

Minería de Datos

Tema 2.- El proceso del descubrimiento del conocimiento a partir de bases de datos (KDD)

M. Julia Flores & José A. Gámez

Departamento de Informática

Universidad de Castilla-La Mancha

Escuela Superior de Ingeniería Informática - ESII

El proceso del descubrimiento del conocimiento a partir de bases de datos (KDD)

1. Introducción al KDD
2. Fases en el proceso de KDD
3. Técnicas de Minería de Datos

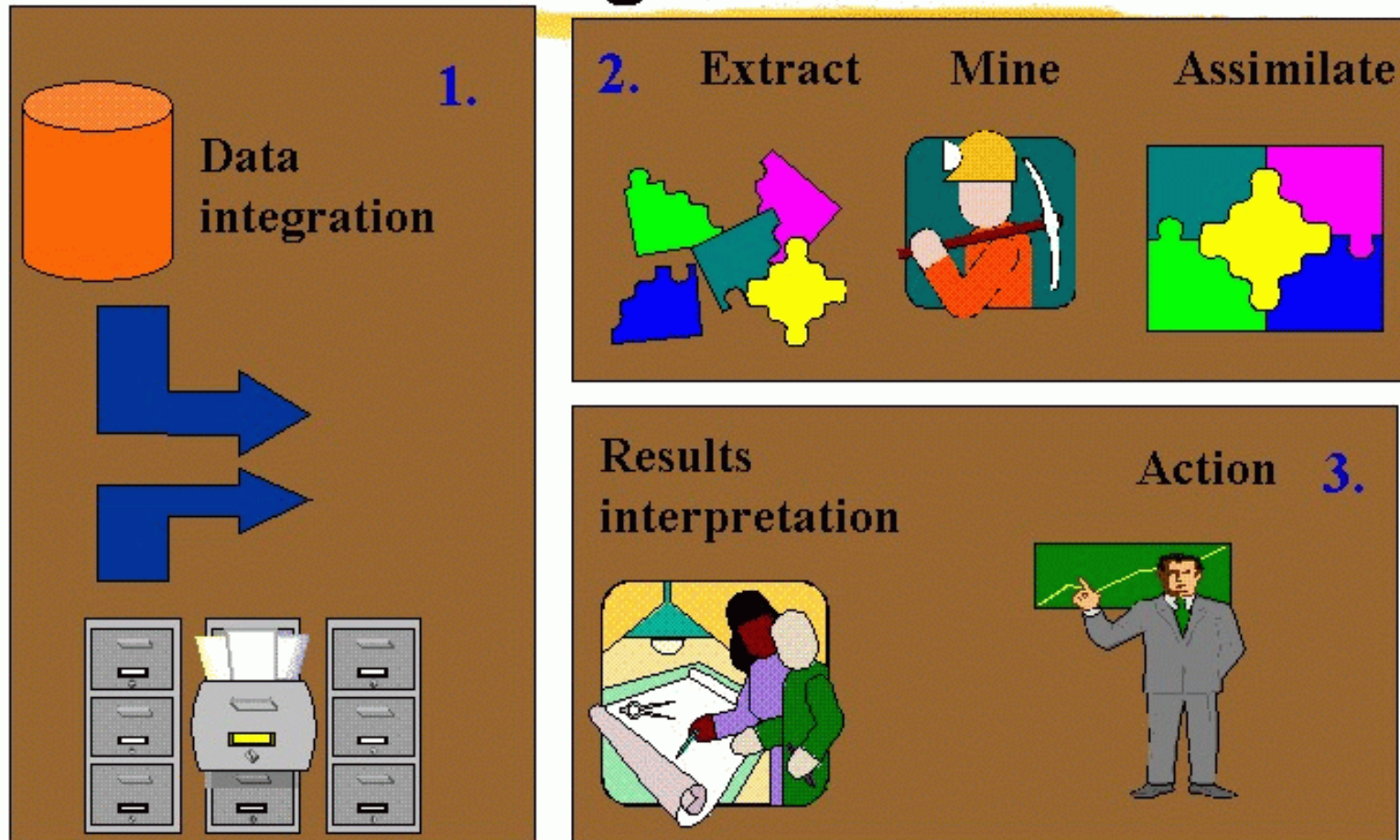
¿Qué es KDD?

Descubrimiento del conocimiento de bases de datos (KDD)

KDD Knowledge Discovery from Databases

- El KDD es el proceso completo de extracción del conocimiento a partir de bases de datos
- El término fue acuñado en 1989 para enfatizar que el conocimiento es el producto final de un proceso de descubrimiento guiado por los datos
- Minería de Datos es sólo un paso en el proceso de KDD
- Informalmente, Minería de Datos \simeq KDD

Data Mining Process



KDD como un proceso (II)

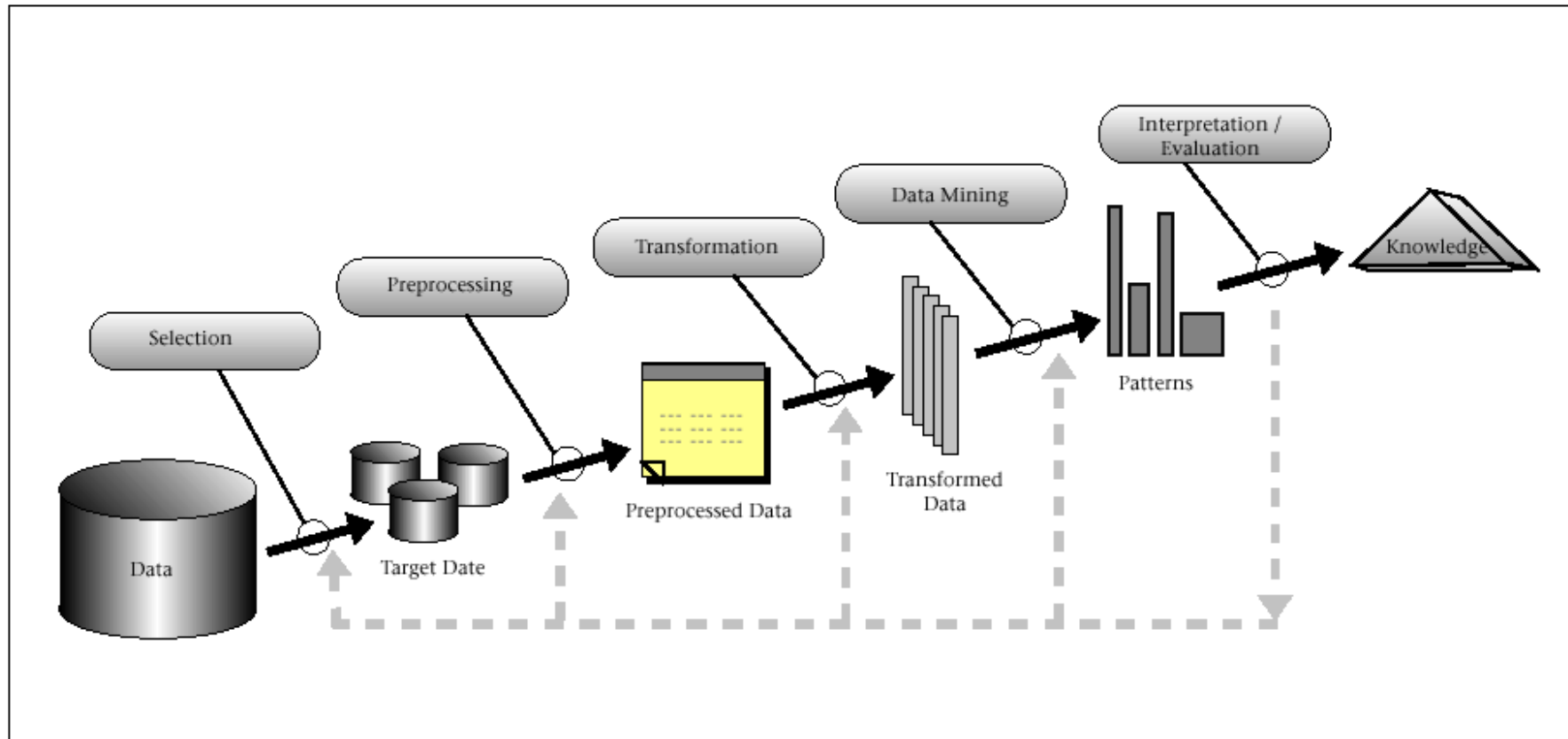
- **KDD** está enfocado al proceso global de descubrimiento del conocimiento a partir de bases de datos. Entre otros aspectos incluye:
 - ▶ cómo son almacenados y accedidos los datos,
 - ▶ cómo pueden escalarse los algoritmos para trabajar con cantidades de datos enormes y seguir siendo eficientes,
 - ▶ cómo pueden interpretarse y visualizarse los resultados,
 - ▶ cómo modelar y dar soporte a la interacción hombre-máquina durante todo el proceso.
- KDD hace especial énfasis en la búsqueda de modelos/patrones comprensibles
- También es importante la robustez frente a grandes conjuntos de datos ruidosos

KDD como un proceso (III)

El proceso del KDD contiene:

- El uso de la base de datos junto con cualquier operación de selección, preprocesamiento, (sub)muestreo, y transformación de la misma
- Algoritmos para obtener patrones/modelos a partir de los datos (¡¡MD en sentido estricto!!)
- Evaluación del resultado de los algoritmos y seleccionar aquellos modelos que puedan considerarse conocimiento

El ciclo del KDD



Etapas/Fases en el proceso de KDD

- 1 Comprensión del dominio de aplicación del problema, identificación de conocimiento a priori y creación del DataWarehouse.
- 2 Selección de datos, limpieza y preprocesamiento.
- 3 Criba de datos: Reducción y proyección.
- 4 Selección de la técnica de MD y aplicación de algoritmos concretos de MD.
- 5 Evaluación, interpretación y presentación de los resultados obtenidos.
- 6 Difusión y uso del nuevo conocimiento.

