UNIVERSIDAD DE LOS ANDES
MERIDA VENEZUELA

UTPL
UNIVERSIDAD TÉCNICA PARTICULAR DE LOJA

Minería de datos: Tareas o tipos clásicos

Jose Aguilar
CEMISID, Escuela de Sistemas
Facultad de Ingeniería
Universidad de Los Andes
Mérida, Venezuela

Tareas de MD

- Clasificación
- Asociación
- Agrupamiento
- Predicción

Reconocimiento de patrones es un tema tangencial

Cierta cercanía entre predicción y clasificación

Ejemplo inicial de MD

predecir si se puede jugar o no dadas las características del clima.

Pronostico	Temperatura	Humedad	Viento	Jugar
soleado	caliente	alta	falso	no
soleado	caliente	alta	verdadero	no
nublado	caliente	alta	falso	si
lluvioso	templado	alta	falso	si
lluvioso	fresco	normal	falso	si
lluvioso	fresco	normal	falso	si
nublado	fresco	normal	verdadero	si
soleado	templado	alta	falso	no
soleado	fresco	normal	falso	si
lluvioso	templado	normal	falso	si
soleado	templado	normal	verdadero	si
nublado	templado	alta	verdadero	si
nublado	caliente	normal	falso	si
lluvioso	templado	alta	verdadero	no

Tareas de MD

- Creando simples reglas decisivas se puede generar un modelo que permitan realizar dicha predicción. Por ejemplo:

Si Pronostico = soleado y humedad = alta, entonces jugar= no

- Generando más reglas para cubrir todos los casos, va emergiendo el conocimiento (en este caso, un **modelo basado en reglas**), que permite la predicción de si jugar o no.

Clasificación

Clasificación

Email: Spam / No es Spam?
Transacciones en línea: Fraudulento (Si / No)?
Tumor: Maligno / Benigno ?

$y \in \{0, 1\}$

0: “Clase negativa” (tumor benigno)
1: “Clase positiva” (tumor malignano)

Clasificación

Obtener una función o modelo que determine la clase de un objeto basado en las características de sus atributos.

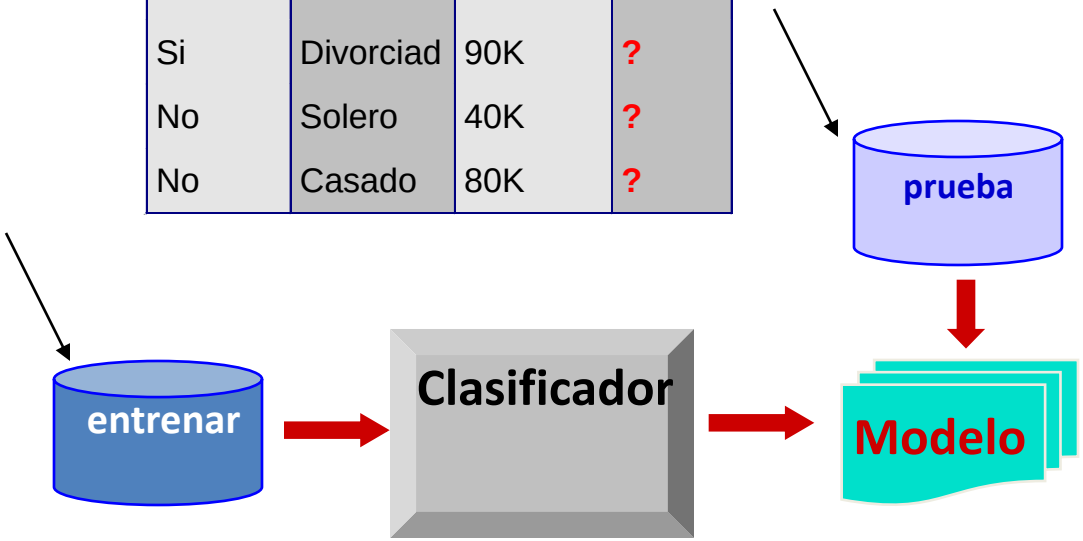
- Para generar dicho modelo o función, es necesario definir un conjunto de datos de entrenamiento, compuesto por objetos que ya tienen su clase asignada, también denominados ejemplos etiquetados.
- El modelo o función es creado analizando las relaciones entre los atributos de los objetos en el conjunto de entrenamiento y las clases.
- Mientras mas variedad de escenarios se presenten en el conjunto de entrenamiento, mas se enriquece el modelo de clasificación (mejores resultados en la clasificación de nuevas entradas no etiquetadas).

Clasificación

	categoría	categoría	continuo	clase
ID	Reemb	Edo Civil	pago Impuest	Engaña
1	Si	Soltero	125K	No
2	No	Casado	100K	No
3	No	Soltero	70K	No
4	Si	Casado	120K	No
5	No	Divorc.	95K	Si
6	No	Casado	60K	No
7	Si	Divorciad	220K	No
8	No	Soltero	85K	Si
9	No	Casado	75K	No
10	No	Soltero	90K	Si

nominales numéricos

Reemb	Edo. Civil	pago Impuest	Engaña
No	Soltero	75K	?
Si	Casado	50K	?
No	Casado	150K	?
Si	Divorciad	90K	?
No	Solero	40K	?
No	Casado	80K	?



Clasificación

Detección de Fraude

Objetivo: Predecir casos fraudulentos en las transacciones con tarjetas de crédito.

Enfoque:

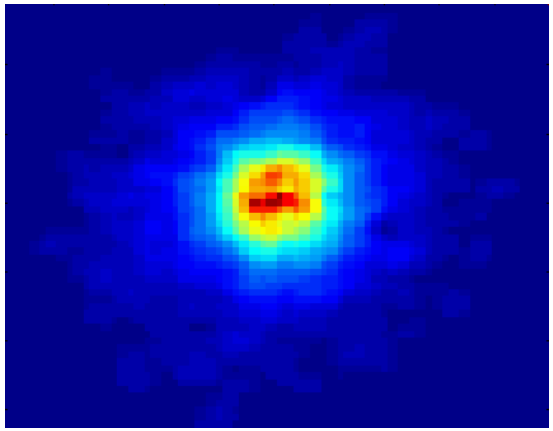
- Utilizar las transacciones con tarjetas de crédito y la información sobre la cuenta del titular como atributos.
  - ¿Cuándo compra un cliente?, ¿qué compra?, ¿con qué frecuencia paga a tiempo?, etc.
- Etiquetar transacciones pasadas: fraudulentas o correctas. Esto forma el atributo de clase.
- Aprender un modelo para las clases de transacciones.
- Utilice este modelo para detectar futuros fraudes mediante la observación de las transacciones de las tarjetas de crédito en una cuenta.

Clasificación

Clasificar cada imagen como una estrella (y su estado de formación) o una imagen no estelar (galaxia) (no-estelar)

