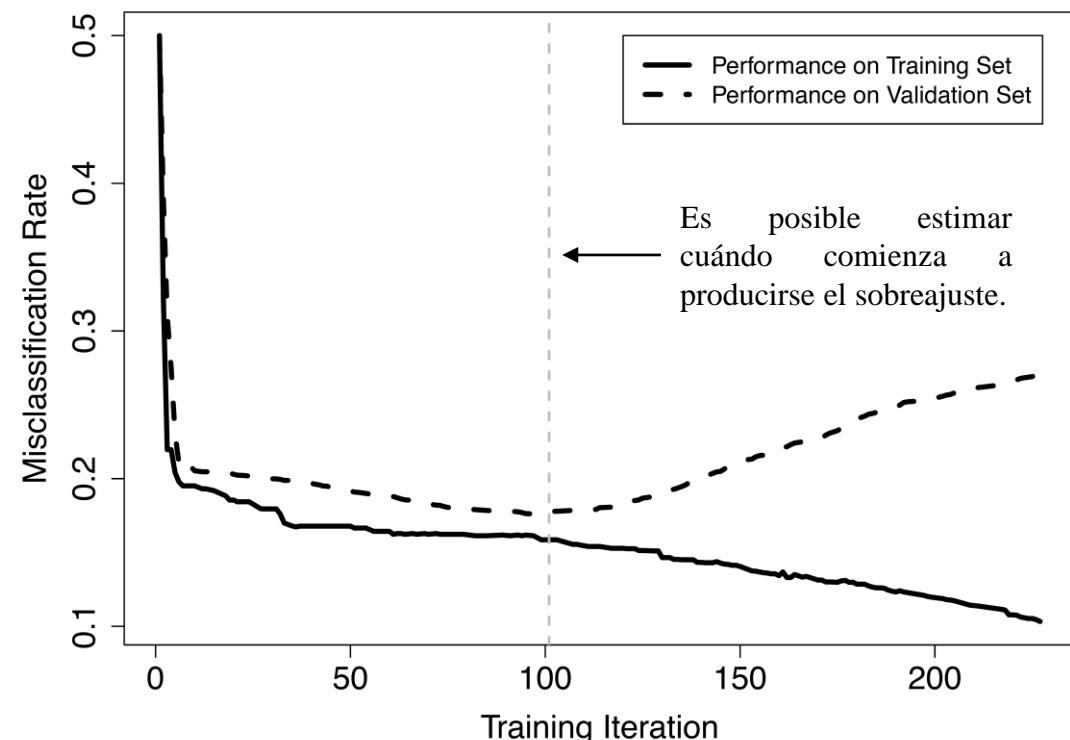
- La estimación de rendimiento del modelo en el conjunto de test, es una estimación pesimista en el sentido que ha sido realizada con un modelo que no utiliza todo el conjunto de datos disponible.

PROCEDIMIENTOS DE EVALUACIÓN: HOLDOUT

- **VENTAJA DEL CONJUNTO DE VALIDACIÓN:** permite generar un modelo que minimice el sobreajuste a los datos de entrenamiento.

❖ *Training iteration:* en cada iteración se consiguen modelos cada vez más flexibles y ajustados a los datos de entrenamiento (el error decrece). La gráfica tiene carácter general, cada iteración en podría representar:

- Cada época en el proceso de entrenamiento de una red neuronal.
- Barido de valores de un hiperparámetro de un determinado modelo. Por ejemplo:
 - Si kNN, la primera iteración comienza con un valor elevado de k ; en cada iteración el valor de k disminuye.
 - Si SVM, barido en el parámetro C que controla la anchura del margen (comenzando con el valor de C más alto y por tanto permitiendo más violaciones del margen).
 - Si Árbol de Decisión, proceso selección de árbol podado: cada iteración representaría un árbol de una determinada profundidad o número de nodos terminales. Así, la primera iteración comienza con el árbol de un solo nodo (mínima profundidad); cada iteración analiza un árbol de mayor tamaño.



- Si SVM, barido en el parámetro C que controla la anchura del margen (comenzando con el valor de C más alto y por tanto permitiendo más violaciones del margen).

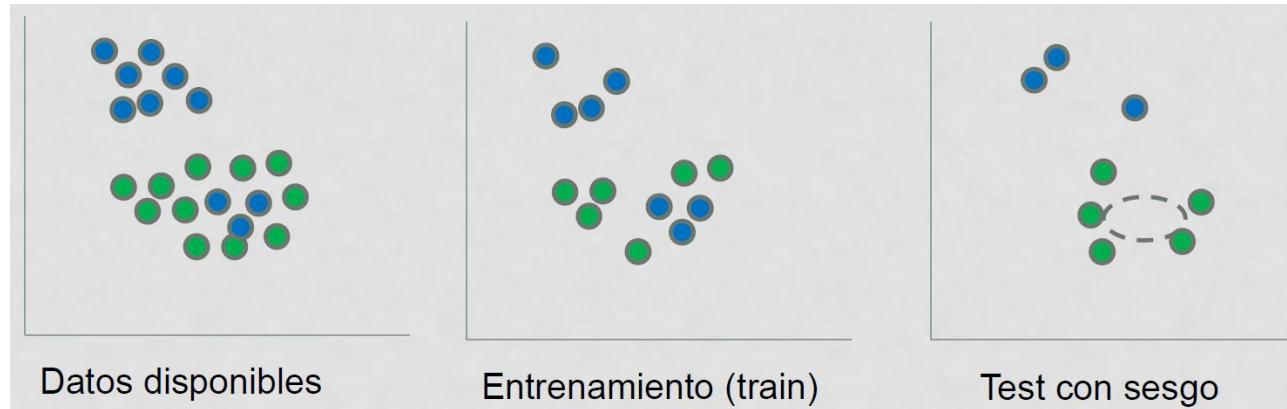
PROCEDIMIENTOS DE EVALUACIÓN: *HOLDOUT*

Ejemplo de metodología de ajuste hiperparámetro y evaluación de un modelo mediante holdout con validación

- Supongamos que vamos a ajustar un árbol de decisión utilizando un conjunto de datos que se ha dividido en Entrenamiento/Validación/Test
- Supongamos que tenemos que decidir la profundidad o tamaño del árbol (número de sus nodos terminales). Procedimiento:
 1. Se generan árboles con distintas profundidades en el conjunto de entrenamiento.
 2. Se evalúan los árboles generados en el conjunto de validación.
 3. Se selecciona el valor de profundidad cuyo árbol haya generado el mínimo error.
 4. Se genera un árbol con el valor de profundidad seleccionado utilizando el conjunto completo de datos de entrenamiento y validación.
 5. Se evalúa el árbol en el conjunto de test.