Fase 1. Dominio del problema y DW

- ¿Realmente estamos ante un problema adecuado para KDD?
- La familiarización con el dominio y la obtención de conocimiento a priori disminuirá el espacio de soluciones posibles => más eficiencia en el resto del proceso
- Unificación de la información en un DataWarehouse a partir de:
 - ▶ Información interna: distintas BBDD diseñadas para trabajo transaccional y de otro tipo (hojas de cálculo, informes,...)
 - ▶ Estudios publicados (demografía, catálogos, páginas,...)
 - ▶ Otras bases de datos (compradas, industrias/empresas afines, ...)
- Sin duda, el resto del proceso será más cómodo si la fuente de datos está **unificada**, es **accesible** (interna) y **dedicada** (desconectada del trabajo transaccional).

Fase 2. Selección, limpieza y preprocesamiento

- A partir del resultado de la fase anterior, mediante navegación por el DataWarehouse y a partir de análisis y visualizaciones previas, **seleccionar el conjunto de datos** adecuado para el resto del proceso.
- **Limpieza de datos (*Data Cleaning*)**, consiste en rellenar valores perdidos, identificar y/o eliminar valores anómalos (*outliers*), suavizar el ruido, eliminar inconsistencias (DW)
- **Preprocesamiento**: transformación de los datos, variables, valores, ...

Limpieza de datos: *data cleaning*

- **Datos perdidos (*missing*):** Rellenarlos manualmente, Ignorarlos, eliminar la fila/columna, usar un valor especial p.e. `unknown`, inferirlos usando técnicas estadísticas, ...
- **Datos anómalos (*outliers*):** Primero hay que identificarlos, y después el tratamiento es parecido al caso anterior, sólo que el valor puede darnos alguna idea.
- **Ruido:** error aleatorio o siguiendo una varianza en los datos. El tratamiento básico es suavizar mediante técnicas estadísticas (*binning*, regresión, ...)
- **Inconsistencias:** registros duplicados, datos inconsistentes, ... Normalmente ya tratado en la elaboración del DW.

Preprocesamiento/transformación

- **Redefinición de los atributos** mediante agrupamiento o separación.
- **Transformación de los atributos**: fecha nacimiento → edad, apellidos → enteros, ...

En ocasiones => almacenar meta-información sobre la información **realmente** almacenada por cada campo.

- **Discretización**. Pasar atributos continuos (o discretos con muchos valores) a casos discretos manejables.

Hay diversas técnicas.

Es imprescindible para muchos algoritmos de MD.

Fase 3. Criba de datos

- **Reducción de casos/filas:** Las técnicas usadas van desde la compresión al muestreo de los datos, pasando por la elección de representantes (*clustering*)
- **Proyección:** Seleccionar el conjunto de atributos adecuado para la tarea específica a realizar.

Suele conocerse también como Selección de variables (*feature subset selection*).

Las técnicas a emplear son: estadísticas, basadas en búsqueda combinadas con métodos empíricos, ...

Fase 4. Minería de Datos

- ¿Qué tipo de conocimiento buscamos? Predictivo o descriptivo.
- ¿Qué técnica es la más adecuada? Clasificación, regresión (predicción numérica), clustering/agrupamiento/segmentación, asociaciones, ...
- ¿Es necesaria la incertidumbre en el modelo resultante? Certeza, probabilidad, lógica difusa, ...
- ¿Qué algoritmo es el más adecuado? P.e. en clustering, *duro*, difuso, jerarquizado; k-means, iterativo, EM, ...

Fase 5. Evaluación, interpretación, ...

- La fase de MD puede producir varias hipótesis de modelos
- Será necesario establecer qué modelos son los más válidos (técnicas habituales son el uso de conjuntos de **test** independientes, ...)
- La interpretación de los mejores modelos (visualización, simplicidad, posibilidad de integración, ventajas colaterales, ...) ayudará a la selección del modelo(s) final(es)

Fase 6. Difusión y uso del nuevo conocimiento

- Elaboración de informes para su distribución
- Usar el nuevo conocimiento de forma independiente
- Incorporarlo a sistemas ya existentes (chequear con el conocimiento ya usado para evitar inconsistencias y posibles conflictos)

La monitorización del sistema en acción dará lugar a nuevos casos que realimentarán el ciclo del KDD.

Las condiciones iniciales pueden variar, invalidando el modelo adquirido.

Técnicas de Minería de Datos

Un **Algoritmo de Minería de Datos** es un procedimiento **bien definido** que toma datos como entrada y produce modelos o patrones como salida.

Un algoritmo de minería de datos puede ser especificado de forma **sistemática** mediante la definición de cinco componentes:

- Tarea
- Estructura del modelo/patrón
- Función objetivo
- Método de búsqueda/optimización
- Técnica de manejo de los datos

Visión sistemática de los alg. de MD (i)

- **Tarea.** Se trata de identificar el tipo de problema a abordar con el algoritmo de MD (clasificación, visualización, clustering,...)
- **Estructura.** Describir el modelo a *aprender*, es decir, cuál será el patrón o modelo que intentaremos descubrir para que represente a los datos (árboles, reglas, ecuaciones, gráficos, ...)
- **Función objetivo.** Es el criterio que intentaremos optimizar durante el proceso de minería y nos medirá la bondad de los modelos encontrados respecto a los datos.

Puede estar basado únicamente en bondad-de-ajuste o por el contrario puede intentar **capturar** generalización.

Visión sistemática de los alg. de MD (ii)

- **Método de búsqueda/optimización.** Es el tipo de método que se usará en el intento de que el patrón obtenido optimice la función objetivo (métodos voraces, heurísticos, probabilísticos, ...)

Dependiendo de si la estructura es fija o no tendremos que realizar **aprendizaje estructural** y/o **aprendizaje paramétrico**.

- **Técnica de manejo de los datos.** Describir la técnica a usar para el almacenamiento, indexado y recuperación de los datos.

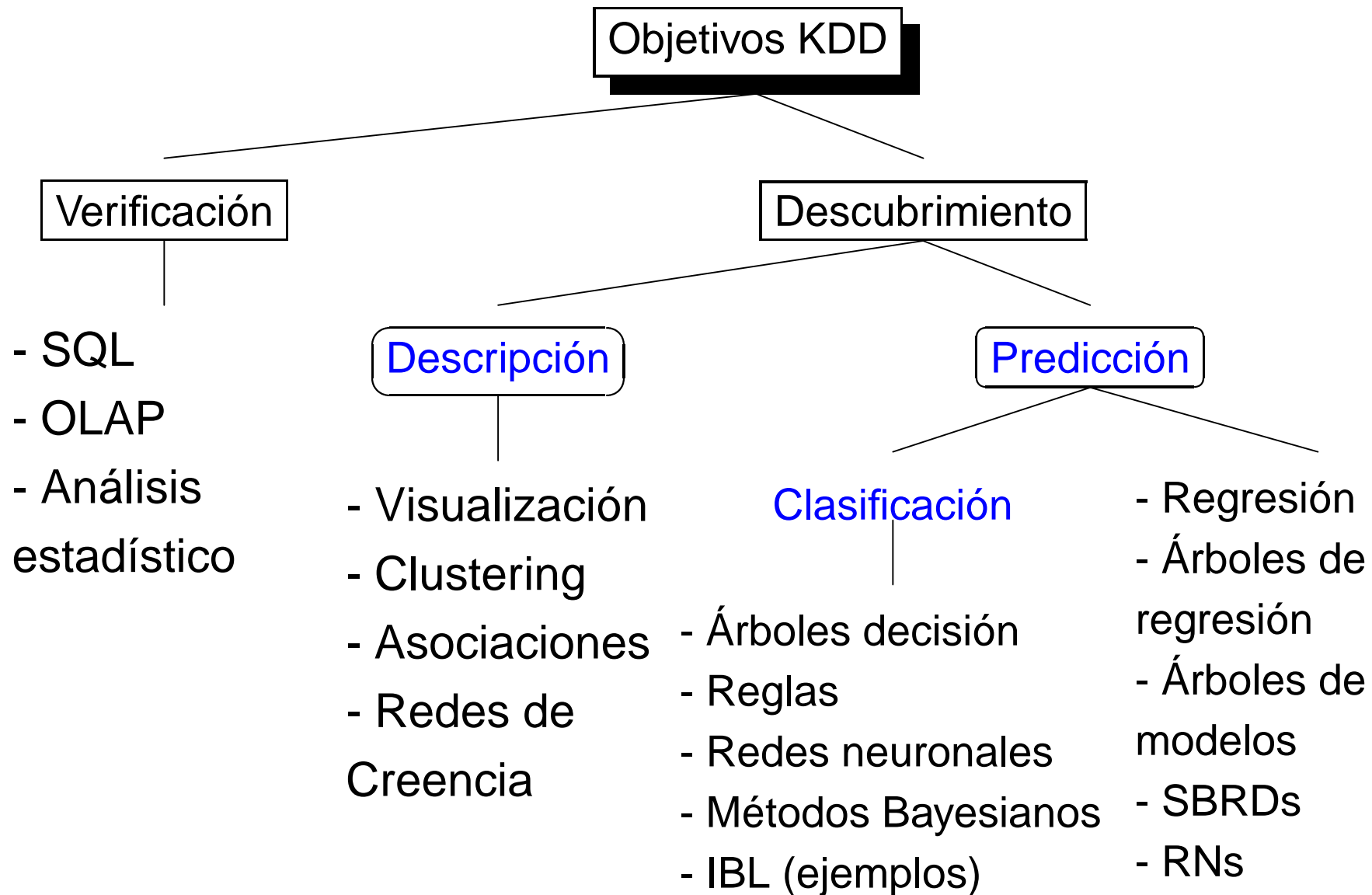
La mayoría de los métodos de aprendizaje automático obvian este paso, debido a que asumen que el volumen de datos es lo suficientemente pequeño para estar en memoria principal.