http://aps.umn.edu

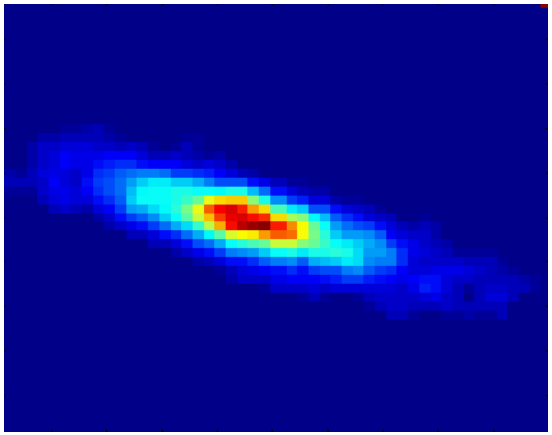
Temprano



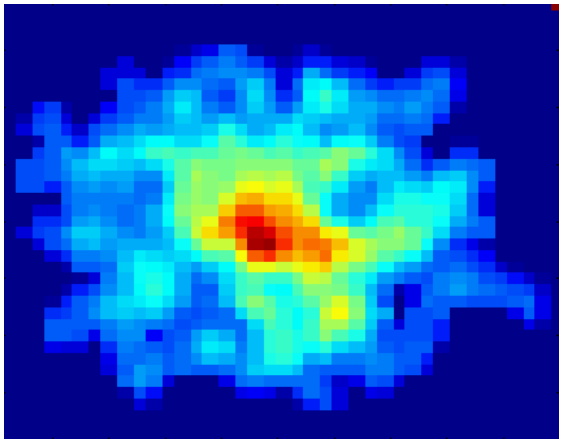
Clases:
- **Estado de Formación**

Atributos:
- **Caract. Imagen,**
- **Caract. Ondas, luz, etc.**

Intermedio

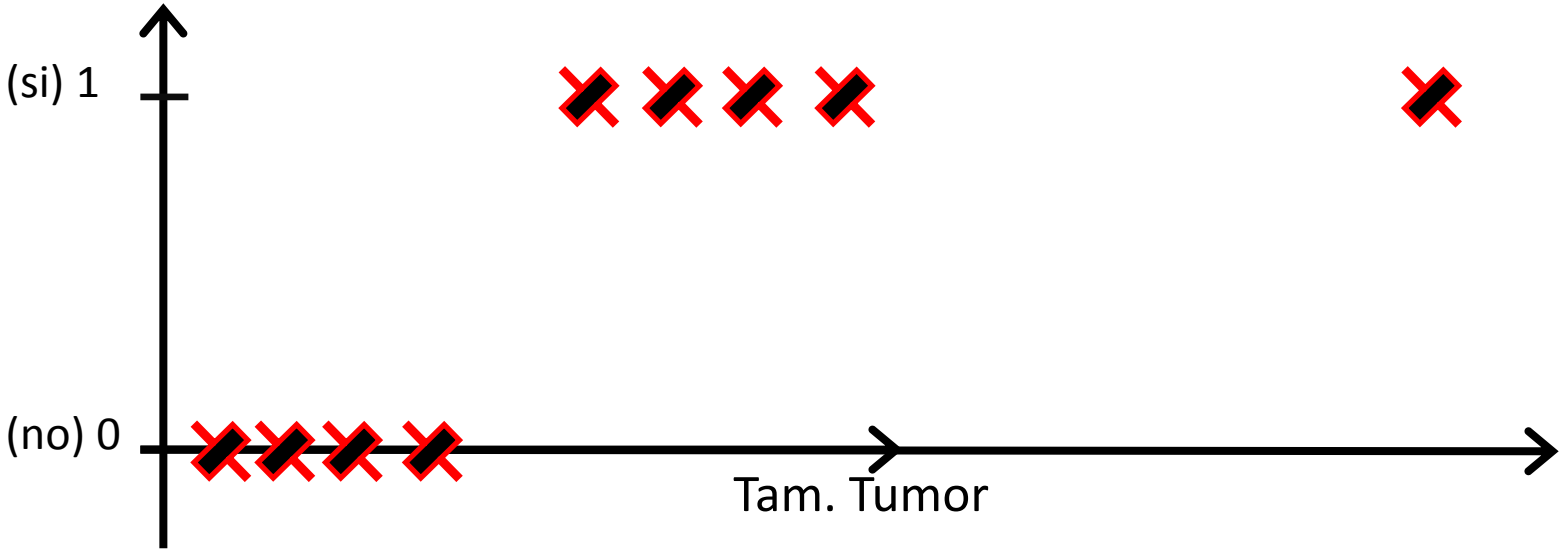


Tarde



Tam. datos:
- 72 millones estrellas, 20 millones galaxias
- BD Imagen: 150 GB

Clasificación



Umbral clasificador $h_\theta(x)$ = 0.:

Si $h_\theta(x) \geq 0.5$, predice “y = 1”

Si $h_\theta(x) < 0.5$, predice “y = 0”

Algoritmos de Clasificación

- **Basados en análisis estadísticos (Clasificación Bayesiana):** Gaussian Naive Bayes, Bernoulli Naive Bayes, La regresión y sus variantes: regresión lineal, regresión logística, isotonic regresión, entre otros,Procesos gaussanianos, Redes bayesianas

- **Basados en Arboles de decisión:** J48, CART, C4.5, ADtree,randomTree, REPTree,

- **Basados en reglas:** ZeroR, M5Rule, ConjunctiveRule, PART

- **Basados en distancia**

- **Basados en redes neuronales:** perceptron simple y el perceptron multicapa

- **Híbridos**

Clasificación Bayesiana

- Dado un conjunto de datos $X = \{x_1, x_2, \dots, x_n\}$,

- El teorema de Bayes es utilizado para estimar la probabilidad de que una hipótesis sea cierta dada una instancia.

Entendiéndose como hipótesis la pertenencia a una clase.

- Dicho teorema viene dado por $P(h_1|x_i)$ probabilidad de que la hipótesis h_1 sea verdadera dada el ejemplo x_i

Redes Bayesianas

Las redes bayesianas son grafos dirigidos acíclico cuyos nodos representan variables aleatorias en el sentido de Bayes

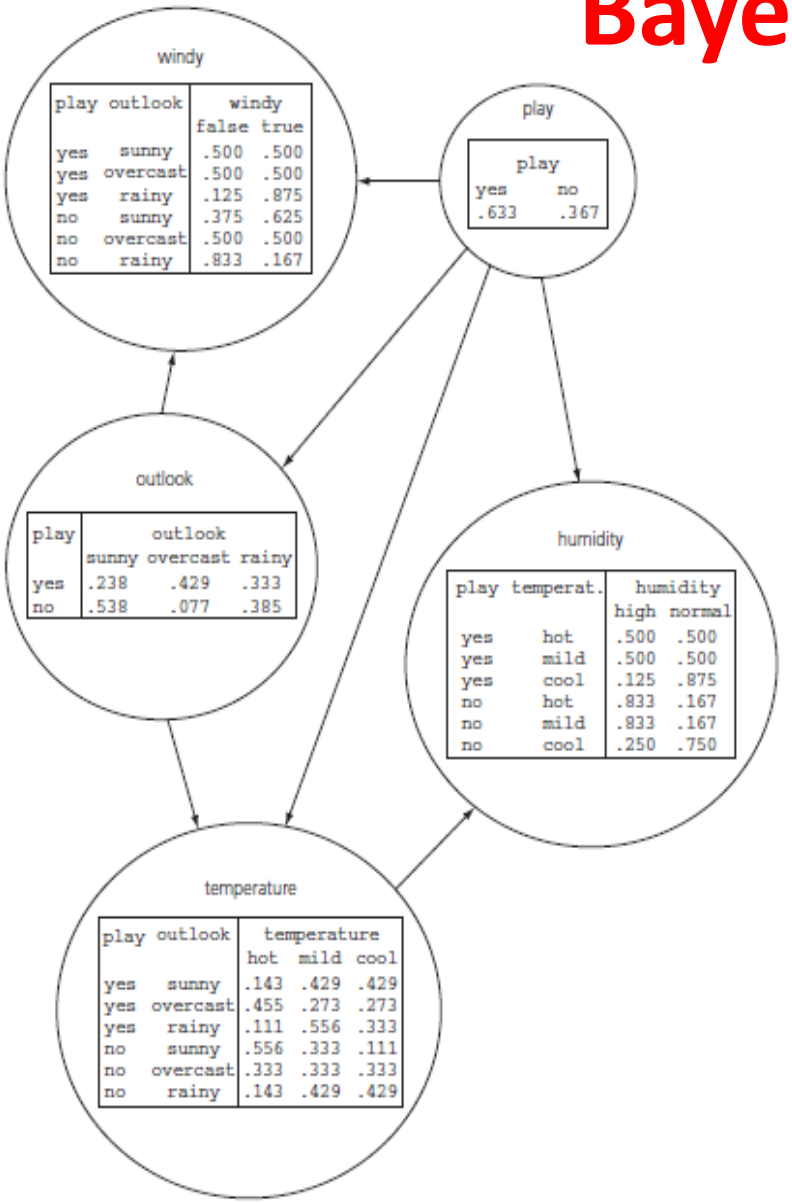
En el teorema de Bayes se expresa la probabilidad condicional de un evento aleatorio A dado B en términos de la distribución de probabilidad condicional del evento B dado A y la distribución de probabilidad marginal de sólo A. Pueden ser cantidades observables, variables latentes, parámetros desconocidos o hipótesis.

$$P(A_i|B) = \frac{P(B|A_i)P(A_i)}{P(B)}$$

Redes Bayesianas

- Las aristas representan dependencias condicionales
- Los nodos que no se encuentran conectados representan variables las cuales son condicionalmente independientes de las otras.
- Cada nodo tiene asociado una función de probabilidad que toma como entrada un conjunto particular de valores de las variables padres del nodo y devuelve la probabilidad de la variable representada por el nodo.