PROCEDIMIENTOS DE EVALUACIÓN: PROBLEMA HOLDOUT

- **NÚMERO REDUCIDO DE DATOS DISPONIBLES:** conjuntos de entrenamiento/test sesgados, no representativos.



El conjunto de test no incluye ninguna muestra de la clase azul de mayor dificultad de predicción, no es representativo del conjunto de entrenamiento.

- ❖ **POSIBLE SOLUCIÓN:** *holdout con train-test repetido (como los sesgos ocurren por azar, la repetición del proceso favorece que los sesgos de unas y otras particiones se cancelen). Procedimiento:*

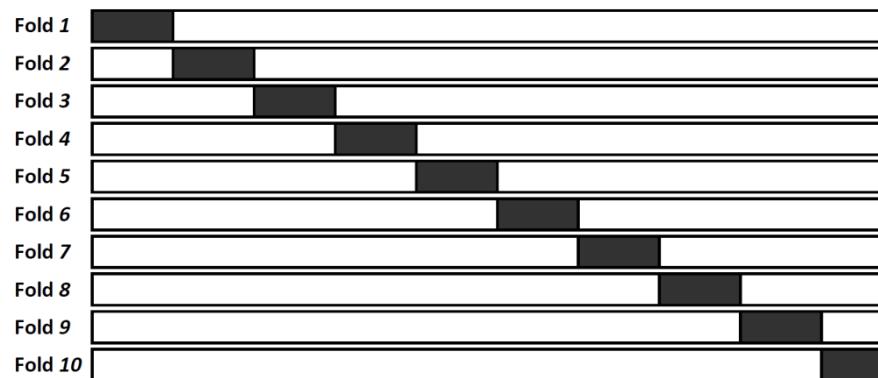
1. Repetir múltiples veces:
 - Selección aleatoria de conjunto de Train y Test (por ejemplo, en proporciones 2/3 y 1/3).
 - Generar el modelo en el conjunto de Train.
 - Evaluar el modelo en el conjunto de Test.
 2. Generar las métricas de rendimiento del modelo como un promedio de sus valores en los distintos conjuntos de test.

- ❖ **PROBLEMA:** las distintas particiones de test pueden solaparse unas con otras por casualidad, no son independientes (lo ideal es que no haya solapes) → **SOLUCIÓN:** *método de validación cruzada*.

PROCEDIMIENTOS DE EVALUACIÓN: VALIDACIÓN CRUZADA

Validación cruzada:

- División del conjunto de datos disponible en K partes iguales (K fold - cross validations)
 - Las divisiones se realizan sin solapamiento en los datos (preferentemente estratificadas).
 - Valores usuales de K : 10, 5 (en cualquier caso, depende del número de datos disponibles).



- Generación de K modelos y evaluación de los mismos (mediante una métrica dada) sobre su partición asociada:
 - Cada modelo k ($k=1, \dots, K$) se entrena con los datos correspondientes a las $K-1$ particiones restantes.
 - Cada modelo entrenado k ($k=1, \dots, K$) se evalúa sobre los datos de la partición k .
- La estimación del rendimiento futuro del modelo se realiza mediante el promedio de la métrica de rendimiento obtenida en el paso anterior para cada partición k ($k=1, \dots, K$).
- La generación del modelo final se realiza mediante un entrenamiento llevado a cabo sobre el conjunto completo de datos disponible.

PROCEDIMIENTOS DE EVALUACIÓN: VALIDACIÓN CRUZADA – OBSERVACIONES

- **Generación de k ($k=1,\dots,K$) modelos y modelo final:** aunque los k modelos generados son parecidos (comparten gran parte de su conjunto de datos de entrenamiento), se genera un único modelo final utilizando para su entrenamiento todo el conjunto de datos disponible.
- **Principal objetivo del método:** el procedimiento, planteado de esta forma, tiene como objetivo proporcionar una estimación del rendimiento futuro de un modelo lo más realista posible cuando el número de observaciones disponibles no sea extenso.
 - ❖ **Es el procedimiento adecuado para evaluar cómo de bueno será un algoritmo de aprendizaje (en términos de generalización) sobre un conjunto reducido de datos:**
 - Cada dato aparece exactamente una vez en un conjunto de test: ningún ejemplo se escapa del entrenamiento ni de la evaluación.
 - El promedio de rendimiento final del modelo puede acompañarse por una medida de la varianza del rendimiento de las evaluaciones llevadas a cabo en cada uno de los conjuntos k de prueba.
- **VARIANTE: VALIDACIÓN CRUZADA DEJANDO UNO FUERA (LEAVE-ONE-OUT)**
 - ❖ Es el caso en el que K es igual al total de observaciones disponibles (N): $K=N$
 - Se generan N modelos (uno por cada dato) y se realizan N evaluaciones específicas para cada dato.
 - ❖ Costoso computacionalmente por lo que se recomienda para conjunto de datos reducidos.

PROCEDIMIENTO DE VALIDACIÓN CRUZADA PARA AJUSTE DE HIPERPARÁMETROS

Parámetros de un modelo:

- Son las variables del modelo que se estiman durante el proceso de entrenamiento (coeficientes en una regresión lineal o polinómica, pesos en una red neuronal, vectores de soporte en una máquina de vector soporte).

Hiperparámetros de un modelo:

- Son parámetros que se configuran antes del entrenamiento del modelo y no forman parte del modelo como tal:
 - **k NN** → k
 - **Árboles de decisión** → número mínimo de instancias para subdividir, número de nodos terminales;
 - **SVM** → C (grado de violación del margen); d (grado del polinomio en kernel polinomial); γ (kernel radial)
- Dependiendo de su valor, el algoritmo generará unos modelos u otros, algunos con errores más altos y otros con errores más bajos. Es por tanto importante encontrar los valores óptimos de los hiperparámetros, aquellos que producen los modelos con menor error.
- Generalmente, estos valores óptimos no se conocen a priori, para establecerlos se deben utilizar reglas genéricas, valores que han funcionado en problemas similares o ajustarlos mediante prueba y error.

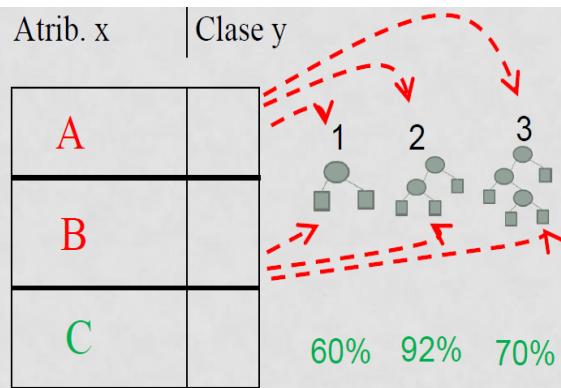
Método de prueba y error:

- ❖ Utilizando un único conjunto de validación (ya visto cuando se presentó el método *holdout* con validación)
- ❖ Mediante validación cruzada (es lo que vamos a ver a continuación).