Al trabajar con grandes volúmenes de datos, este apartado es muy importante.

Visión sistemática de los alg. de MD (iii)

	ID3	RNs-Backprop.	A priori
Tarea	Clasificación	Regresión/ clasificación	Descubrimiento de reglas
Estructura	Árbol de decisión	red neuronal	reglas de asociación
Función objetivo	ganancia de información	error cuadrático	soporte/ confianza
Método de búsqueda	voraz	gradiente descendiente	primero-mejor + poda
Manejo de los datos	??	??	lecturas secuenciales

Taxonomía de técnicas de Minería de Datos

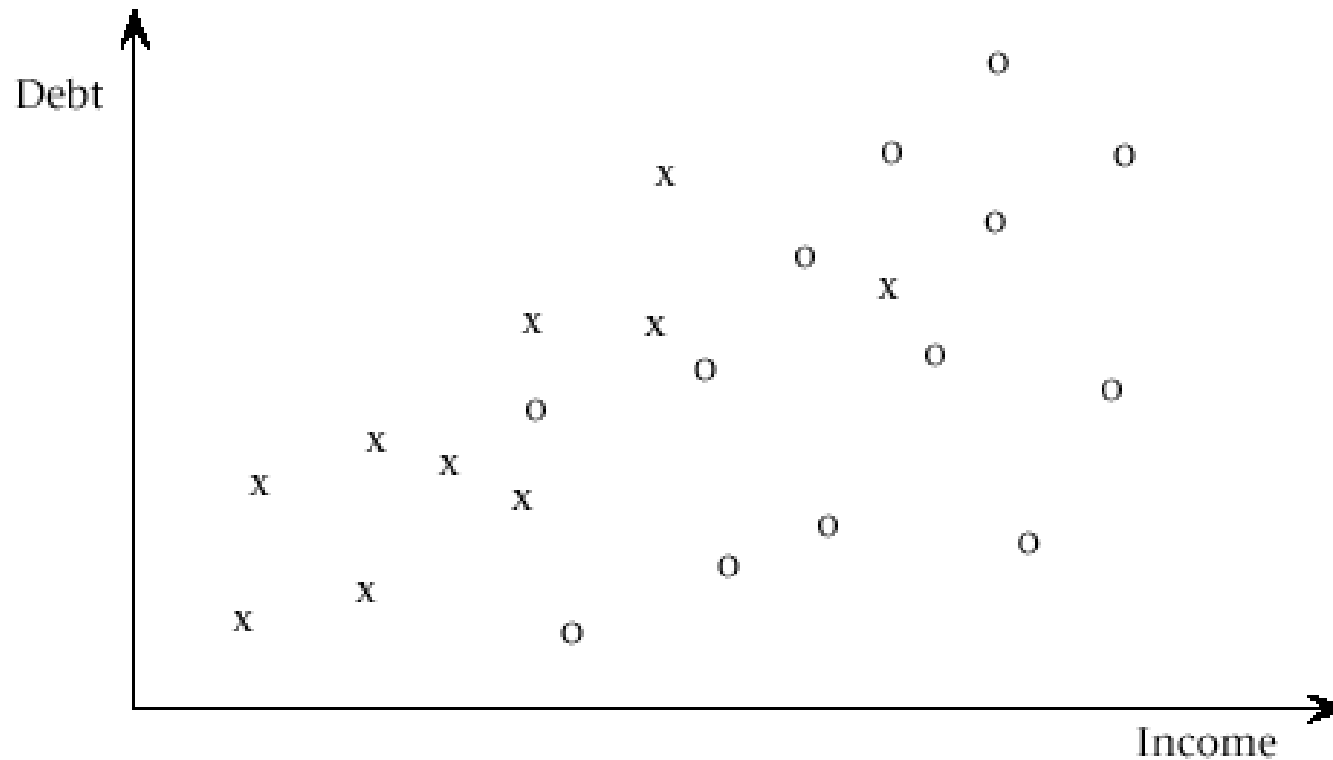


Clasificación

Clasificación: A partir de los valores que toman ciertos atributos o variables predictoras, decidir la pertenencia a una de las posibles clases.

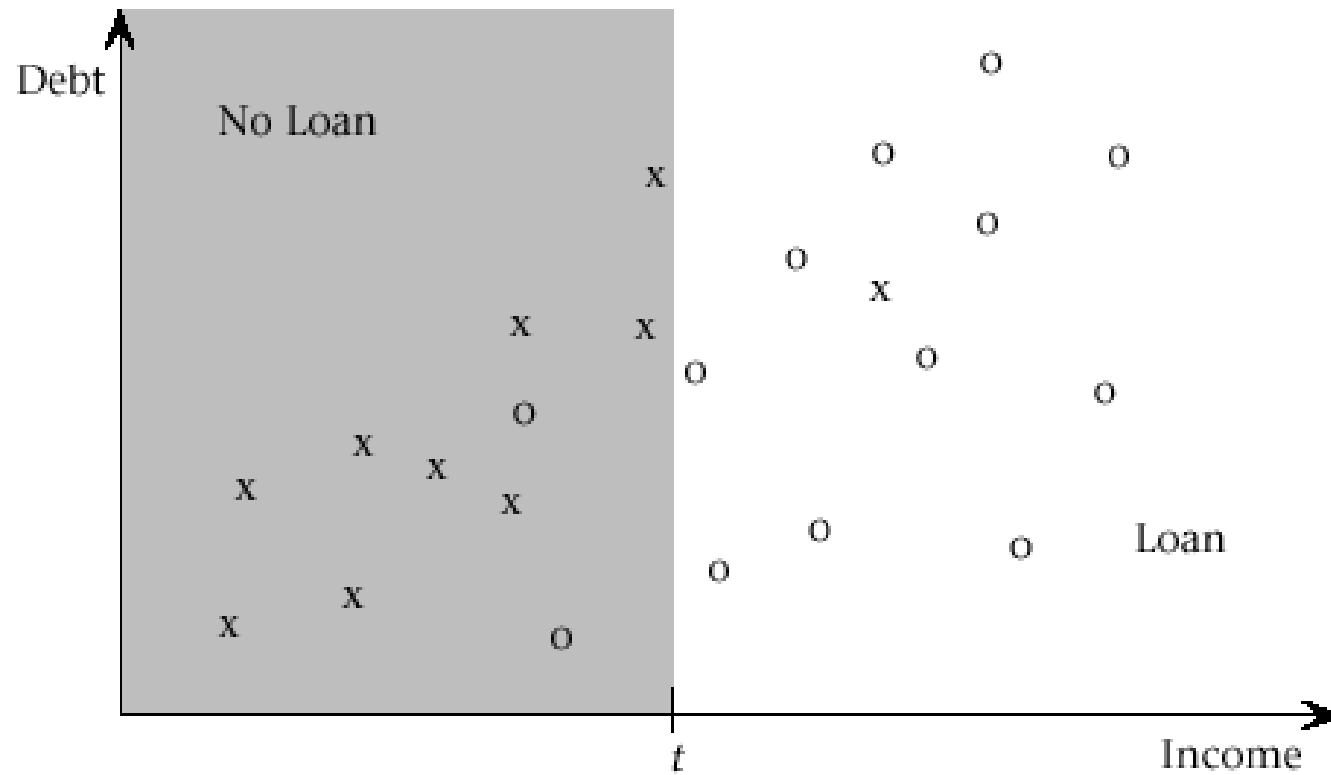
- Umbrales
- Clasificadores lineales
- Árboles de clasificación
- Clasificadores basados en reglas
- Clasificadores Bayesianos
- Redes Neuronales
- Métodos basados en ejemplos (IBL)