Haciendo predicciones con Redes Bayesianas



probabilidad incondicional

⬇

Probabilidad Conjunta

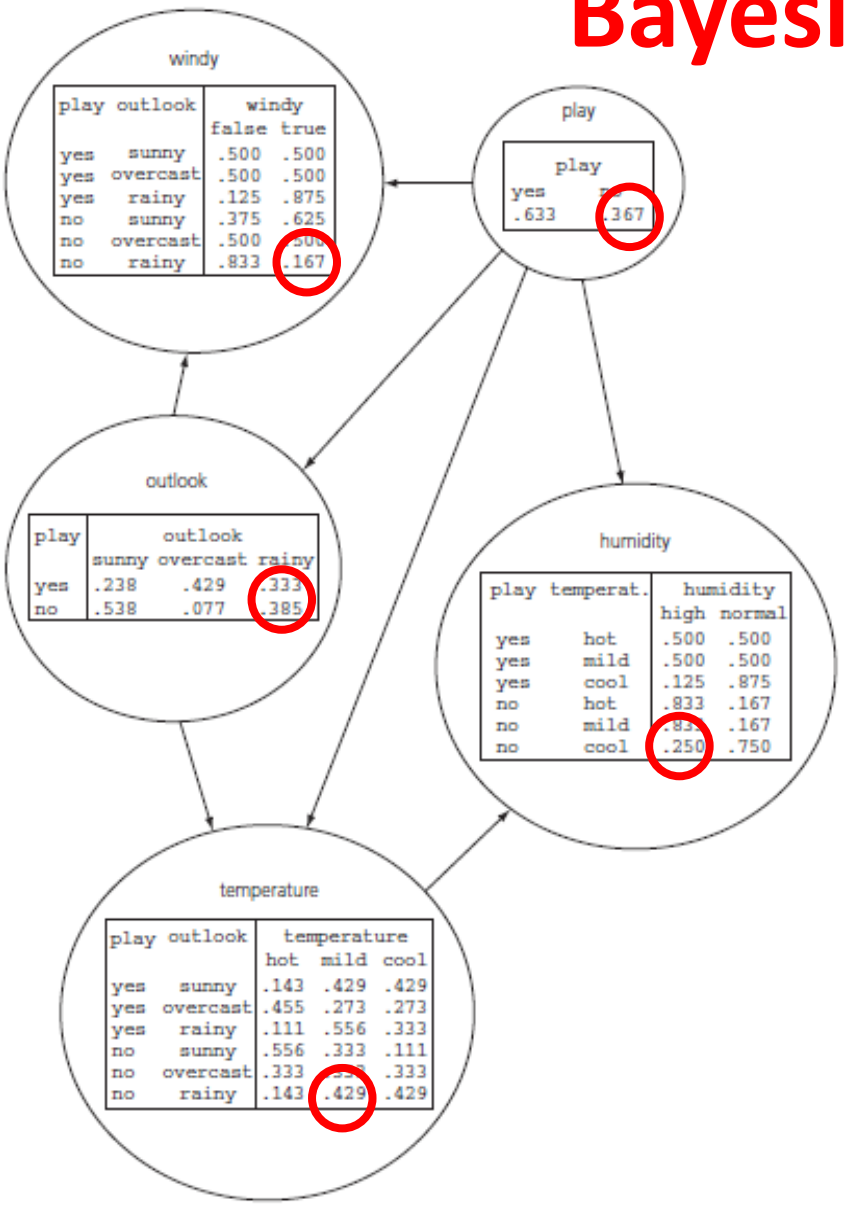
⬇

Distribución de Probabilidades

Por ejemplo, considerar la posibilidad de una instancia con valores

panorama = lluvias, temperatura = frío, humedad = alto, y con viento = existe

Haciendo predicciones con Redes Bayesianas

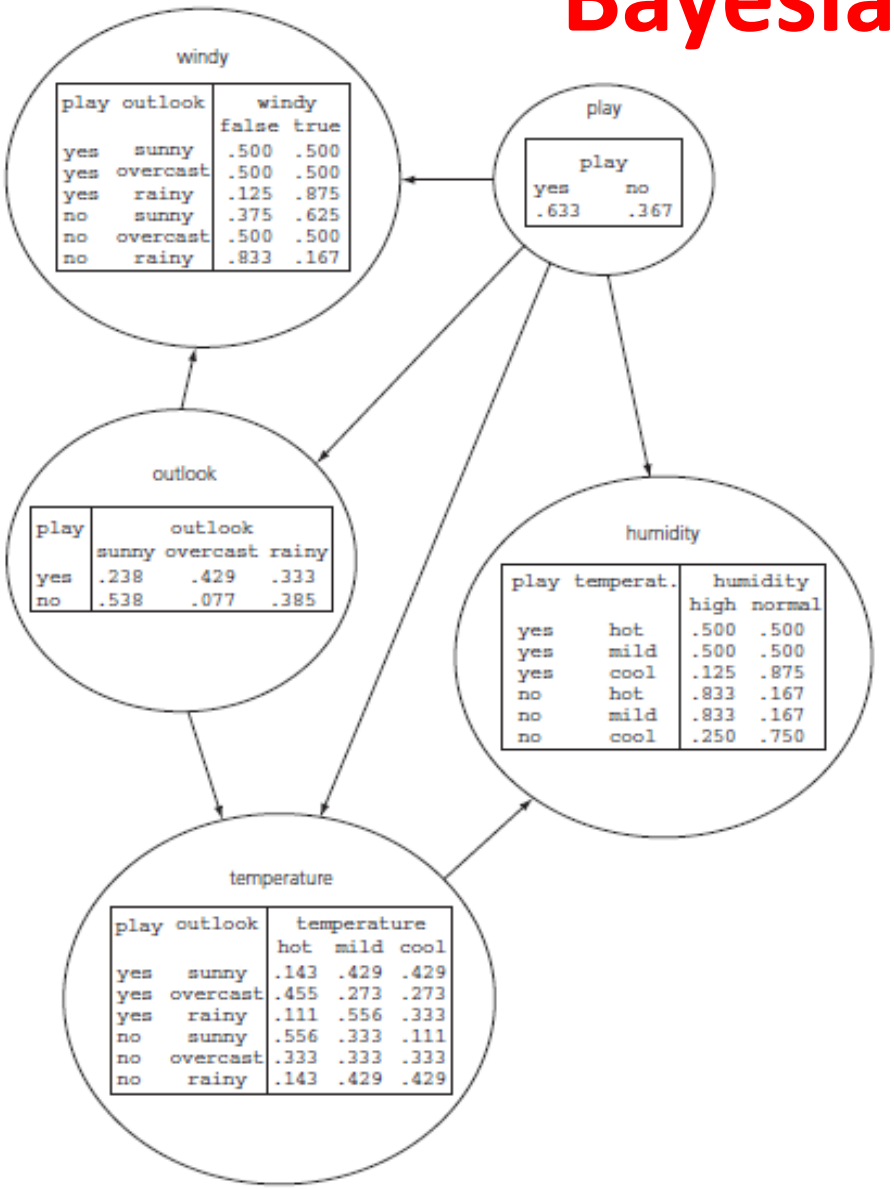


Para **panorama = lluvias, temperatura = frío, humedad = alto, y con viento = existe**

Calcular **la probabilidad para jugar = no**, en la red da probabilidad:

- 0.367 desde el nodo Play,
- 0.385 desde outlook,
- 0.429 desde temperature,
- 0.250 de humidity, y
- 0.167 de windy

Haciendo predicciones con Redes Bayesianas



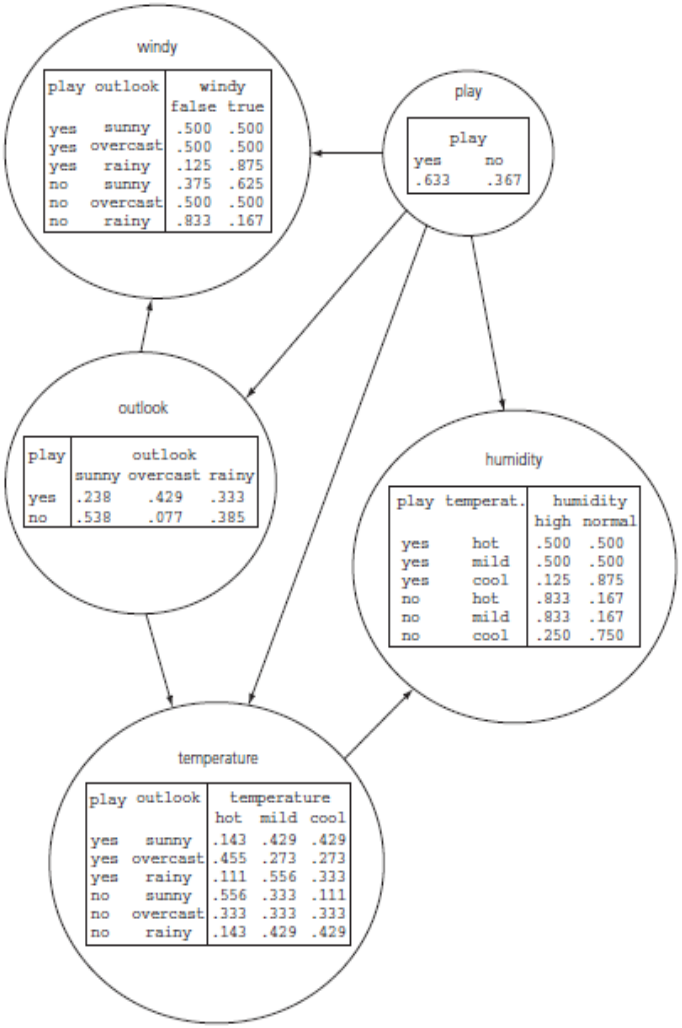
El producto es 0,0025.

El mismo cálculo para el **juego = si** es 0.0077.

Sin embargo, estos no son la respuesta final:

las probabilidades finales deben sumar 1

Haciendo predicciones con Redes Bayesianas



En realidad, **son las probabilidades conjuntas**

Pr [jugar = no, E] y Pr [jugar = si, E]

donde E representada los valores de los atributos de la instancia (evidencias) que se quiere evaluar.

Para obtener las probabilidades condicionales

Pr [jugar = no|E] y Pr [jugar = si|E],

normalizar las probabilidades conjuntas dividiéndolas por su suma.

Esto da que las probabilidades para jugar = no es 0,245 y para jugar = si es 0.755

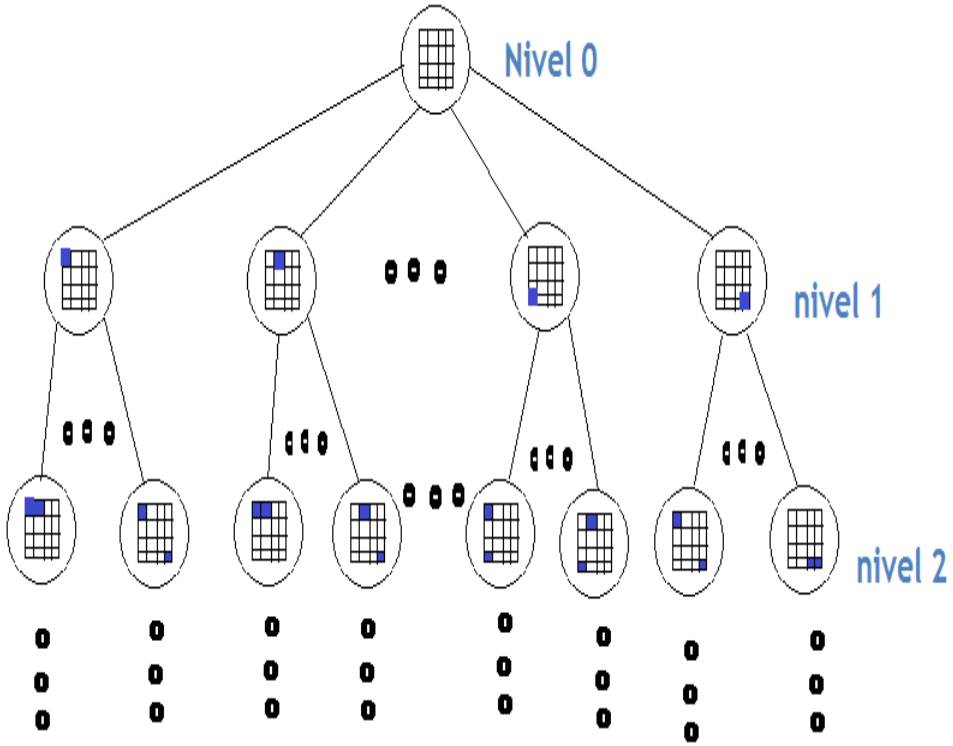
Aprendizaje de Redes Bayesianas

El aprendizaje, en general, de redes bayesianas consiste en inducir un modelo, estructura y parámetros asociados, a partir de datos.