□ **Ejemplo de metodología de ajuste hiperparámetro y evaluación de un modelo mediante validación cruzada**

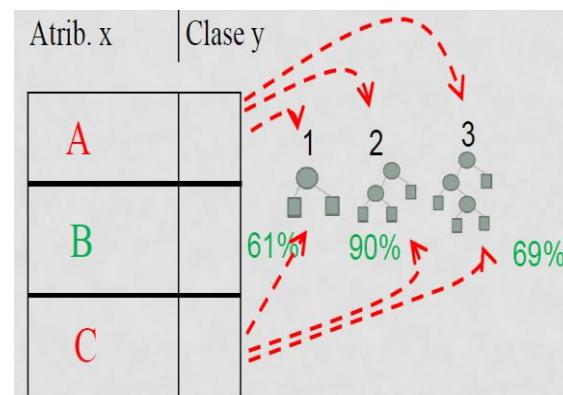
→ Supongamos que vamos a ajustar un árbol de decisión utilizando un conjunto de datos que se ha dividido en 3 partes (en la figura, X-Y-Z) para aplicar *3-fold cross validation*. Ilustración del procedimiento para establecer la profundidad del árbol (entre tres valores posibles):

Entrenamos en A-B, evaluamos (en el ejemplo acierto en la clasificación) en C



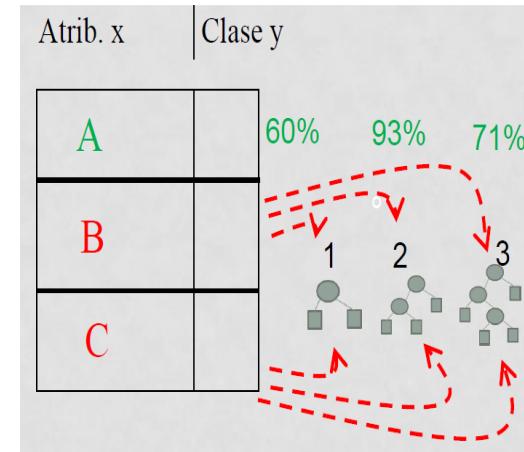
Estimamos la calidad de cada profundidad mediante la media de los valores del acierto obtenido para cada *fold*.

Entrenamos en A-C, evaluamos en B



Construimos el modelo final utilizando todos los datos y para el valor de profundidad de mayor acierto medio

Entrenamos en B-C, evaluamos en A



Atrib. x		Clase y		
Datos disponibles		1	2	3
A		60%	93%	71%
B		61%	90%	69%
C		60%	92%	70%

Medias	60.33%	91.66 %	70%

Atrib. x		Clase y		
Construcción del modelo final con profundidad 2, pero con todos los datos.		Modelo final con profundidad 2		

RECONOCIMIENTO DE OBJETOS

□ DESARROLLO DE SISTEMAS DE RECONOCIMIENTO DE OBJETOS: ASPECTOS PRÁCTICOS

- Evaluación de modelos
 - Metodología de evaluación
 - Métricas de evaluación

□ **MÉTRICAS DE REGRESIÓN:**

Valores reales y predichos de la salida del conjunto de observaciones: (y_1, y_2, \dots, y_n) ; $(\hat{y}_1, \hat{y}_2, \dots, \hat{y}_n)$

$$\rightarrow \textbf{MSE} \text{ (Mean Squared Error): } MSE = \frac{RSS}{n} = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2 \quad RSS \text{ (Sum of Residual Squares):}$$

$$\rightarrow \textbf{RMSE} \text{ (Root Mean Squared Error): } RMSE = \sqrt{MSE}$$

$$\rightarrow \textbf{RSE} \text{ (Residual Standard Error): } RSE = \sqrt{\frac{1}{n-2} RSS} = \sqrt{\frac{1}{n-2} \sum_{i=1}^n (y_i - \hat{y}_i)^2}$$

$$\rightarrow \textbf{MAE} \text{ (Mean Absolute Error): } MAE = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i| \Rightarrow \text{Valores mal predichos } (abs(y_i - \hat{y}_i) \uparrow\uparrow) \text{ tienen menos peso en la media.}$$

❖ **Problema de estas métricas:** su valor es relativo a la escala de la variable de salida

→ Ejemplo: si multiplicamos las variables de salida y_i por 1000, los valores de RMSE y MAE serán del orden de 1000 veces más grandes, sin que eso implique que el modelo sea 1000 veces peor.

❖ **Solución:** construir métricas relativas al error que comete el modelo dado por la media de los valores de salida \bar{y} .

$$\rightarrow \textbf{RSE} \text{ (Relative Squared Error): } RSE = \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y})^2} \quad \rightarrow \textbf{RRSE} \text{ (Root Relative Squared Error): } RRSE = \sqrt{RSE}$$

$$\rightarrow \textbf{RAE} \text{ (Relative Absolute Error): } RAE = \frac{\sum_{i=1}^n |y_i - \hat{y}_i|}{\sum_{i=1}^n |y_i - \bar{y}|}$$

Son números típicamente entre 1 (el modelo acierta igual que la media) y 0 (el modelo no comete errores), aunque pueden ser mayores que 1 si el modelo es muy malo.

□ MÉTRICAS DE CLASIFICACIÓN: CLASIFICACIÓN BINARIA

❖ Matriz de confusión:

→ Tabla cruzada que muestra el conteo de los aciertos y errores del clasificador en cada una de las clases (predicciones) en relación a la clase real de las observaciones (*ground-truth*):

- Clasificación binaria – 2 clases: positiva (“+” o “1”) y negativa (“–” o “0”)
- Tipos de predicciones:
 - **Verdadero Positivo (TP, True Positive):** observación positiva, predicha positiva.
 - **Falso Positivo (FP, False Positive):** observación negativa, predicha positiva.
 - **Verdadero Negativo (TN, True Negative):** observación negativa, predicha negativa.
 - **Falso Negativo (FN, False Negative):** observación positiva, predicha negativa.