Ejemplo de clasificación



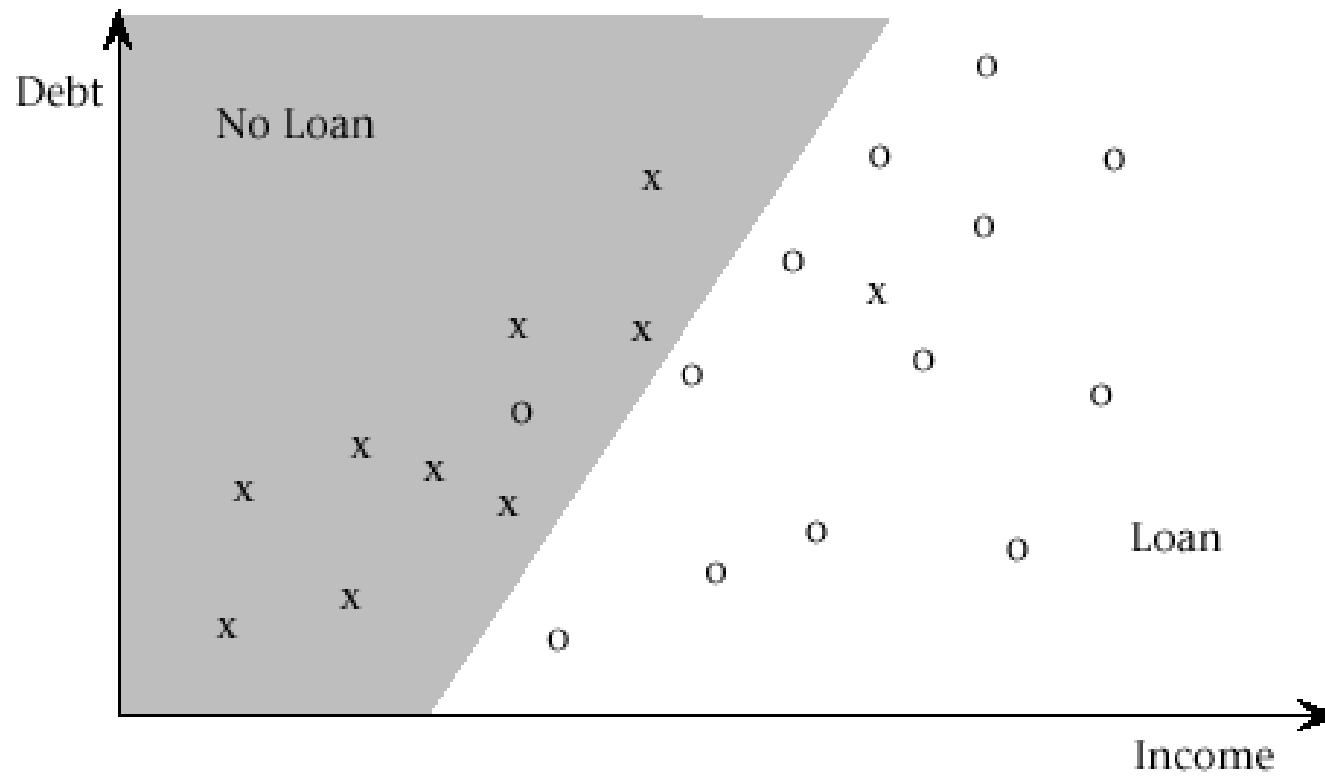
Ejemplo de problemas con dos atributos (deuda e ingresos) y dos clases prestar (o) y no prestar (x)

Ejemplo de clasificación (II)

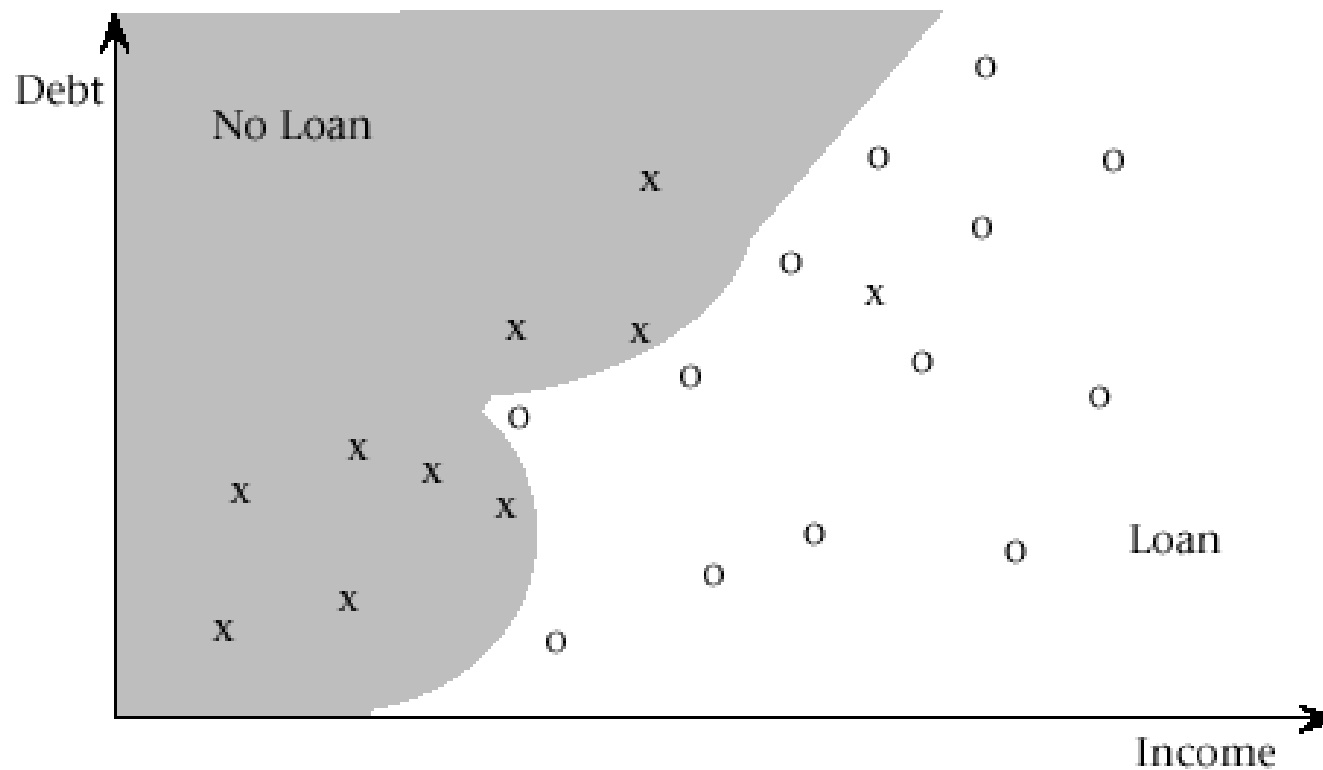


Clasificación usando un umbral (3 errores)

Ejemplo de clasificación (III)

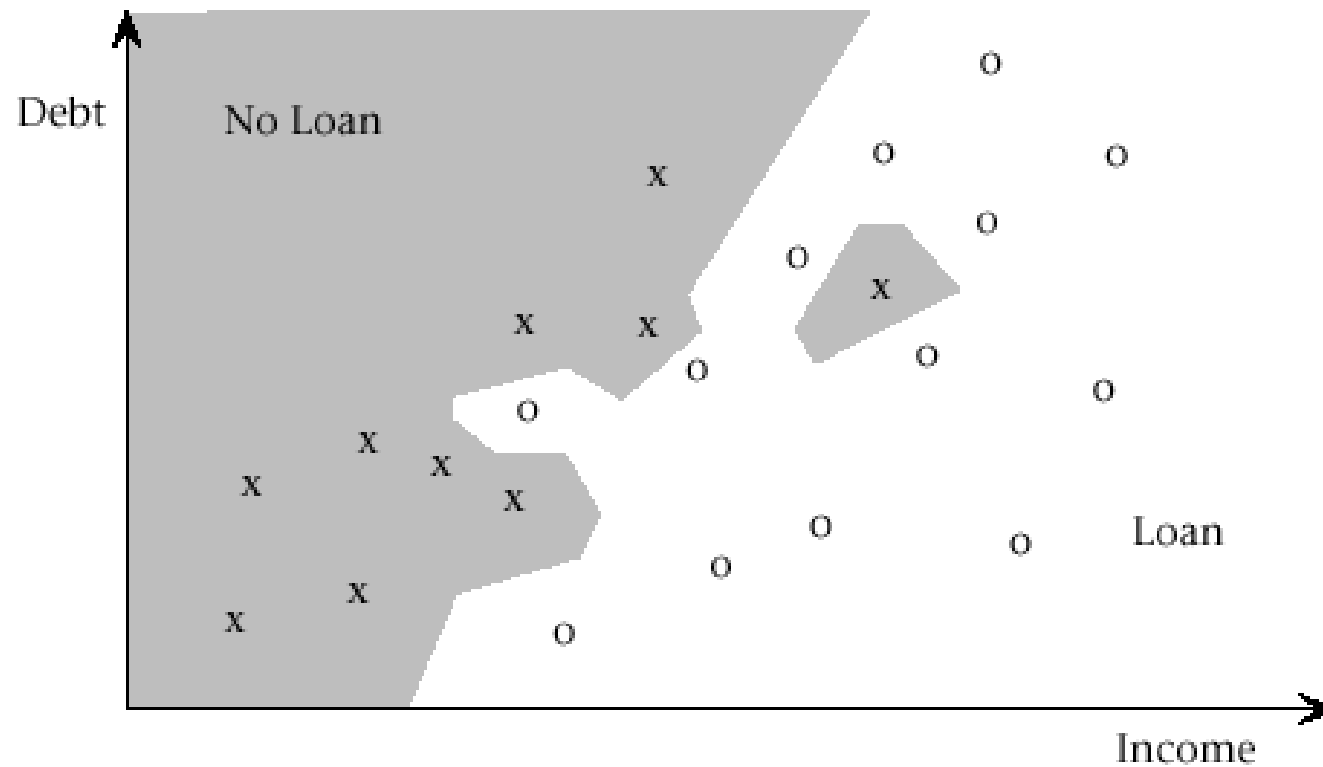


Ejemplo de clasificación (IV)