Este puede dividirse naturalmente en dos partes:

- **Aprendizaje estructural**. Obtener la estructura o topología de la red.
- **Aprendizaje paramétrico**. Dada la estructura, obtener las probabilidades asociadas.

Manejo de Incertidumbre



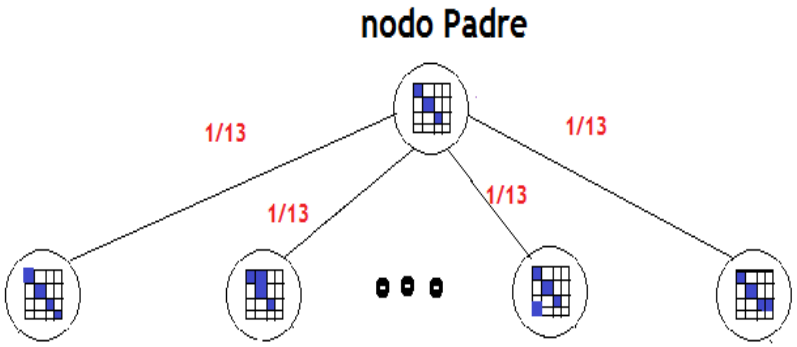
Red bayesiana para el manejo de incertidumbre

- $MUE = \max(U(a_i) * \sum_{i=0}^{n} P_i(^{a_i}/_{P_i}))$

Caso juego: Según la función MUE la mejor acción será aquella en la cual la razón dada entre la utilidad y la probabilidad de que el oponente obtenga una mala jugada sea máxima.

Modelo Matemático de Aprendizaje

Se tiene el siguiente Árbol con 13 nodos



Red bayesiana en su estado de máxima confusión

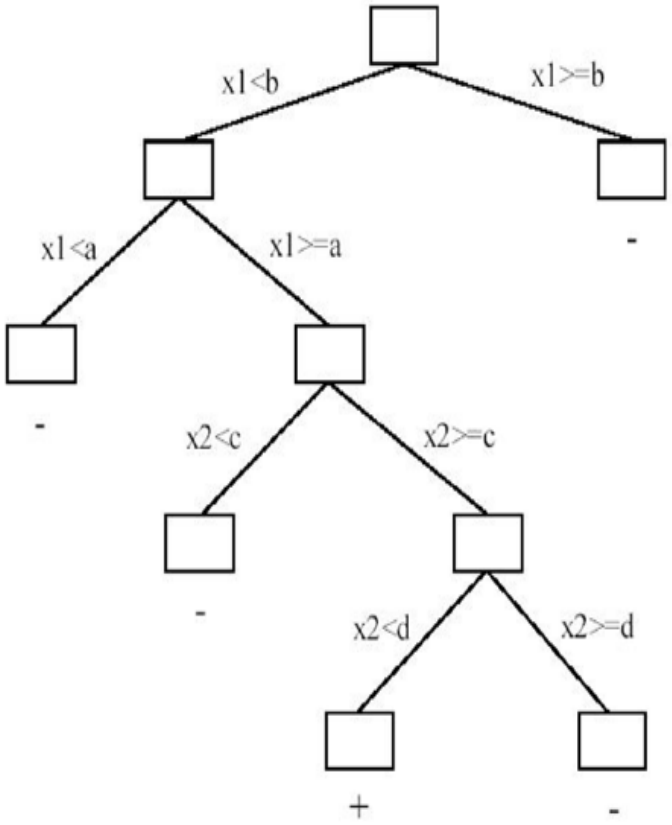
Según acción del adversario sea buena o no, la rama debe ser premiada (o penalizada) y las del resto de hermanos inversamente modificadas (aprendizaje reforzado)

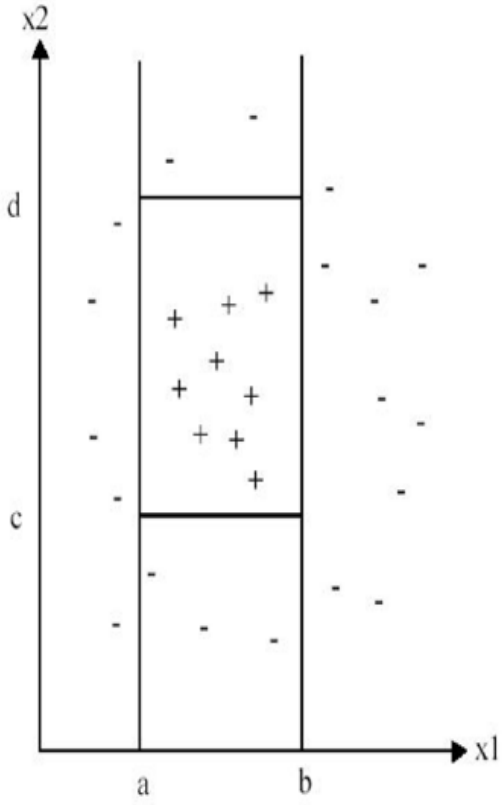
Para actualizar las ramas se pueden usar los siguientes valores:

- $\mathbf{p_{obj}}$= 6/10 se suma (resta) a la rama evaluada para premiar (castigar)
- $\mathbf{p_{resto}}$=2/10 se resta (suma) al resto de ramas para penalizar (premiar)

Basados en Arboles de decisión

Este algoritmo se basa en la construcción de un árbol, en el cual los nodos son los atributos y las hojas la clases.





se utiliza la **entropía** como métrica para tomar la decisión de cómo particionar el árbol, ya que indica la medida de información que proporciona cada atributo.

indica los atributos que aportan más información, por lo tanto son los nodos cercanos a la raíz

Arboles de decisión

Los árboles de decisión son unos de los algoritmos clasificadores más conocidos y usados en las tareas de Data Mining, ya que son una forma de representación sencilla para clasificar instancias.

Árboles de decisión son particiones de un conjunto de datos

Objetivo: Segmentar la población para encontrar grupos homogéneos según una cierta variable.

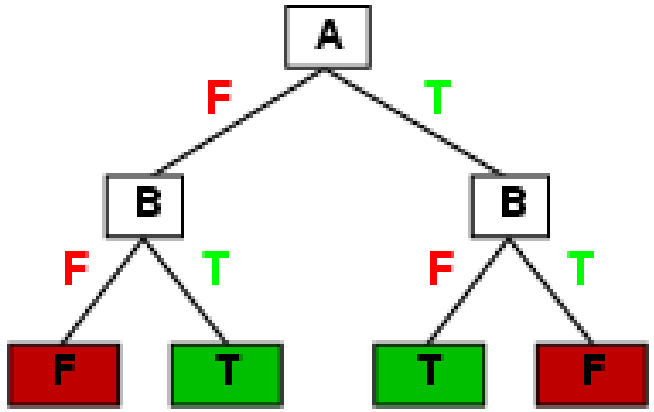
Árbol de Decisión

- Parte de un conjunto de entrenamiento

- Un árbol de decisión es consistente para cualquier conjunto de entrenamiento, cuando hay un camino a una hoja para cada uno de los miembros del conjunto

- Está basado en la idea de tablas de la verdad:

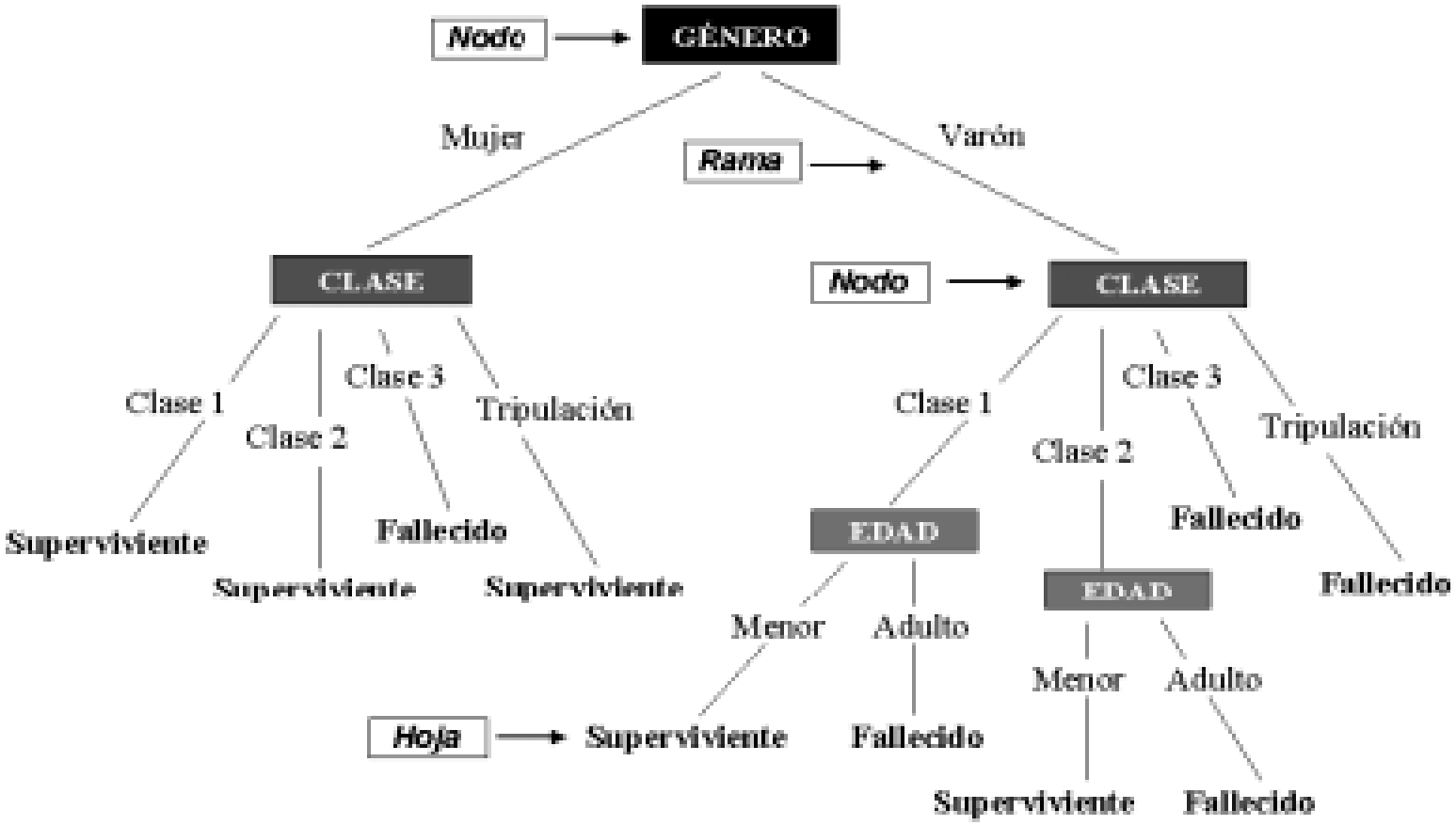
A	B	A xor B
F	F	F
F	T	T
T	F	T
T	T	F



Es una estrategia de aprendizaje inductivo

ÁRBOLES DE DECISION

Suelen ser empleados en tareas de clasificación, y también, aunque menos, en tareas de predicción



Ej. Acontecimientos relativos al hundimiento del Titanic

Construcción del Árbol de Decisión

- Idea: escoger atributo "más significativo" como raíz del (sub)-árbol

¿Cómo?

- Si hay + y - ejemplos escoger atributo que mejor los divida (mayor discriminante)
- Si hay particiones con + y -, buscar un 2do atributo para seguir partiendo

Macroalgoritmo AD(ejemplos, atributos)

```
Si ejemplos no vacios entonces
    Si ejemplos clasificados entonces
        regresar (clasificación)
    de lo contrario
        mejor: escoger_atributo(atributos, ejemplos)
        arbol: un nuevo árbol de decisión con mejor como raíz
        por cada valor Vi de mejor
            Subejemplos:ejemplos con mejor=Vi
            Subarbol: AD(Subejemplos, atributos)
            Arbol: actualizar(nueva rama con etiqueta Vi y Subarbol)
Regresa(arbol)
```

Árbol de Decisión

Toma como entrada una situación y da como salida una decisión (por ejemplo: si/no)

- Ejemplo: **decidir si esperar o no por una mesa en un restaurant basado en los siguientes criterios**

1. ¿Hay otro restaurant cerca? (Alternativa)
2. ¿Hay un bar confortable para esperar? (Bar)
3. ¿Hoy es Viernes o Sábado? (Día)
4. ¿Hay hambre? (EdoM)
5. Numero de personas en el restaurant (Patrón: Vacio, Algo, lleno)
6. ¿Precio? ($, $$, $$$)
7. ¿Esta lloviendo? (Edo.D)
8. ¿Se tiene una reservación?
9. Tipo de restaurant (Francés, Italiano, Japonés, Hamburguesa)
10. Tiempo de espera estimado (0-10min, 10-30min, 30-60, >60min)

Ejemplos para construir un AD: Tabla de decisiones

Ejemplos

Criterios

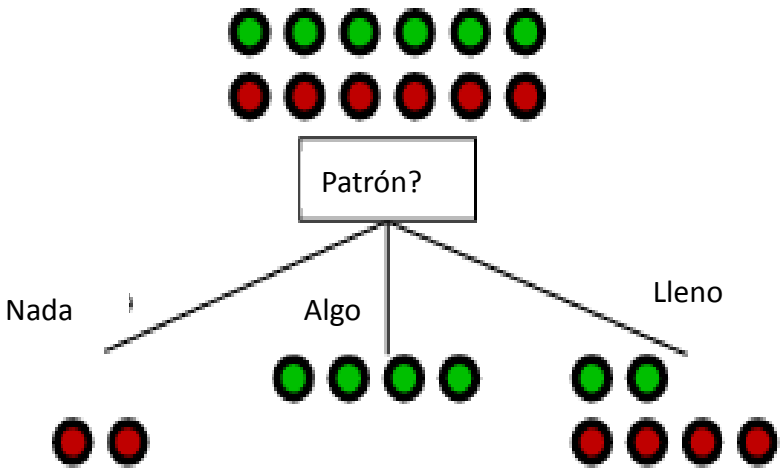
Qué aprendo?

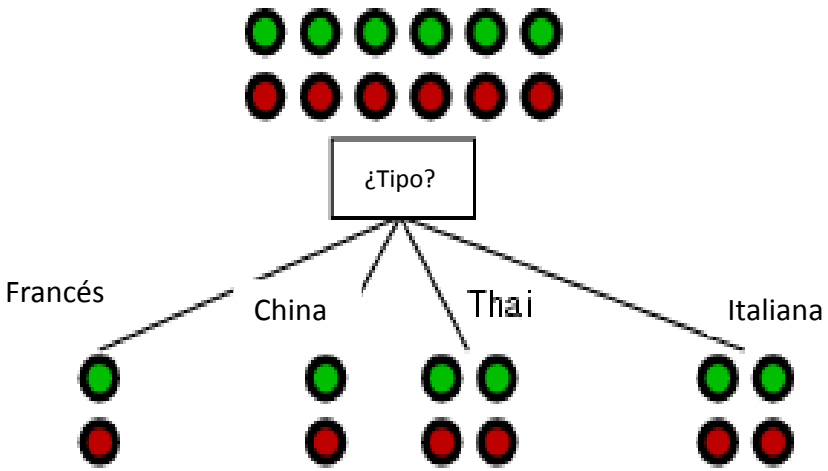
Ej	Alt	Bar	Dia	EdM	Patr	Prec	EdD	Tipo	RES	T --->	Espera
X1	S	N	N	S	Alg	$$$	N	Franc	S	0-10	S
X2	S	N	N	S	llen	$	N	Jap	S	10-15	N
X3	N	S	N	N	Alg	$	N	Hamb	N	0	S
...											
X12	S	S	S	S	llen	$	N	Hamb	N	10	S

Escoger un atributo

aprender reglas (clases)

¿***Patrón*** es una mejor escogencia que ***Tipo***?





Basado en conceptos vinculados a *contenidos de información,* p.ej.:

$$Info(p,n) = -p\log_2(p) - n\log_2(n)$$

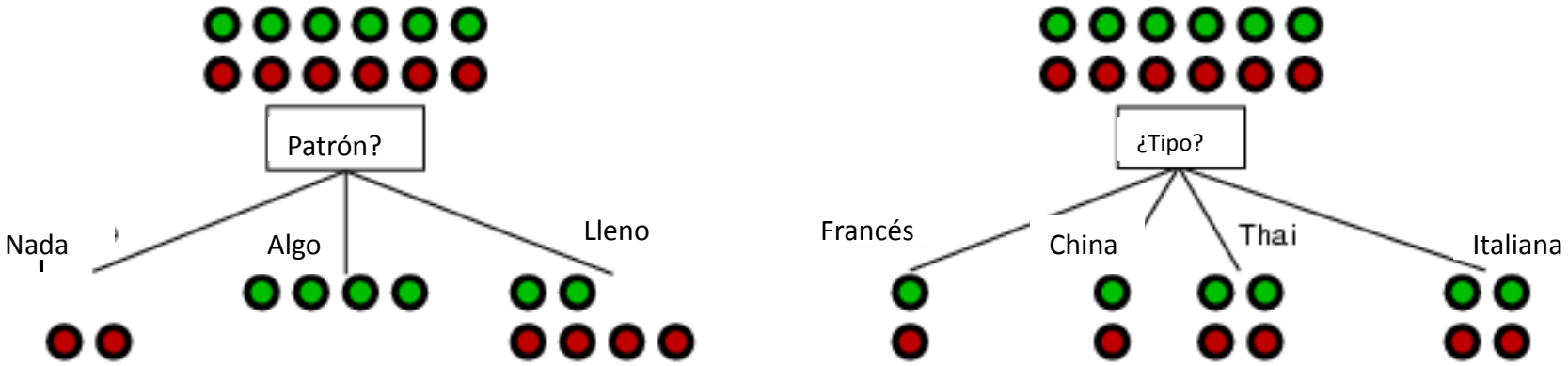
Es una medida de la entropía (grado de desorden) de los ejemplos

n: numero de ejemplos - p: numero de ejemplos +

Escoger un atributo

aprender reglas (clases)

¿***Patrón*** es una mejor escogencia que ***Tipo***?