		Prediction	
		Positive	Negative
Actual	Positive	TP	FN
	Negative	FP	TN

Diagonales:

- Aciertos de clasificación: TP + TN
- Errores de clasificación: FP + FN

Filas:

- Número de muestras positivas: TP + FN
- Número de muestras negativas: FP + TN

Columnas:

- N° de muestras predichas positivas: TP + FP
- N° de muestras predichas negativas: FN + TN

PRINCIPALES MÉTRICAS DE EVALUACIÓN DE UN CLASIFICADOR BINARIO SOBRE UN DETERMINADO CONJUNTO DE DATOS

❖ RELATIVAS AL NÚMERO TOTAL DE OBSERVACIONES

Tasa de aciertos (*Accuracy*, *Acc*)

Tasa de observaciones clasificadas correctamente por el clasificador

$$Acc = \frac{Nº\ Aciertos\ Clasificación}{Nº\ observaciones} = \frac{TP + TN}{TP + FN + FP + TN}$$

Tasa de errores (*Error Rate*, *ER*)

Tasa de observaciones clasificadas incorrectamente por el clasificador

$$ER = \frac{Nº\ Errores\ Clasificación}{Nº\ observaciones} = \frac{FP + FN}{TP + FN + FP + TN} = 1 - Acc$$

❖ RELATIVAS AL NÚMERO TOTAL DE OBSERVACIONES POSITIVAS

Sensibilidad (*Sensitivity*, *Se* ; *Recall*, *R*)

Tasa de observaciones positivas clasificadas correctamente

$$Se (= R) = \frac{Nº\ Aciertos\ Clase\ +}{Nº\ Observaciones\ Clase\ +} = \frac{TP}{TP + FN}$$

Tasa de Falsos Negativos (*False Negative Rate*, *FNR*)

Tasa de observaciones positivas clasificadas incorrectamente (falsos negativos)

$$FNR = \frac{Nº\ Errores\ Clase\ +}{Nº\ Observaciones\ Clase\ +} = \frac{FN}{TP + FN} = 1 - Se$$

		Prediction	
		Positive	Negative
Actual	Positive	TP	FN
	Negative	FP	TN

Diagonales:

- Número total aciertos: TP + TN
- Número total errores: FP + FN

Filas:

- N° muestras +: TP + FN
- N° muestras -: FP + TN
- N° total muestras: TP+FN+FP+TN

Columnas:

- N° muestras predichas +: TP + FP
- N° muestras predichas -: FN + TN

PRINCIPALES MÉTRICAS DE EVALUACIÓN DE UN CLASIFICADOR BINARIO SOBRE UN DETERMINADO CONJUNTO DE DATOS

❖ RELATIVAS AL NÚMERO TOTAL DE OBSERVACIONES NEGATIVAS

Especificidad (Specificity, Sp)

Tasa de observaciones negativas clasificadas correctamente

$$Sp = \frac{Nº Aciertos Clase -}{Nº Observaciones Clase -} = \frac{TN}{FP + TN}$$

Tasa de Falsos Positivos (False Positive Rate, FPR)

Tasa de observaciones negativas clasificadas incorrectamente (falsos positivos)

$$FPR = \frac{Nº Errores Clase -}{Nº Observaciones Clase -} = \frac{FP}{FP + TN} = 1 - Sp$$

❖ OTRAS MÉTRICAS COMÚNMENTE UTILIZADAS

Precisión (Precision, P)

Tasa de observaciones predichas positivas clasificadas correctamente

$$P = \frac{Nº Aciertos Clase +}{Nº Observaciones Predichas +} = \frac{TP}{TP + FP}$$

Medida F (F -measure, F_1 -score, F_1)

Media armónica de R y P (inversa de la media altimétrica de los inversos)

$$F_1 = \frac{2}{\frac{1}{P} + \frac{1}{R}} = \frac{2 * R * P}{P + R}$$

		Prediction	
		Positive	Negative
Actual	Positive	TP	FN
	Negative	FP	TN

Diagonales:

- Número total aciertos: TP + TN
- Número total errores: FP + FN

Filas:

- N° muestras +: TP + FN
- N° muestras -: FP + TN
- N° total muestras: TP+FN+FP+TN

Columnas:

- N° muestras predichas +: TP + FP
- N° muestras predichas -: FN + TN

MÉTRICAS EVALUACIÓN CLASIFICADOR BINARIO:

COMENTARIOS SOBRE *RECALL*, *Precision* y *F₁-SCORE*

Sensibilidad (*Recall*, *R*): $R = \frac{Nº\ Aciertos\ Clase}{Nº\ Observaciones\ Clase} = \frac{TP}{TP + FN}$

Precisión (*Precision*, *P*): $P = \frac{Nº\ Aciertos\ Clase}{Nº\ Observaciones\ Predichas} = \frac{TP}{TP + FP}$

Medida *F* (*F-measure*, *F₁-score*, *F₁*): $F_1 = \frac{2 * R * P}{P + R}$

		Prediction	
		Positive	Negative
Actual	Positive	TP	FN
	Negative	FP	TN

- *Recall*: cuanto más próxima a 1, el número de observaciones + clasificadas correctamente se aproxima al número de observaciones de esa clase → hay pocos errores en el clase + → Pocos Falsos Negativos
- *Precision*: cuanto más próxima a 1, el número de observaciones predichas + será próximo al número de aciertos del clasificador en esa clase → hay pocas observaciones predichas + de forma errónea → Pocos Falsos Positivos.
- *Un clasificador que todo lo clasifique positivo*: tendrá $R = 1$, pero P será baja debido a los falsos positivos.
- *Un clasificador que realice 1 sola predicción positiva (clasificando el resto de observaciones como de clase negativa)*: si la observación predicha positiva es correcta tendrá $P = 1$, pero R será baja (sólo acierta 1 muestra +, el resto son falsos negativos).
- Se deben analizar los valores de *R* y *P* de forma conjunta. Interesa que un clasificador tenga alta *R* y *P* simultáneamente (clasificador perfecto: $R = P = 1$) → La métrica *F₁* facilita el análisis conjunto de *R* y *P* a través de un solo dato.

□ MÉTRICAS EVALUACIÓN CLASIFICADOR BINARIO: CLASES DESBALANCEADAS

❖ **Problemas de clasificación binaria con clases no balanceadas:** *es necesario analizar el conjunto de las métricas anteriores para captar el impacto de los diferentes tipos de errores.*

→ Supongamos que analizamos el rendimiento de un clasificador únicamente con su tasa de aciertos sobre un conjunto de datos con 990 ejemplos positivos y 10 negativos. Si el clasificador predice todo como positivo, tendría una tasa de acierto de 0,99 y, sin embargo, es “malo”.

❖ **No todos los aciertos o errores podrían tener la misma importancia.**

→ *Diagnóstico de enfermedades:* se suele asociar la clase positiva con la clase donde la enfermedad está presente; en este caso los Falsos Negativos adquieren especial relevancia, porque son casos donde el clasificador predice de forma errónea la ausencia de enfermedad.