Clasificación no lineal, p.e. Red Neuronal Multicapa. (1 error)

Ejemplo de clasificación (V)



Clasificación basada en ejemplos, p.e. k-vecinos más próximos, (0 errores)

Árboles de Decisión/Clasificación

outlook	temperature	humidity	windy	play
sunny	hot	high	FALSE	no
sunny	hot	high	TRUE	no
overcast	hot	high	FALSE	yes
rainy	mild	high	FALSE	yes
rainy	cool	normal	FALSE	yes
rainy	cool	normal	TRUE	no
overcast	cool	normal	TRUE	yes
sunny	mild	high	FALSE	no
sunny	cool	normal	FALSE	yes
rainy	mild	normal	FALSE	yes
sunny	mild	normal	TRUE	yes
overcast	mild	high	TRUE	yes
overcast	hot	normal	FALSE	yes
rainy	mild	high	TRUE	no

Árboles de clasificación: ID3

```
outlook = sunny
|  humidity = high: no
|  humidity = normal: yes
outlook = overcast: yes
outlook = rainy
|  windy = TRUE: no
|  windy = FALSE: yes
```

Number of Leaves : 5

Size of the tree : 8

Árboles de clasificación: C4.5

outlook	temperature	humidity	windy	play
sunny	85	85	FALSE	no
sunny	80	90	TRUE	no
overcast	83	86	FALSE	yes
rainy	70	96	FALSE	yes
rainy	68	80	FALSE	yes
rainy	65	70	TRUE	no
overcast	64	65	TRUE	yes
sunny	72	95	FALSE	no
sunny	69	70	FALSE	yes
rainy	75	80	FALSE	yes
sunny	75	70	TRUE	yes
overcast	72	90	TRUE	yes
overcast	81	75	FALSE	yes
rainy	71	91	TRUE	no

Árboles de clasificación: C4.5

```
outlook = sunny
|   humidity <= 75: yes (2.0)
|   humidity > 75: no (3.0)
outlook = overcast: yes (4.0)
outlook = rainy
|   windy = TRUE: no (2.0)
|   windy = FALSE: yes (3.0)
```

Number of Leaves : 5

Size of the tree : 8

Reglas

Para el árbol obtenido por C4.5:

Rule 1: IF outlook = overcast
THEN class Yes [70.7%]

Rule 2: IF outlook = rainy AND windy = false
THEN class Yes [63.0%]

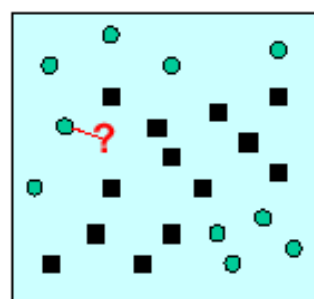
Rule 3: IF outlook = sunny AND humidity > 75
THEN class No [63.0%]

Rule 4: IF outlook = rainy AND windy = true
THEN class No [50.0%]

Default class: Yes

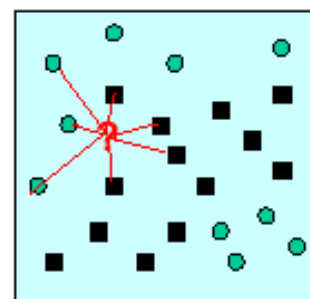
Métodos basados en ejemplos

- La propia base de datos (ejemplos) constituye el modelo
- Las propiedades (clase) de un nuevo ejemplo, se obtienen de las propiedades de ejemplos similares.
- Ejemplos: Método del/os vecino(s) más próximo(s), razonamiento basado en casos



1-nearest neighbor

Clasifica
círculo

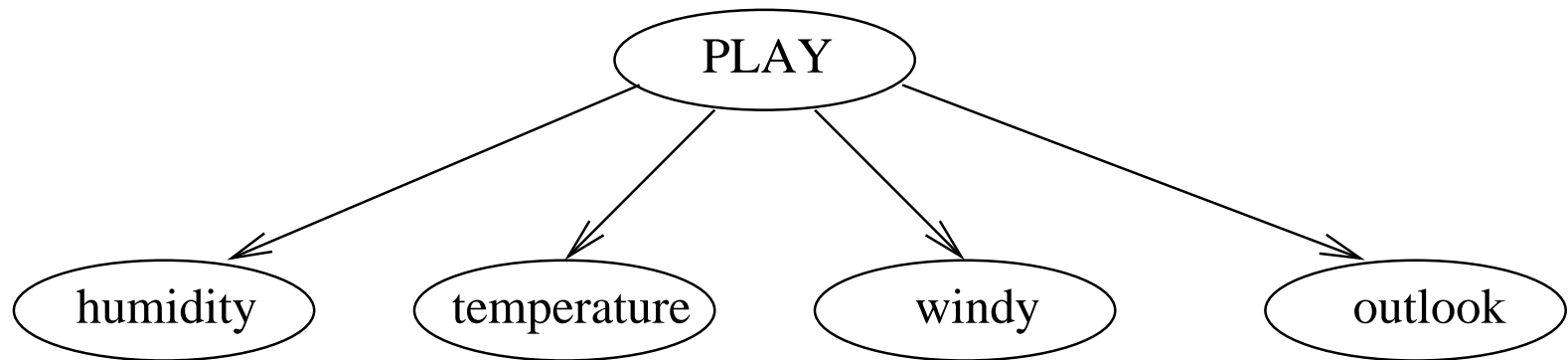


7-nearest neighbor

Clasifica
cuadrado

Clasificadores Bayesianos

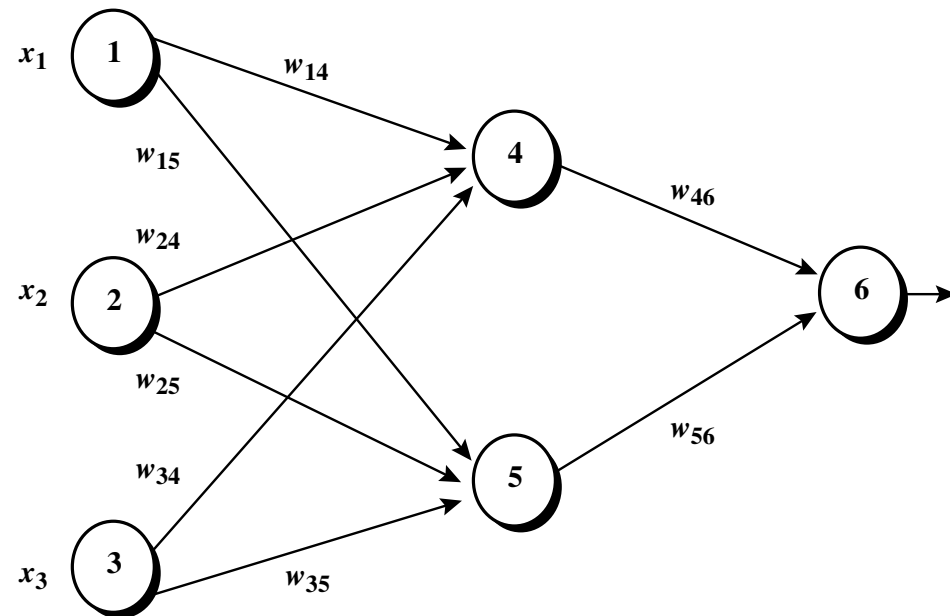
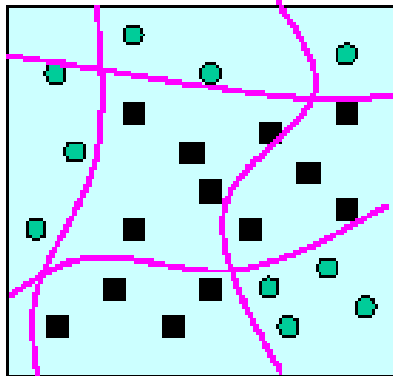
- Basados de una forma u otra en el teorema de Bayes
- Los hay desde muy sencillos (Naive Bayes) hasta muy sofisticados (TAN, basados en redes Bayesianas, ...)
- **Naive Bayes**: Los atributos se suponen independientes dada la clase



Redes Neuronales

- Perceptrón (sin capas ocultas) \Rightarrow clasificadores lineales
- Redes Neuronales Multicapa. Las unidades internas permiten

particiones no lineales.