Donde:

I es entropía de los ejemplos:

y

$$IG(A) = I(\frac{p}{p+n}, \frac{n}{p+n}) - resto(A)$$

V: posibles valores de A

p_i y n_i? ver siguiente lamina

$$I(\frac{p}{p+n}, \frac{n}{p+n}) = -\frac{p}{p+n}\log_2\frac{p}{p+n} - \frac{n}{p+n}\log_2\frac{n}{p+n}$$

$$\text{resto(A)} = \sum_{i=1}^{v} \left|\frac{p_i - n_i}{p+n}\right| I(\frac{p_i}{p_i+n_i}, \frac{n_i}{p_i+n_i})$$

Escoger un atributo

aprender reglas (clases)

¿Quién es p_i? p_i puede ser $p_i = \frac{\left|E_i^+\right|}{\left|E_i^+\right| + \left|E_i^-\right|}$

Donde E_i^+ es el porcentaje de ejemplos clasificados como + por el valor v_i del atributo A

Una Formula general para escoger a los atributos:

Como hay que elegir el atributo con mayor información (menor entropía), otra posibilidad es calcular una **función de merito (FM)**

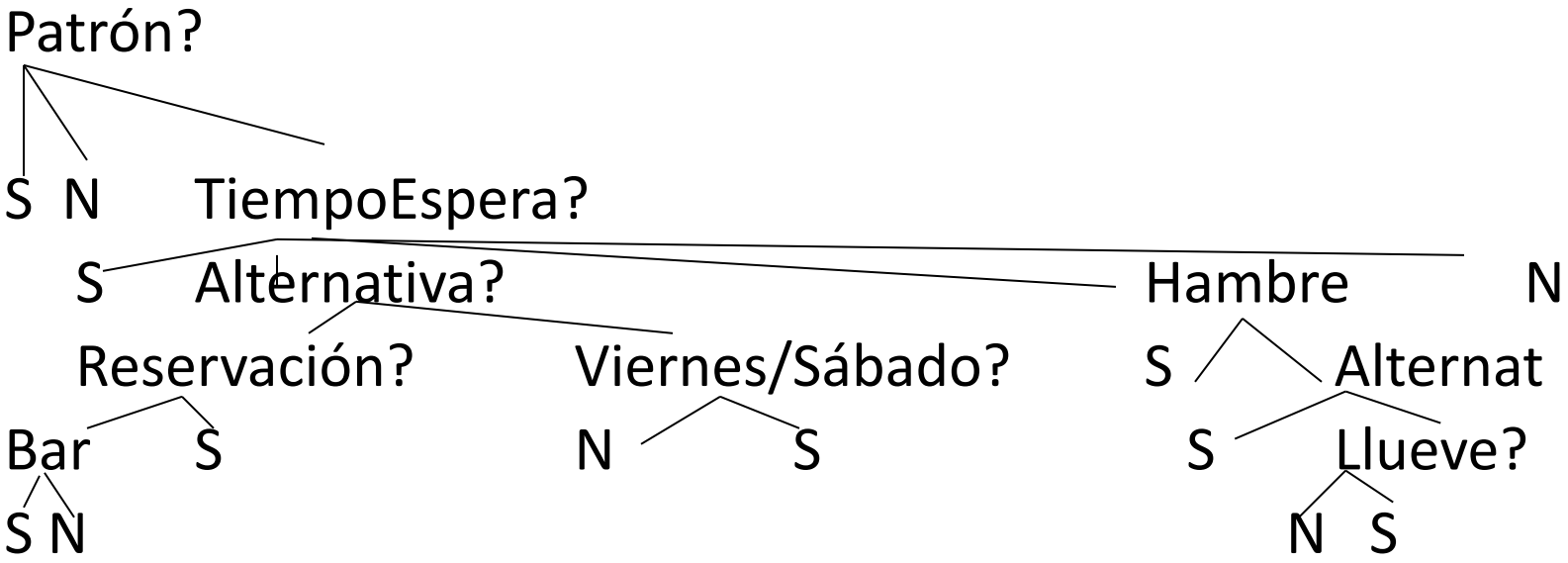
$$\mathrm{FM(A)} = \sum_{i=1}^{v} \mathrm{r_i \inf o(p_i, n_i)}$$

$p_i =$ % ejemplos clasificados como + en la rama i

$$r_i = \left| \frac{p_i - n_i}{p + n} \right|$$

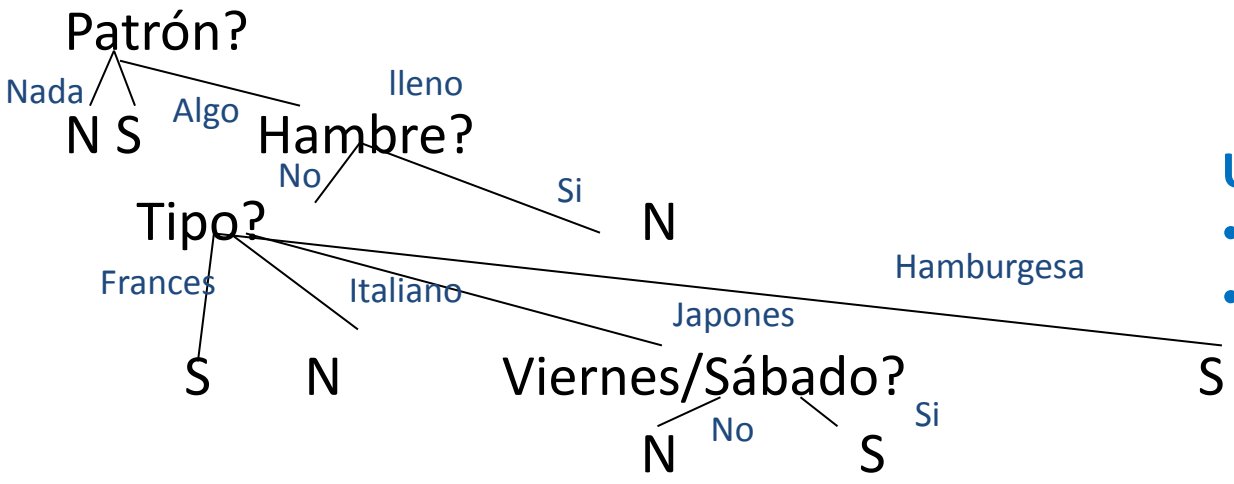
Árbol de Decisión

Para nuestro ejemplo inicial:



Arbol de Decisión y Lógica de Predicado

∀r espera(r) => Patrón(r, algo) O (Patrón(r, full) Y NoHambre(r) Y tipo(r, francés)) O (Patrón(r, full) Y NoHambre(r) Y tipo(r, hamburguesa)) O (Patrón(r, full) Y NoHambre(r) Y tipo(r, Japones) Y viernes/Sabado(r))



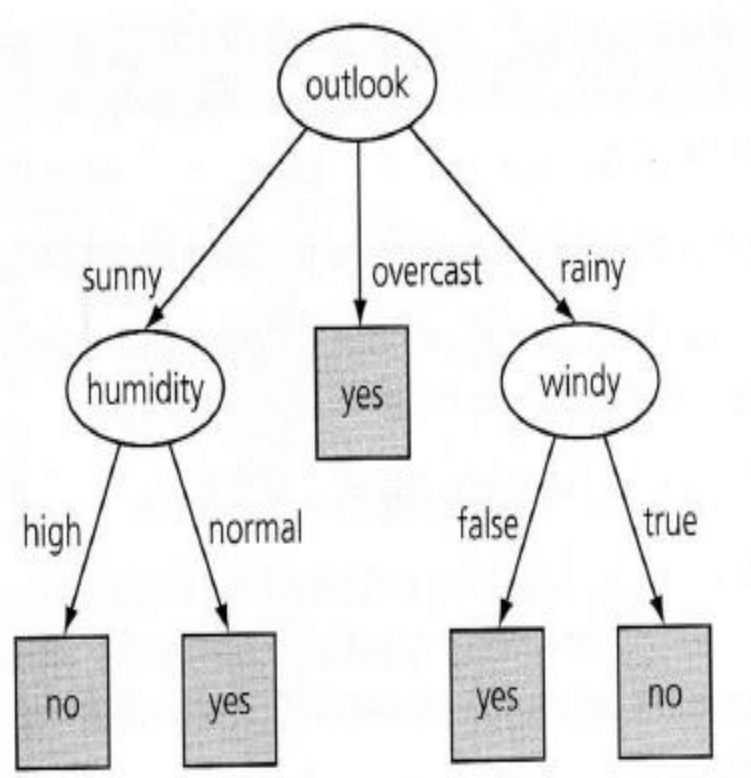
Uso de operadores:

- **Para unir ramas O**
- **Para seguir una rama Y**

Construcción de árboles de decisión

Se completa el árbol completando cada rama hasta cumplir un ciertos compromisos:

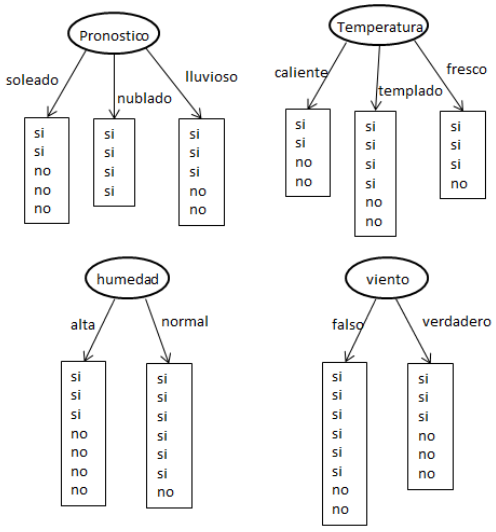
- **Número mínimo de hojas.**
- **Cobertura**: Mínimo número (o porcentaje) de casos posibles cubiertos correctamente de la BD.
- **Precisión:** Error de clasificación menor de un umbral puesto. Por ejemplo: precisión del 80%. Significa, que pararemos en esa hoja cuando el número de clases clasificadas correctamente sea mayor o igual al 80%.