❖ **Ejemplo:**

Actual Class\Predicted class	cancer = yes	cancer = no	Total	Recognition(%)
cancer = yes	90	210	300	30.00 (<i>sensitivity</i>)
cancer = no	140	9560	9700	98.56 (<i>specificity</i>)
Total	230	9770	10000	96.40 (<i>accuracy</i>)

→ *Clasificador con Acc = 0,9640 . Podría parecer un valor elevado que indicaría que es adecuado para diagnosticar cáncer. Sin embargo, el valor tan alto de Acc está motivado por su alto rendimiento prediciendo observaciones de la clase negativa (ausencia de cáncer), con Sp = 0,9856 (tiene una tasa de falsos positivos muy baja: 1-Sp).*

→ *La sensibilidad es tan sólo de Se = 0,3000, lo cual motiva una tasa de falsos negativos del 0,70 (de las 300 observaciones con cáncer disponibles, el clasificador diagnostica ausencia de enfermedad en el 70% de los casos).*

EVALUACIÓN CLASIFICADORES BINARIOS QUE MANEJAN INCERTIDUMBRE

❖ **CLASIFICADORES QUE MANEJAN INCERTIDUMBRE:** predicen la clase de una observación cuantificando el grado de incertidumbre de la predicción. Ejemplos de clasificadores vistos en esta asignatura:

→ *Clasificadores Bayesianos, kNN, árboles de decisión: (clasificadores de decisión probabilísticos)*

- Se predice la clase de una observación *cuantificando la probabilidad de pertenencia* de la observación a la clase.
- Clasificación binaria: por defecto, el clasificador predice que una observación es de la clase positiva si su probabilidad asociada es superior a 0,5 (umbral de probabilidad: 0,5).

→ *Clasificadores LDA, SVM (clasificadores binarios de decisión basados en distancia):*

- Se predice la clase de una observación a partir de la evaluación de una función de decisión que cuantifica la distancia de la observación a la frontera de separación entre las clases.
- Clasificación binaria: por defecto, se predice que la observación es de la clase positiva si la evaluación de la función de decisión es mayor que 0 (umbral de distancia = 0).

✓ **Umbral de aceptación de un clasificador binario:** umbral que se utiliza para aceptar que una determinada observación es de la clase positiva (la función de probabilidad o distancia utilizada para predecir la clase, evaluada en la observación bajo consideración, es superior al umbral). El valor de este umbral por defecto se establece en 0,5 (clasificadores probabilísticos) o 0 (clasificadores basados en distancia):

→ *Variando el umbral, obtendríamos para el mismo clasificador y conjunto de datos, distintas predicciones y, por tanto, distintos valores de las métricas de evaluación del clasificador.*

→ *Subir o bajar el umbral hace que el clasificador sea más o menos estricto al predecir la clase positiva de una observación.*

EVALUACIÓN CLASIFICADORES BINARIOS QUE MANEJAN INCERTIDUMBRE: EJEMPLO

CLASIFICANDO SPAM (Kelleher, J.D., MacNamee, B., D'Arcy, A. Machine Learning for Predictive Data Analysis):

Supongamos que se ha diseñado un clasificador probabilístico para clasificar correos electrónicos en **SPAM (clase positiva)** o **HAM** (correo bueno, **clase negativa**). El clasificador se ha enfrentado a un conjunto de 20 correos, para los cuales sabemos si son SPAM o HAM. Los resultados son los que muestra la tabla:

ID	Clase	Pred.	Resultado	ID	Clase	Pred.	Resultado
1	spam	ham	FN	11	ham	ham	TN
2	spam	ham	FN	12	spam	ham	FN
3	ham	ham	TN	13	ham	ham	TN
4	spam	spam	TP	14	ham	ham	TN
5	ham	ham	TN	15	ham	ham	TN
6	spam	spam	TP	16	ham	ham	TN
7	ham	ham	TN	17	ham	spam	FP
8	spam	spam	TP	18	spam	spam	TP
9	spam	spam	TP	19	ham	ham	TN
10	spam	spam	TP	20	ham	spam	FP

- Para obtener la clase de predicción de cada muestra, el clasificador ha cuantificado la probabilidad de que el correo bajo consideración sea SPAM (clase positiva). Si esta probabilidad es $> 0,5$, entonces la clase de predicción es “SPAM”; en caso contrario es “HAM”.

$$TP = 6; FP = 2; TN = 9; FN = 3$$

- Para el umbral de aceptación aplicado de 0,5:

$$Acc = 0,75 ; P = 0,75; R = \frac{2}{3} ; F_1 = 0,706 ; FPR = \frac{2}{11}$$

¿ Cómo evoluciona el rendimiento del clasificador si variamos el umbral ?

¿ Cuál es el umbral de mayor rendimiento? ¿Cómo y qué criterios se aplican para su selección?

□ EVALUACIÓN CLASIFICADORES BINARIOS QUE MANEJAN INCERTIDUMBRE: EJEMPLO

CLASIFICANDO SPAM (Kelleher, J.D., MacNamee, B., D'Arcy, A. Machine Learning for Predictive Data Analysis):

¿ Cómo evoluciona el rendimiento del clasificador si variamos el umbral ?

¿ Cuál es el umbral de mayor rendimiento? ¿Cómo y qué criterios se aplican para su selección?

Para analizar estas cuestiones, debemos analizar las predicciones probabilísticas del clasificador (la tabla anterior mostraba directamente la predicción de clase para un umbral de aceptación de 0,5). La siguiente tabla muestra la probabilidad de pertenencia de cada correo a la clase positiva “SPAM” (se han ordenado las muestras en orden de probabilidad creciente):

ID	Clase	Predic. Prob.	ID	Clase	Predic. Prob.
7	ham	0.001	5	ham	0.302
11	ham	0.003	14	ham	0.348
15	ham	0.059	17	ham	0.657
13	ham	0.064	8	spam	0.676
19	ham	0.094	6	spam	0.719
12	spam	0.160	10	spam	0.781
2	spam	0.184	18	spam	0.833
3	ham	0.226	20	ham	0.877
16	ham	0.246	9	spam	0.960
1	spam	0.293	4	spam	0.963

- A partir de estas predicciones expresadas en términos de probabilidad de la clase positiva, es posible analizar el rendimiento del clasificador para distintos umbrales de aceptación.

EVALUACIÓN CLASIFICADORES BINARIOS QUE MANEJAN INCERTIDUMBRE: EJEMPLO

CLASIFICANDO SPAM (Kelleher, J.D., MacNamee, B., D'Arcy, A. Machine Learning for Predictive Data Analysis):

¿ Cómo evoluciona el rendimiento del clasificador si variamos el umbral ?

¿ Cuál es el umbral de mayor rendimiento? ¿Cómo y qué criterios se aplican para su selección?

Variando el umbral de predicción

Resultados de las predicciones de la clase para distintos valores de umbrales: 0,1, 0,25, 0,5, 0,75 y 0,9

- Cada valor de umbral refleja un punto de operación o modo de trabajo del clasificador, con sus valores propios de métricas de rendimiento.
- La aplicación del clasificador requiere la selección de un punto de operación (umbral de trabajo).
- Criterios de selección: basados en el análisis de curvas de rendimiento del clasificador en sus posibles puntos de operación (Curvas ROC, PR).