Regresión

Regresión es la denominación general aplicada a los métodos cuya variable *clase* es continua.

Se trata de **aproximar el valor numérico** de dicha variable, conociendo los valores del resto de atributos.

Posibles aplicaciones:

- Estimación de bio-masa en un bosque a partir de fotos via satélite,
- Estimación de probabilidades de supervivencia ante determinados diagnósticos o tratamientos,
- Estimación de la demanda de cierto producto en función de clientes, fechas, ...
- Estimación de consumos, rendimientos, ...

Ejemplo: CPU

Vendor	MCYT	Memoria RAM		Cache	Canales		Rendimiento
		MMIN	MMAX	CACH	CHMIN	CHMAX	PRP
dec	133	1000	12000	9	3	12	54
dec	133	1000	8000	9	3	12	41
dg	700	384	8000	0	1	1	34
dg	700	256	2000	0	1	1	19
hp	90	256	1000	0	3	10	18
hp	105	256	2000	0	3	10	20
ibm	57	4000	24000	64	12	16	171
ibm	26	16000	32000	64	16	24	361
.....							
ncr	56	2000	8000	0	1	8	41
nixdorf	200	1000	2000	0	1	2	21
siemens	240	512	2000	8	1	5	22
siemens	105	2000	4000	8	3	8	31

Regresión Lineal

Se trata de **obtener una expresión/ecuación** que prediga la cantidad numérica.

P.e. para el problema de la CPU (excluyendo el vendedor):

$$\text{PRP} = -55.9 + 0.0489 \cdot \text{MCYT} + 0.0153 \cdot \text{MMIN} + 0.0056 \cdot \text{MMAX} + 0.6410 \cdot \text{CACH} \\ - 0.2700 \cdot \text{CHMIN} + 1.480 \cdot \text{CHMAX}$$

Regresión con árboles

- Un **árbol de regresión** es un árbol de decisión cuyas hojas predicen una cantidad numérica

El valor de predicción es la media de los ejemplos que llegan a cada hoja.

- Un **árbol de modelos** es un árbol de regresión que contiene una expresión de regresión lineal en cada hoja.

Permite aproximar funciones continuas.

Árbol de regresión para CPU

PRP =

- 56.1

+ 0.049 MYCT

+ 0.015 MMIN

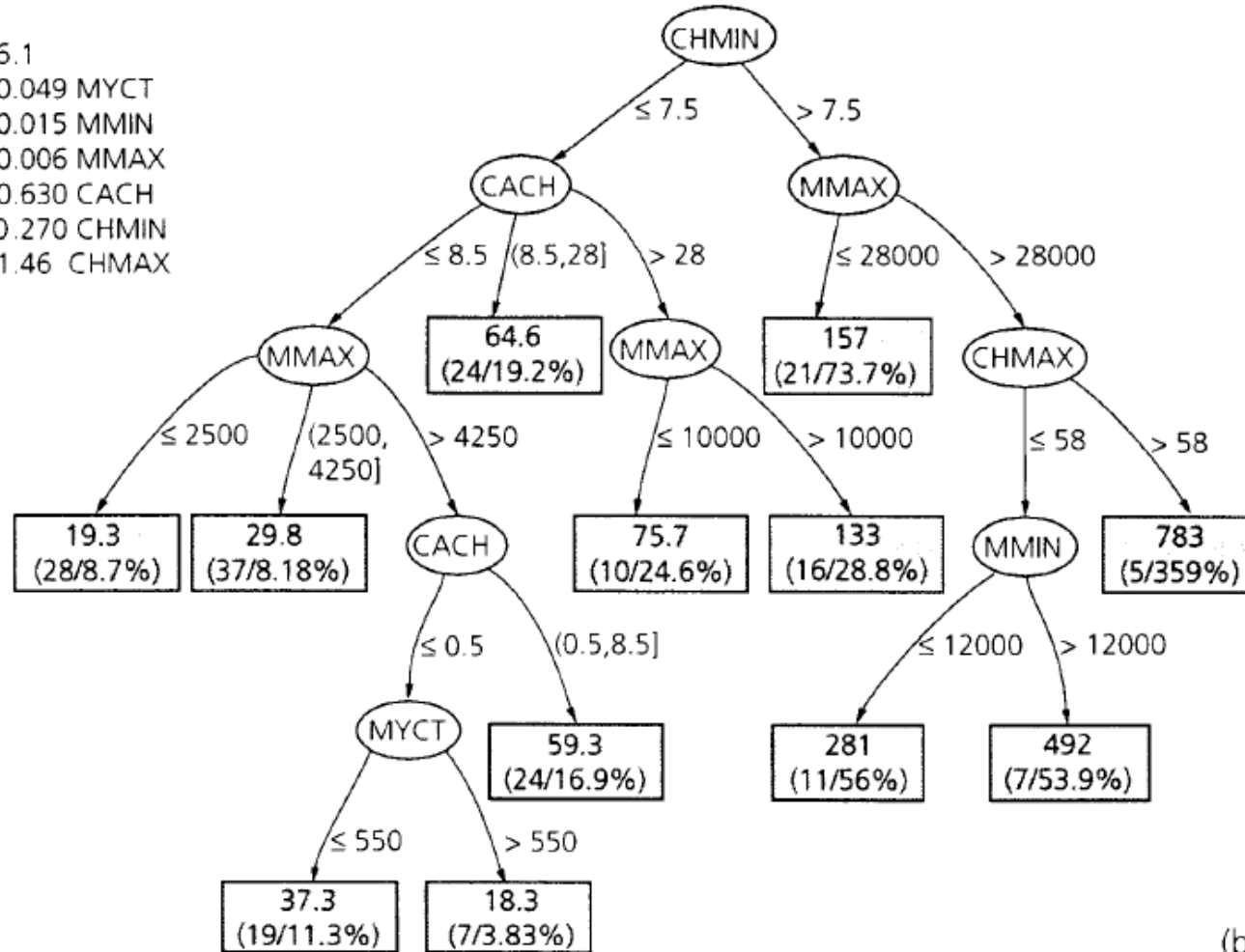
+ 0.006 MMAX

+ 0.630 CACH

- 0.270 CHMIN

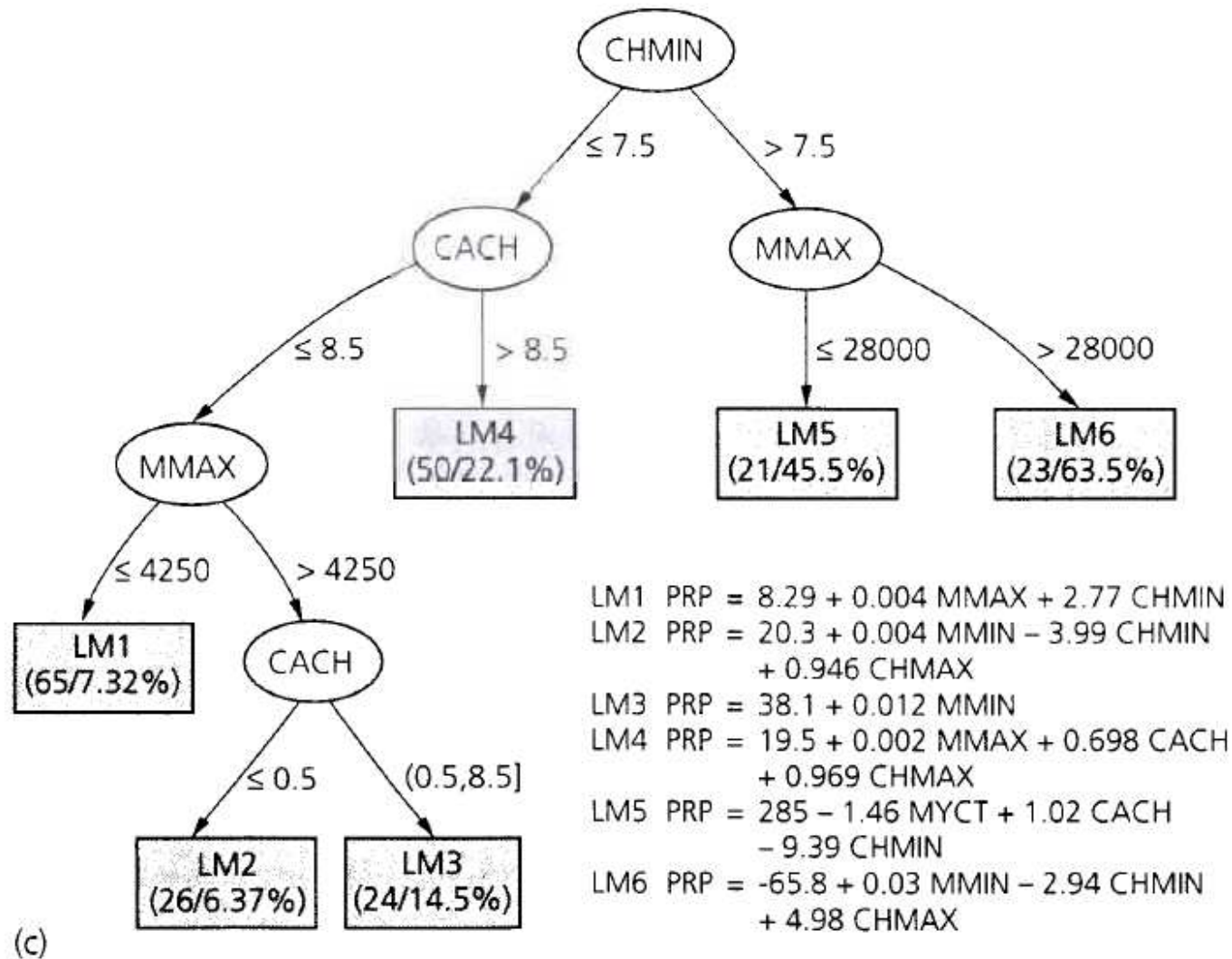
+ 1.46 CHMAX

(a)