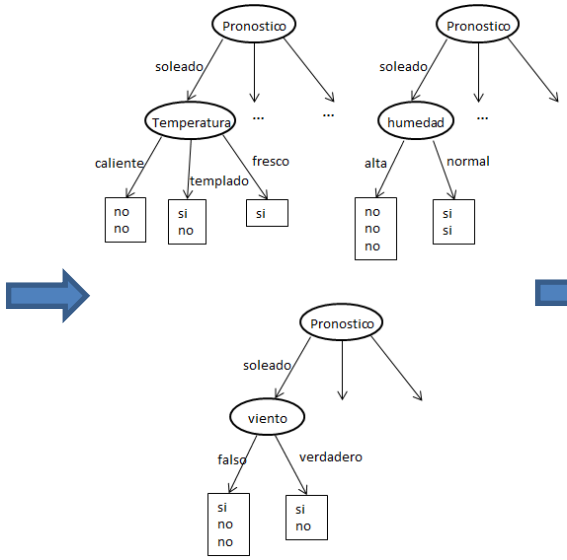
Arboles de decisión:

Construcción

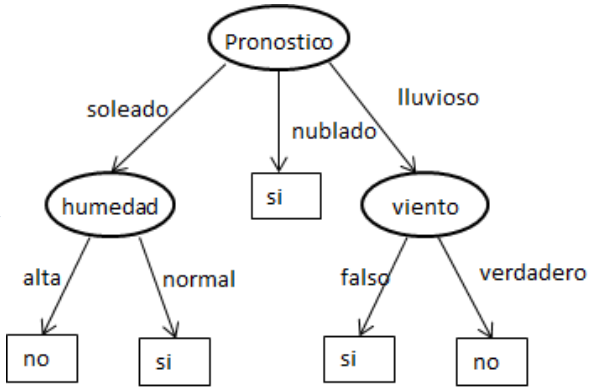
Árbol por atributo



Combinación de un atributo con el resto



Fusión de los Árboles combinados



Podado de un Árbol

¿Cómo decidir si desea reemplazar un nodo interno con una hoja?

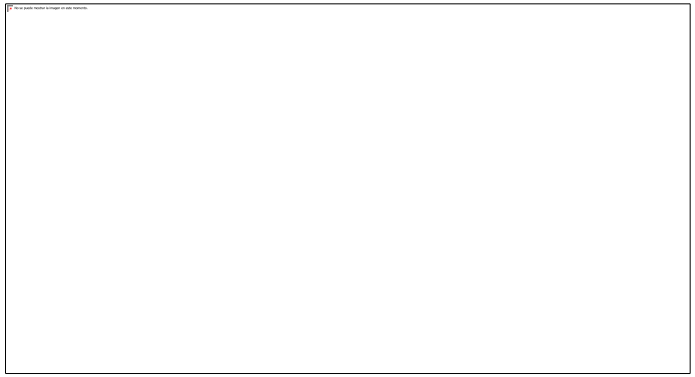
Imaginemos que la verdadera probabilidad de error en el nodo es *q*, y que las *N* instancias son generados por un proceso de Bernoulli con parámetro *q*, de la que *E* son los errores.

El intervalo de confianza viene dado por:

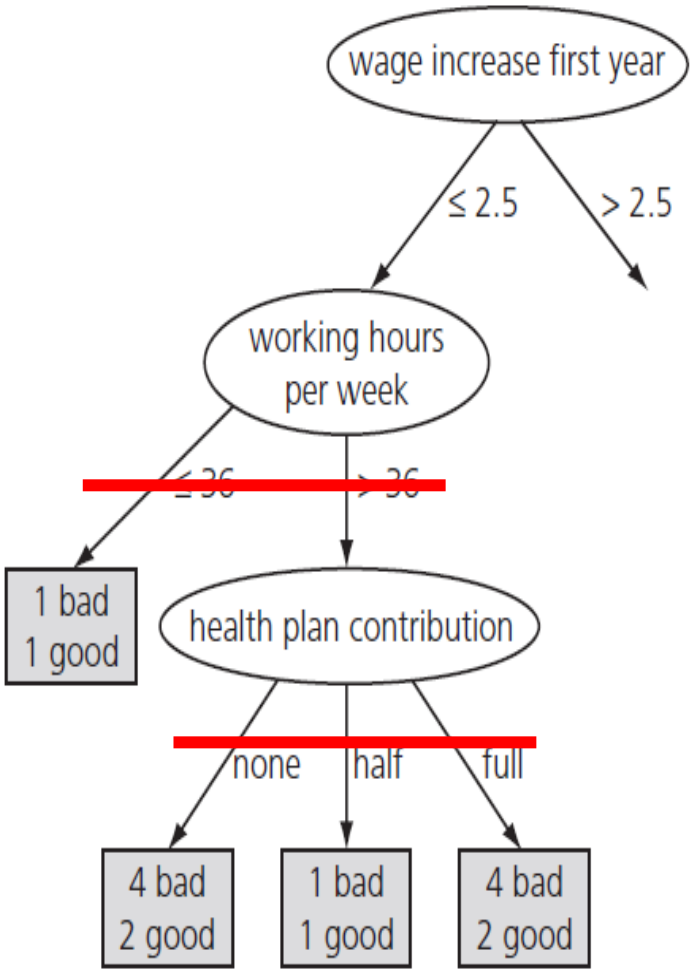
donde *N* es el número de muestras, *f=E/N* es el porcentaje de error observado, y *q* es la tasa de error.

Esto conduce a un límite superior de confianza para *q*.

Ahora usamos ese límite superior de confianza como una **estimación (pesimista) para la tasa de error *e* en el nodo:**



Podado de un Árbol



None: E = 2, N = 6, y por lo que f = 0,33'. e = 0,47. tasa de error de formación es del 33%, se utilizará la estimación pesimista del 47%.
Half: E = 1, N = 2, e = 0.72.
Full: Tiene el mismo valor de *e* como el primero.

El siguiente paso es combinar las estimaciones de error para estos tres hojas en la relación entre el número de ejemplos que se refieren, 6: 2: 6, lo que conduce a una estimación de error combinado de 0,51.

Health plan contribution: f = 5/14. e = 0.46. Debido a que este es menor que el error de estimación combinada de los tres niños, ellos no se podan.

Working hours per week: La estimación de error para la primera, con E = 1 y N = 2, es e = 0,72, y para el segundo es e = 0,46. La combinación de estos, 2 : 14, conduce a un valor que es mayor que la estimación del error para el nodo de horas de trabajo, por lo que el subárbol se poda y se sustituye por un nodo hoja.

Basados en reglas: Tablas de decisión

Es la forma más simple y más rudimentaria para representar la salida de la máquina de aprendizaje.

Pronostico	Temperatura	Humedad	Viento	Jugar
soleado	caliente	alta	falso	no
soleado	caliente	alta	verdadero	no
nublado	caliente	alta	falso	si
lluvioso	templado	alta	falso	si
lluvioso	fresco	normal	falso	si
lluvioso	fresco	normal	falso	si
nublado	fresco	normal	verdadero	si
soleado	templado	alta	falso	no
soleado	fresco	normal	falso	si
lluvioso	templado	normal	falso	si
soleado	templado	normal	verdadero	si
nublado	templado	alta	verdadero	si
nublado	caliente	normal	falso	si
lluvioso	templado	alta	verdadero	no

Deducción de reglas rudimentarias

Relation: weather.symbolic

No.	1: outlook Nominal	2: temperature Nominal	3: humidity Nominal	4: windy Nominal	5: **play** Nominal
3	overcast	hot	high	FALSE	yes
7	overcast	cool	normal	TRUE	yes
12	overcast	mild	high	TRUE	yes
13	overcast	hot	normal	FALSE	yes
4	rainy	mild	high	FALSE	yes
5	rainy	cool	normal	FALSE	yes
6	rainy	cool	normal	TRUE	no
10	rainy	mild	normal	FALSE	yes
14	rainy	mild	high	TRUE	no
1	sunny	hot	high	FALSE	no
2	sunny	hot	high	TRUE	no
8	sunny	mild	high	FALSE	no
9	sunny	cool	normal	FALSE	yes
11	sunny	mild	normal	TRUE	yes

Evaluando los atributos de los datos

	Attribute	Rules	Errors	Total errors
1	outlook	sunny → no	2/5	4/14
		overcast → yes	0/4	
		rainy → yes	2/5	
2	temperature	hot → no*	2/4	5/14
		mild → yes	2/6	
		cool → yes	1/4	
3	humidity	high → no	3/7	4/14
		normal → yes	1/7	
4	windy	false → yes	2/8	5/14
		true → no*	3/6	

Posible atributo

```
outlook: sunny    → no
         overcast → yes
         rainy    → yes
```

Reglas

Tareas de MD

Macro-algoritmo

```
For each attribute A,
   For each value VA of the attribute, make a rule as
   follows:
      count how often each class appears
      find the most frequent class Cf
      create a rule when A=VA;class attribute value = Cf
   End For-Each
   Calculate the error rate of all rules
   End For-Each
Chose the rule with the smallest error rate
```

Modelización estadística

Datos de tiempo

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

Cual es la probabilidad de Jugar (o no) dado xxx??