ID	Clase	Prob.	Pred. (0.10)	Pred. (0.25)	Pred. (0.50)	Pred. (0.75)	Pred. (0.90)
7	ham	0.001	ham	ham	ham	ham	ham
11	ham	0.003	ham	ham	ham	ham	ham
15	ham	0.059	ham	ham	ham	ham	ham
13	ham	0.064	ham	ham	ham	ham	ham
19	ham	0.094	ham	ham	ham	ham	ham
12	spam	0.160	spam	ham	ham	ham	ham
2	spam	0.184	spam	ham	ham	ham	ham
3	ham	0.226	spam	ham	ham	ham	ham
16	ham	0.246	spam	ham	ham	ham	ham
1	spam	0.293	spam	spam	ham	ham	ham
5	ham	0.302	spam	spam	ham	ham	ham
14	ham	0.348	spam	spam	ham	ham	ham
17	ham	0.657	spam	spam	spam	ham	ham
8	spam	0.676	spam	spam	spam	ham	ham
6	spam	0.719	spam	spam	spam	ham	ham
10	spam	0.781	spam	spam	spam	spam	ham
18	spam	0.833	spam	spam	spam	spam	ham
20	ham	0.877	spam	spam	spam	spam	ham
9	spam	0.960	spam	spam	spam	spam	spam
4	spam	0.963	spam	spam	spam	spam	spam
Tasa de acierto			0.700	0.700	0.750	0.700	0.650
Precisión (R)			0.600	0.637	0.750	0.800	1.000
Recall (R)			1.000	0.778	0.667	0.444	0.222
Tasa Falso Positivo (FPR)			0.545	0.364	0.182	0.091	0.000

□ EVALUACIÓN CLASIFICADORES BINARIOS QUE MANEJAN INCERTIDUMBRE: CURVAS ROC / PR

CURVAS DE RENDIMIENTO DE CLASIFICADOR BINARIO: representación gráfica de la variación de determinadas métricas de rendimiento con el umbral. Las más utilizadas en la práctica son:

➤ **CURVA ROC (*Receiver Operating Characteristic*):** representa la variación de la sensibilidad (*recall*) frente a la tasa de falsos positivos (*FPR*)

➤ **CURVA PR (*Precision/Recall*):** representa la variación de la precisión (*precision*) frente a la sensibilidad (*recall*)

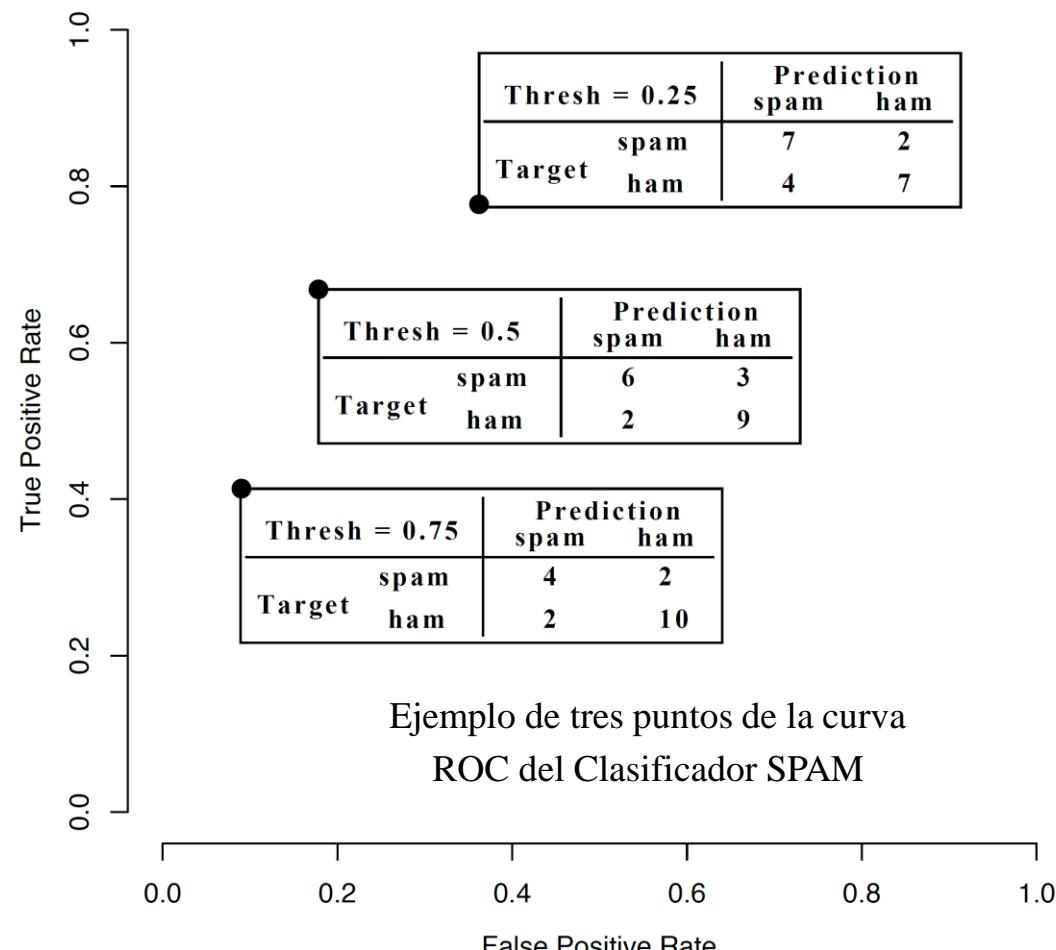
→ Las curvas muestran simultáneamente los dos tipos de errores (positivos y negativos) para todos los umbrales posibles:

- La sensibilidad está asociada con la tasa de falsos negativos

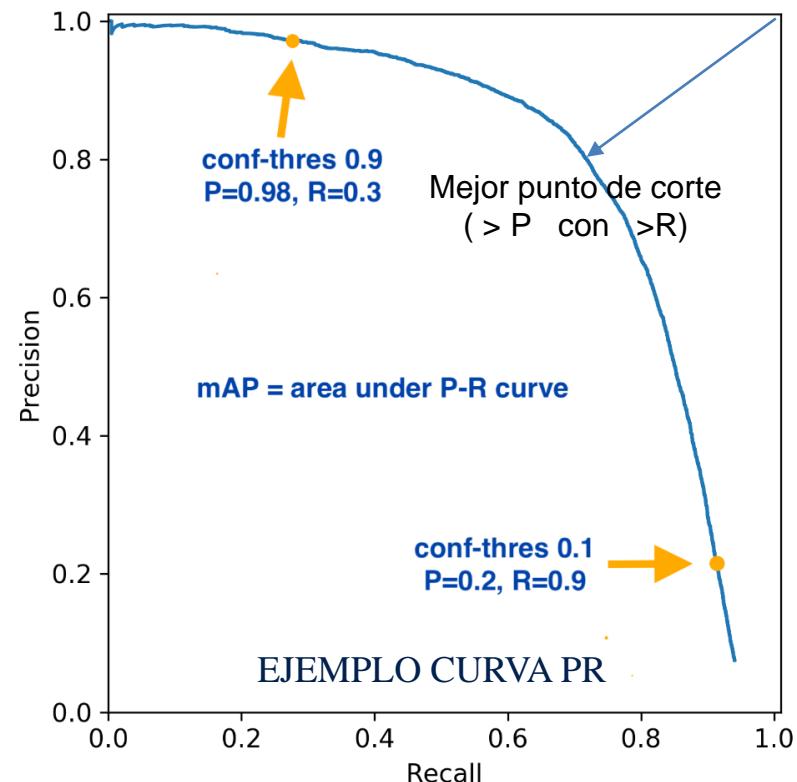
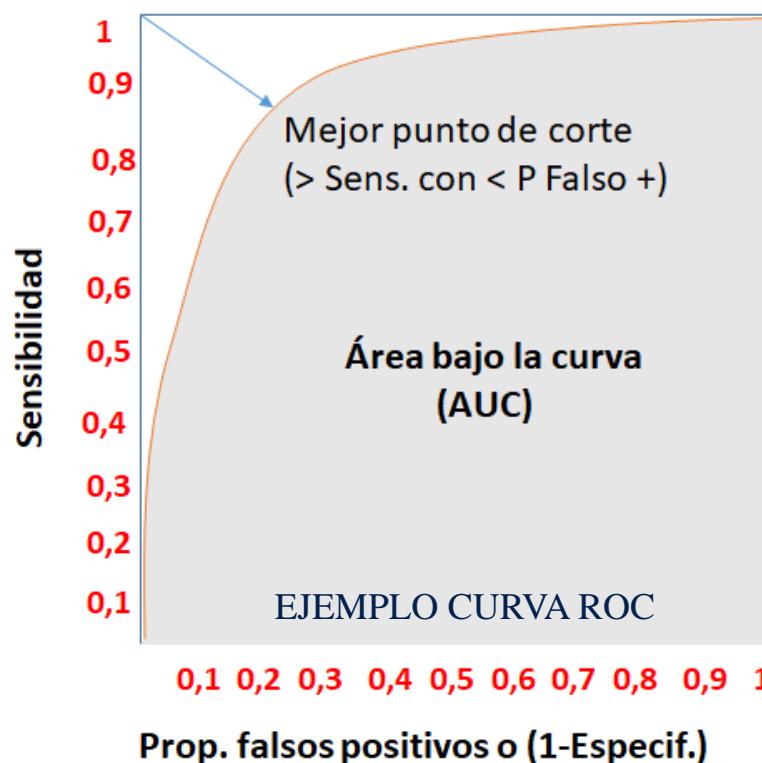
$$TFN = 1-R$$

- La precisión es una métrica indicadora de los falsos positivos

$$P = TP/(TP+FP)$$



□ EVALUACIÓN CLASIFICADORES BINARIOS QUE MANEJAN INCERTIDUMBRE: CURVAS ROC / PR



→ Las curvas permiten seleccionar el punto de operación en base al compromiso entre ambas métricas:

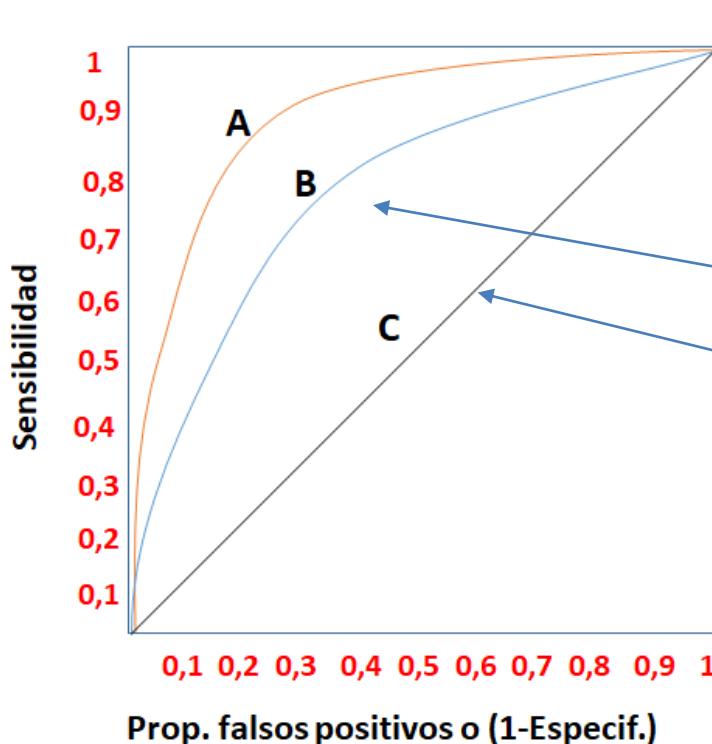
- **CURVA ROC:** punto de la curva más cerca del punto $(0, 1)$ (punto de rendimiento ideal: $Se = R = 1$ ($TFN = 0$) con $FPR = 0$ ($Sp = 1$)).
- **CURVA PR:** punto más cercano a la esquina superior derecha $(1, 1)$: valores de P y R altos simultáneamente.