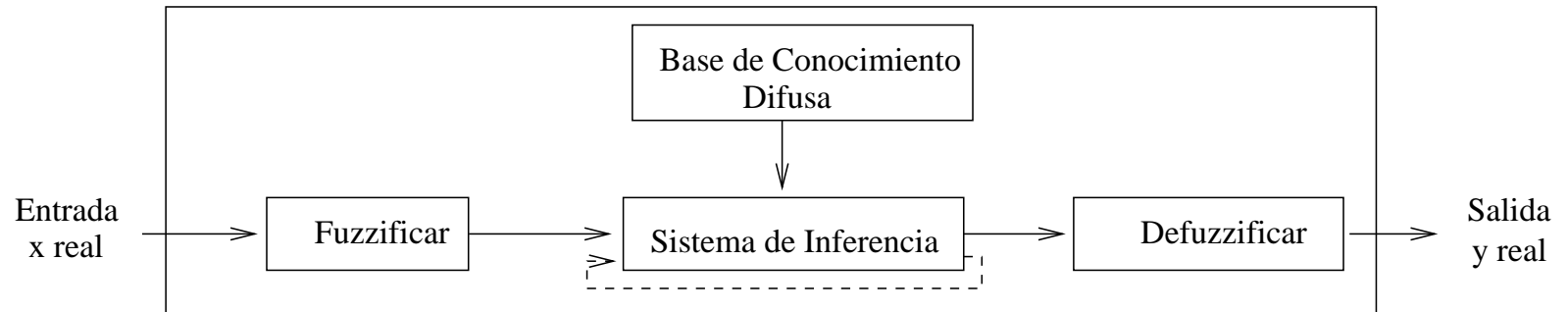
(b)

Árbol de modelos para CPU



Sistemas Basados en Reglas Difusas

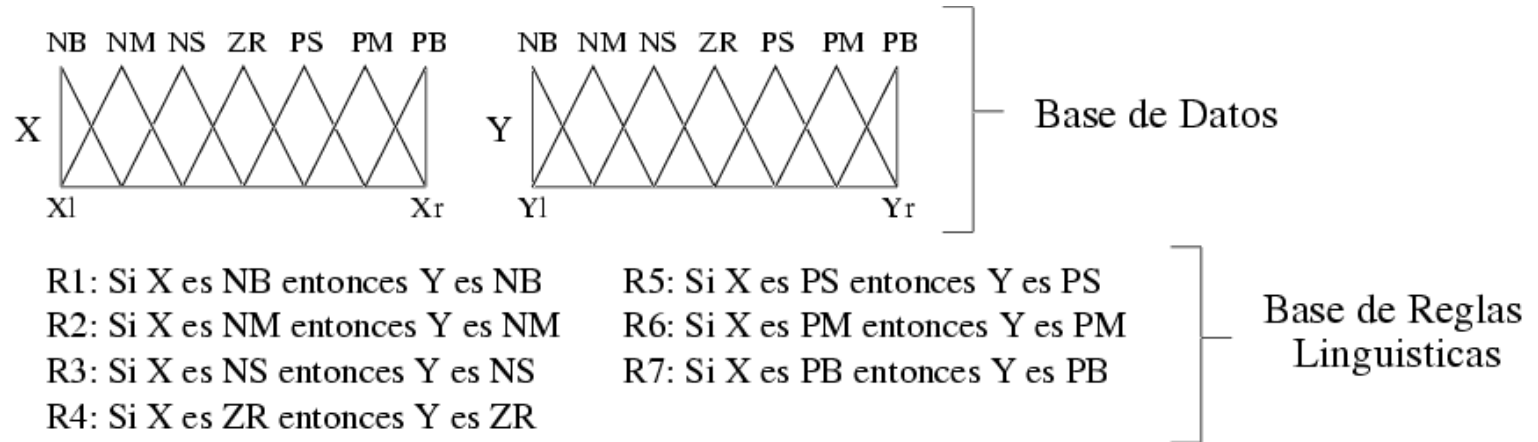
- Los SBRD con defuzzificación son aproximadores universales.



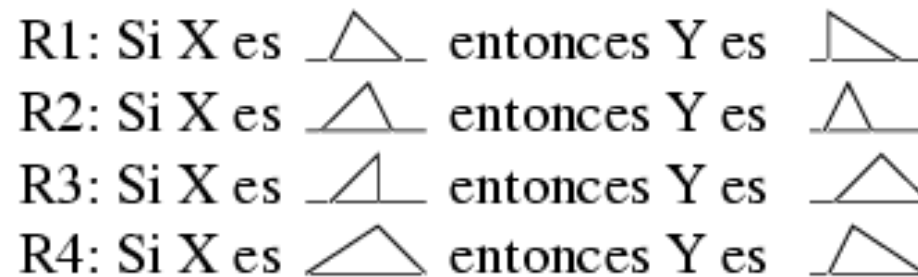
- Hay distintos tipos de modelos de SBRDs (aproximativos, descriptivos), basados en ID3 difuso, ...

SBRDs: descriptivos vs aproximativos

■ Descriptivos



■ Aproximativos



Reglas asociativas

Es la técnica más usada en lo que se conoce como *Market Basket Analysis*

En contraposición con los clasificadores, ahora se trata de predecir el valor de distintos atributos o combinaciones de los mismos.

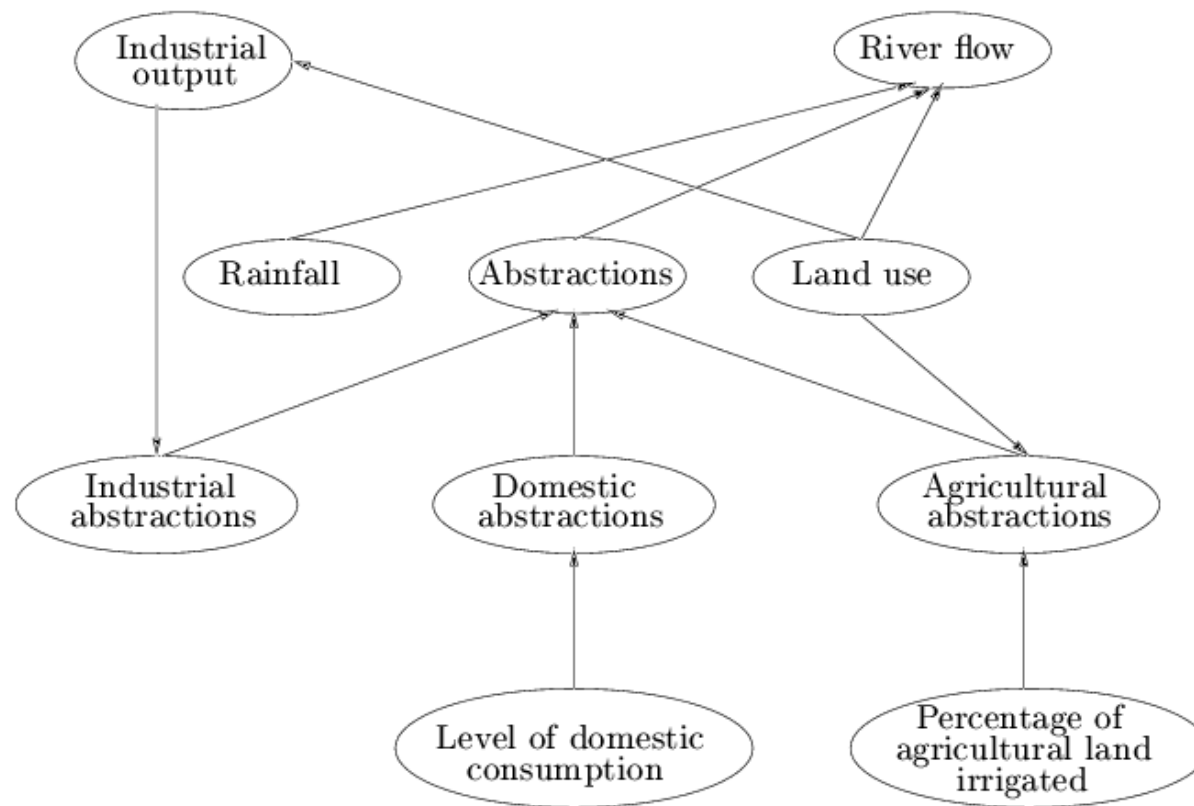
Si pañales Entonces cerveza

Si pizza congelada Entonces coca-cola

Si guerra-de-las-galaxias Entonces expediente-X

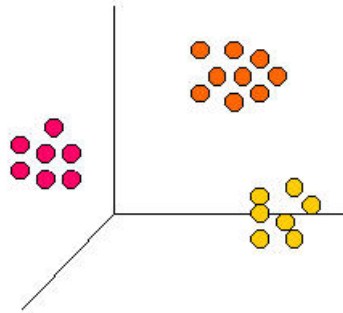
Redes de Creencia

Además de ayudarnos a predecir valores mediante propagación probabilística, nos muestran la estructura interna del problema, así como las relaciones y dependencias entre las variables:



Clustering/Agrupamiento

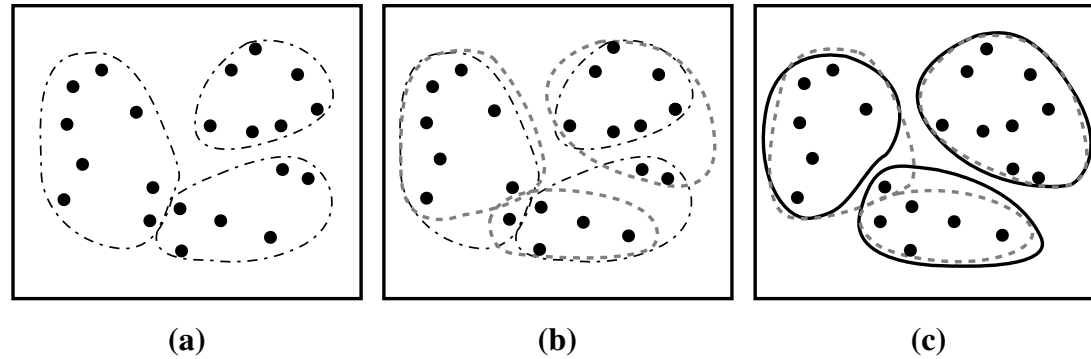
- El **clustering** no es más que una técnica que nos permite agrupar objetos **similares** en grupos (*clusters*)



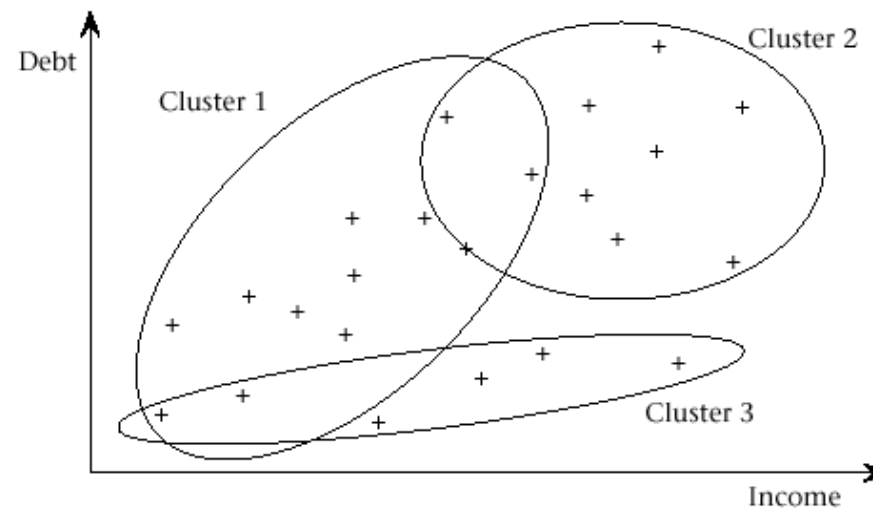
- Los objetos de un mismo grupo son similares entre si y distintos a los de los demás clusters (Clustering duro)
- Clustering difuso: un objeto puede pertenecer a más de un cluster
- Hay también modelos de clustering jerárquicos, en los que los objetos se agrupan de manera arborescente
- El problema puede ser supervisado (se conoce el número de clusters) o no supervisado (no se conoce el número de clusters)

Clustering

■ Clustering duro con k-means

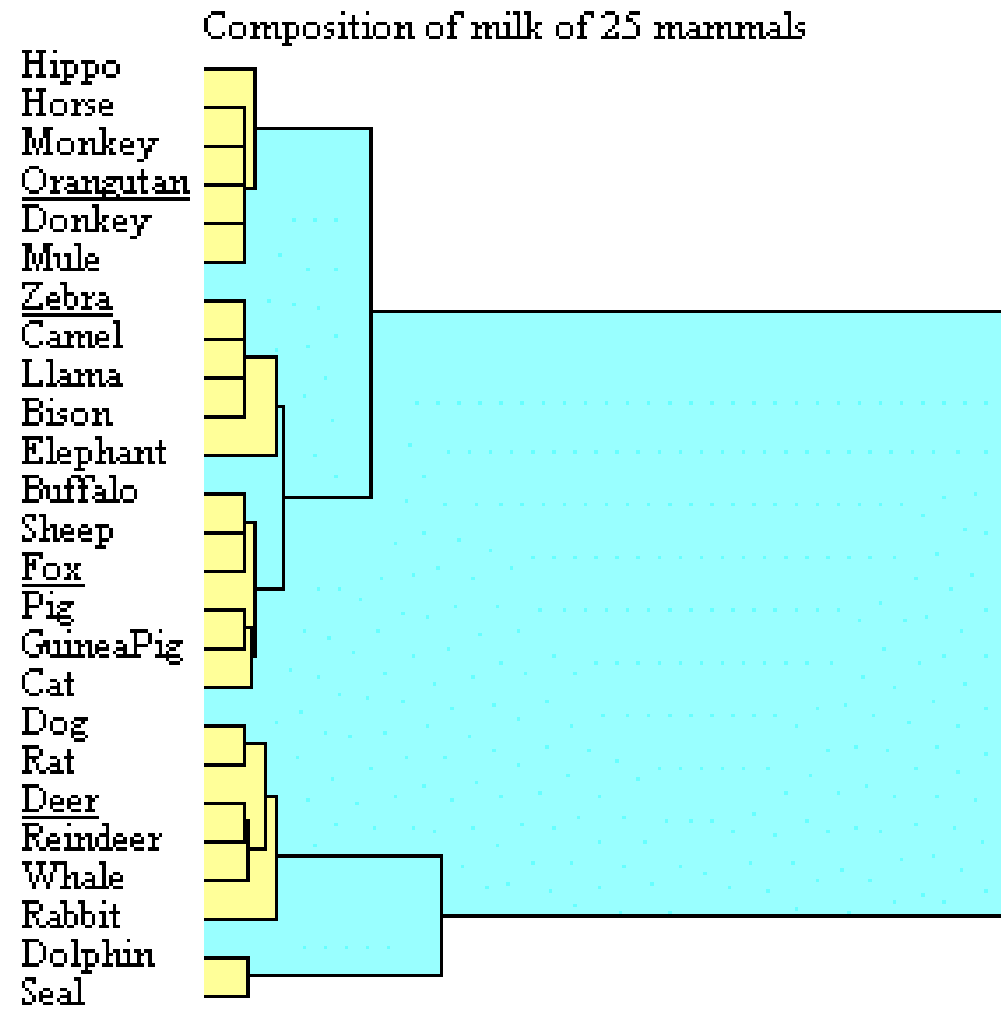


■ Clustering con solapamientos



Clustering jerárquico

Clustering jerárquico



CPU: clustering supervisado

Los valores que aparecen son los de los centroides del k-means

#c	#ej	prob.	MCYT	Memoria RAM		Cache	Canales		Rendimiento
				MMIN	MMAX	CACH	CHMIN	CHMAX	PRP
0	38	0.18	49.7	8150	31287	80	13.3	44.7	332
1	171	0.82	238	1693	7464	12.9	2.7	12.3	47.4
0	34	0.16	46.5	8625	32088	86.2	13.9	47.2	354.1
1	155	0.74	147.7	1889	8350	14.9	3.14	13.9	53.0
2	20	0.1	906	4000	1	0.95	2.05	25.1	

CPU: clustering no supervisado

Los valores que aparecen son medias.

#c	#ej	prob.	MCYT	Memoria RAM		Cache	Canales		Rendimiento
				MMIN	MMAX	CACH	CHMIN	CHMAX	PRP
0	56	0.27	173.3	1342.0	5841.4	6.9	3.1	18.2	35.8
1	5	0.02	104	2000	32000	33.6	7	41.2	189.2
2	34	0.15	90	2046	11254	16.2	1.45	5.9	59.67
3	45	0.21	44.7	5063	18537	50.3	7.6	20.2	146.7
4	11	0.05	27	15179	43478	107	16.63	58.4	610
5	6	0.02	130	2127	15828	104.9	22.97	106.5	188.22
6	52	0.24	506	688.2	3627	1.51	0.98	3.14	24.42