EVALUACIÓN CLASIFICADORES BINARIOS QUE MANEJAN INCERTIDUMBRE: CURVAS ROC / PR

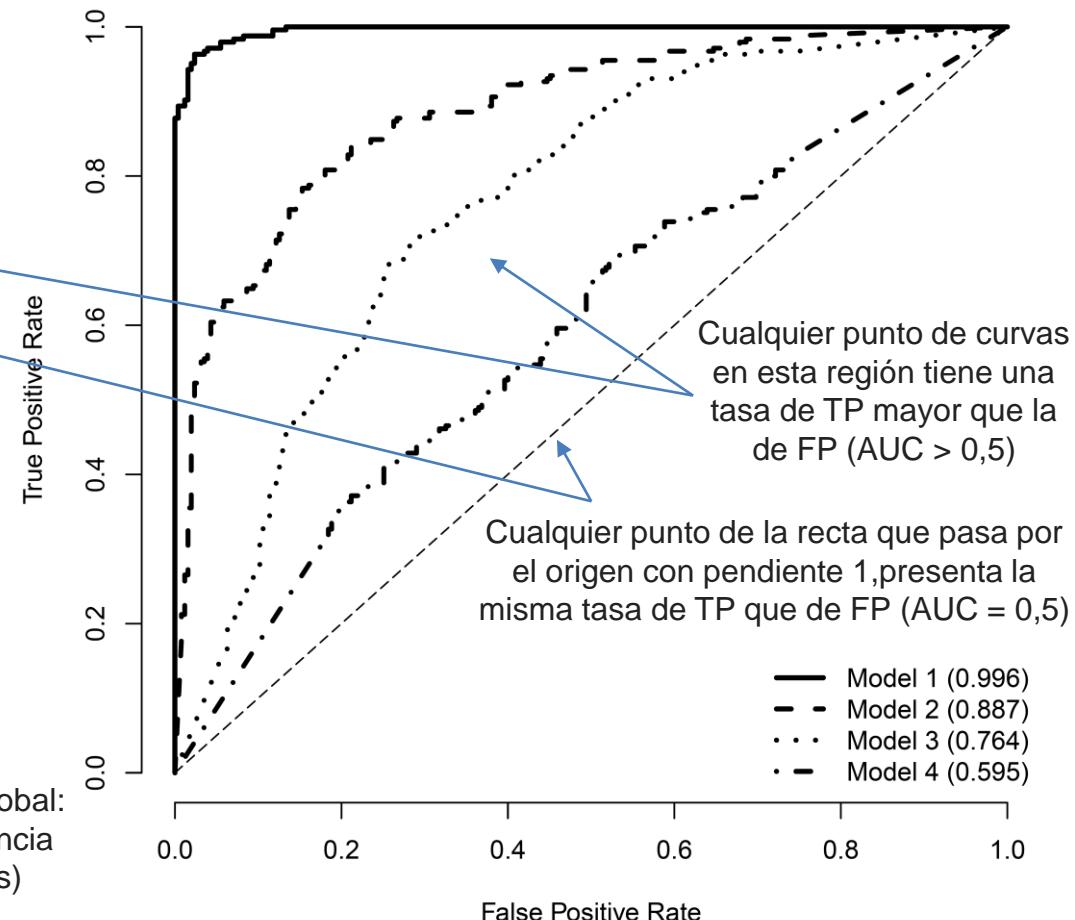
Las curvas permiten comparar el rendimiento de clasificadores, a través de la métrica:

❖ **AREA BAJO LA CURVA (AUC, Area Under Curve) ROC o PR:**

→ Métrica que permite medir el rendimiento global de un clasificador, resumido en todos sus posibles puntos de operación. Valores entre 0 y 1: el rendimiento es mayor cuando $AUC \rightarrow 1$



Los modelos A y 1 son los de mayor rendimiento global: valor de AUC más próximo a 1 (mayor es la diferencia entre la tasa de TP y FP en los diferentes puntos)



□ EVALUACIÓN CLASIFICADOR MULTICLASE

❖ **Matriz de confusión en un problema de n clases:**

		Predicted			
		C ₁	C ₂	...	C _n
Actual	C ₁	N ₁₁	N ₁₂	...	N _{1n}
	C ₂	N ₂₁	N ₂₂	...	N _{2n}
	:	:	:	:	:
	C _n	N _{n1}	N _{n2}	...	N _{nn}

❖ **Métricas de evaluación de un clasificador multiclas:**

- Procedimiento de cálculo: supone una extensión del cálculo de las métricas de rendimiento vistas para clasificación binaria a problemas de clasificación que involucra más de 2 clases:
 1. **Cálculo de métricas asociadas a cada clase del problema:** las métricas asociadas a una determinada clase se calculan considerando un problema de clasificación binaria donde las muestras de la clase positiva son las de la clase en cuestión y las de la clase negativa, son las muestras del resto de las clases.
 2. **Cálculo de métricas de rendimiento del clasificador:** Las métricas asociadas al rendimiento del clasificador se calculan promediando el valor obtenido para cada clase del problema, o haciendo un promedio ponderado según el número de instancias de cada clase:

$$MR = \frac{\sum_{i=1}^n MR_i}{n} ; \text{ weighted } MR = \frac{\sum_{i=1}^n |C_i| * MR_i}{\sum_{i=1}^n |C_i|}$$

MR_i : Métrica de rendimiento de la clase i

$|C_i|$: N° instancias de la clase i

□ EVALUACIÓN CLASIFICADOR MULTICLASE: EJEMPLO.

predicted → real ↓	Class_1	Class_2	Class_3
Class_1	94	16	10
Class_2	21	113	16
Class_3	4	4	92

Clase Real	Matriz de Confusión Clase 1		Clase Predicha	
	Clase 1 (+)	Clases 2-3 (-)	Clase 1 (+)	Clases 2-3 (-)
Clase Real	Clase 1 (+)	94	26	
	Clases 2-3 (-)	25	225	

Matriz de Confusión Clase 2		Clase Predicha	
		Clase 2 (+)	Clases 1-3 (-)
Clase Real	Clase 2 (+)	113	37
	Clases 1-3 (-)	20	200

Clase Real	Matriz de Confusión Clase 3		Clase Predicha	
	Clase 3 (+)	Clases 1-2 (-)	Clase 3 (+)	Clases 1-2 (-)
Clase Real	Clase 3 (+)	92	8	
	Clases 1-2 (-)	26	244	

$$Acc = \frac{Nº Aciertos Clasificación}{Nº observaciones} = \frac{TP + TN}{TP + FN + FP + TN} \rightarrow \begin{cases} Acc_1 = 0,8622 \\ Acc_2 = 0,8459 \\ Acc_3 = 0,9081 \end{cases} \rightarrow \begin{cases} Acc = 0,8721 \\ wAcc = 0,8680 \end{cases}$$

$$Se = \frac{Nº Aciertos Clase +}{Nº Observaciones Clase +} = \frac{TP}{TP + FN} \rightarrow \begin{cases} Se_1 = 0,7833 \\ Se_2 = 0,7533 \\ Se_3 = 0,9200 \end{cases} \rightarrow \begin{cases} Se = 0,8189 \\ wSe = 0,8081 \end{cases}$$

$$Sp = \frac{Nº Aciertos Clase -}{Nº Observaciones Clase -} = \frac{TN}{FP + TN} \rightarrow \begin{cases} Sp_1 = 0,9000 \\ Sp_2 = 0,9091 \\ Sp_3 = 0,9037 \end{cases} \rightarrow \begin{cases} Sp = 0,9043 \\ wSp = 0,9047 \end{cases}$$

FUENTES CONSULTADAS

- “Razonamiento asistido por computador”, Máster en Lógica, Computación e Inteligencia Artificial, 2017-18, José Luis Ruiz Reina, Universidad de Sevilla, Recurso disponible: <https://www.cs.us.es/cursos/rac-2017/>
- “Aprendizaje automático para el análisis de datos”, Grado en Estadística y Empresa, Ricardo Aler, Universidad Carlos III de Madrid. OpenCourseWare: <http://ocw.uc3m.es/ingenieria-informatica/aprendizaje-automatico-para-el-analisis-de-datos>