Comportamiento atributos

Outlook			Temperature			Humidity			Windy			Play	
	yes	*no*		*yes*	*no*		*yes*	*no*		*yes*	*no*	*yes*	*no*
sunny	2	3	hot	2	2	high	3	4	false	6	2	9	5
overcast	4	0	mild	4	2	normal	6	1	true	3	3		
rainy	3	2	cool	3	1								
sunny	2/9	3/5	hot	2/9	2/5	high	3/9	4/5	false	6/9	2/5	9/14	5/14
overcast	4/9	0/5	mild	4/9	2/5	normal	6/9	1/5	true	3/9	3/5		
rainy	3/9	2/5	cool	3/9	1/5								

probabilidades

Modelización estadística

Table 4.3 A new day.

Outlook	Temperature	Humidity	Windy	Play
sunny	cool	high	true	?

Cual es la probabilidad de Jugar (o no) dado xxx??

Probabilidad de que sea **si:** 2/9 x 3/9 x 3/9 x 3/9 x 9/14 = 0.0053

Probabilidad de que sea **no**: 3/5 x 1/5 x 4/5 x 3/5 x 5/14 = 0.0206

Normalización

$$\text{Probability of } yes = \frac{0.0053}{0.0053 + 0.0206} = 20.5\%,$$

$$\text{Probability of } no = \frac{0.0206}{0.0053 + 0.0206} = 79.5\%.$$

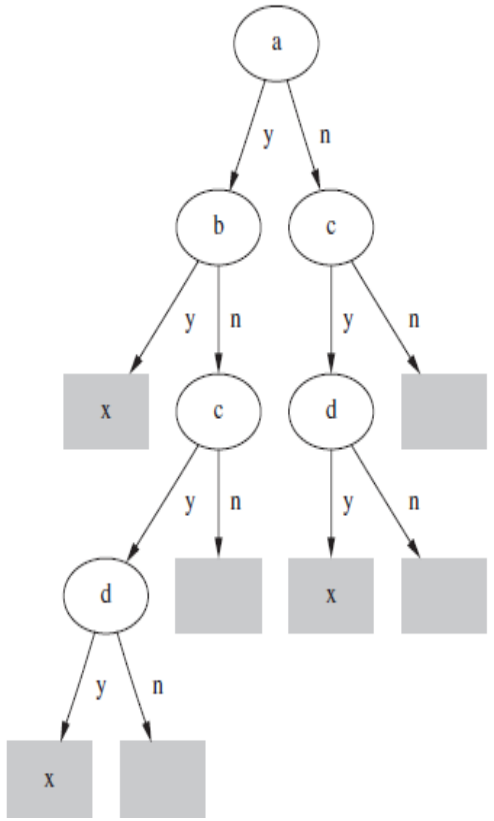
sunny	2/9	3/5	hot	2/9	2/5	high	3/9	4/5	false	6/9	2/5	9/14	5/14
overcast	4/9	0/5	mild	4/9	2/5	normal	6/9	1/5	true	3/9	3/5		
rainy	3/9	2/5	cool	3/9	1/5								

Reglas de clasificación

Las reglas de clasificación son una alternativa popular a los árboles de decisión

Por ejemplo:

```
If outlook = sunny and humidity = high then play = no
If outlook = rainy and windy = true    then play = no
If outlook = overcast                  then play = yes
If humidity = normal                   then play = yes
If none of the above                   then play = yes
```



Utilidad de una categoría

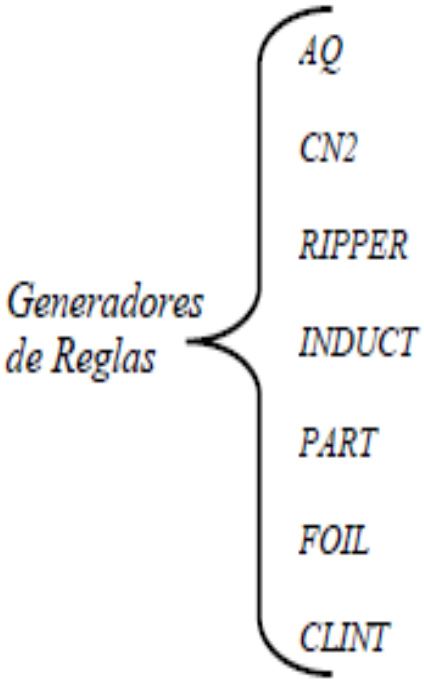
- Mide la calidad general de una partición

$$CU(C_1, C_2, \ldots, C_k) = \frac{\sum_\ell \Pr[C_\ell] \sum_i \sum_j \left(\Pr[a_i = v_{ij} | C_\ell]^2 - \Pr[a_i = v_{ij}]^2 \right)}{k}$$

$\Pr[a_i = v_{ij} \mid C_\ell]$ es una estimación de la probabilidad de que el atributo a_i tiene un valor v_{ij}, en el grupo C_l

donde C_1, C_2,. . ., C_k son los k grupos; la suma exterior es de los grupos; las siguientes sumas interiores de los atributos a_i y sus posibles valores v_{i1}, v_{i2},. . . .

Algunos Sistemas de Generación de reglas



- Algunas reglas inducidas pueden derivar de la construcción de un árbol de decisión, siendo primero generado el árbol de decisión y después trasladado a un conjunto de reglas

- Otros algoritmos se basan en el uso de técnicas de aprendizaje con lógica de predicados (ILP, Inductive Logic Programming). (FOIL, FFOIL, CLINT, etc.)

Basados en distancia

- El principio de los algoritmos basados en distancia viene definido por la distancia entre ejemplos, una distancia pequeña podría significar **alta similaridad** y viceversa.

- Para realizar el entrenamiento de dichos algoritmos, se ordenan los ejemplos que están más cercanos entre si.

- Seguidamente, se utiliza aprendizaje supervisado utilizando la etiqueta de los ejemplos para la creación de las clases.

Funciones de distancia

Basados en distancia

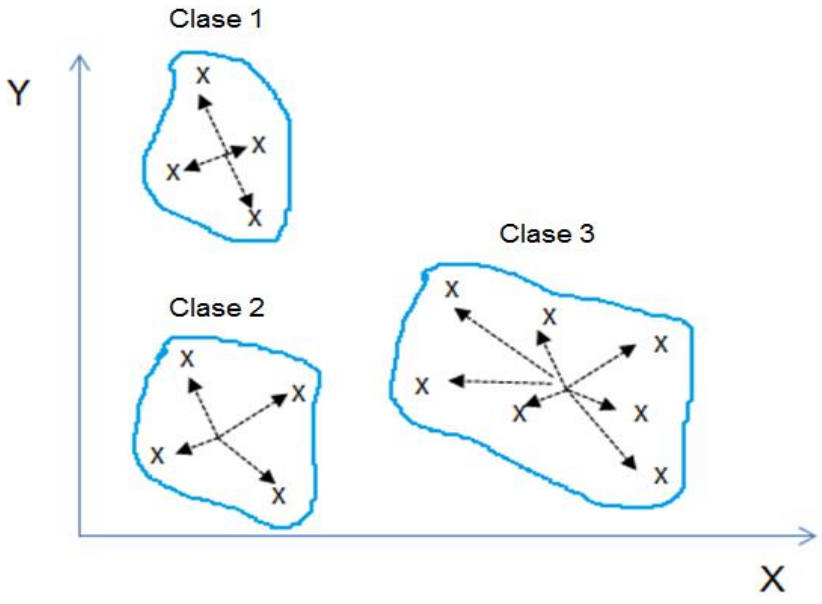
- $u = \{u_1, u_2, \ldots, u_n\}$
- $v = \{v_1, v_2, \ldots, v_n\}$

Cual es la distancia entre *v* y *u*?

Métrica	Expresion
Minkowski	$D(u,v) = \left(\sum_{i=1}^{n} \lvert u_i - v_i \rvert^p\right)^{1/p}$
Euclideana	$D(u,v) = \sum_{i=1}^{n} (u_i - v_i)^2$
City-Block	$D(u,v) = \sum_{i=1}^{n} \lvert u_i - v_i \rvert$
Mahalanobis	$D(u,v) = (u_i - v_i)^T S^{-1} (u_i - v_i)$ S^{-1} es la matriz de covarianza de los datos
Pearson	$D(u,v) = (1 - r)/2$ $r = \sum_{i=1}^{n} \left(\frac{x_i - \bar{x}}{\sigma_x}\right)\left(\frac{y_i - \bar{y}}{\sigma_y}\right)$ Donde $\bar{x}$ y $\bar{y}$ son las medias de x y y respectivamente. σ_x y σ_y son las desviaciones estándar de x y y

Basados en distancia

- **Vecino más cercano:** básicamente se trata de detectar quien es(son) el(los) vecino(s) más cercano(s) a un ejemplo, calculando la distancia del mismo a todos los ejemplos del conjunto de entrenamiento.



K-NN

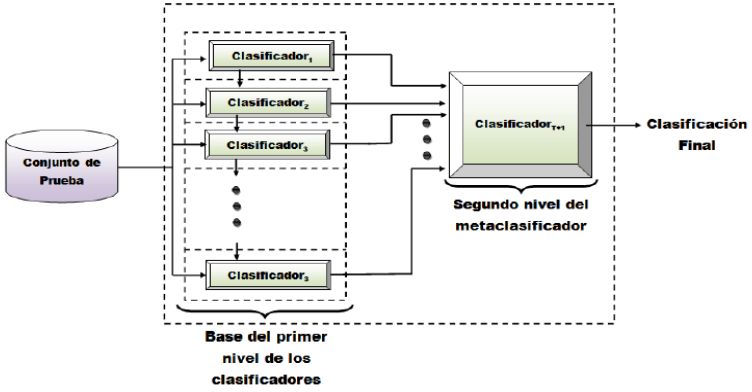
Dada la entrada $x \in T$

1. Encontrar los *K* vecinos más cercanos sobre los datos de entrenamiento $x_1, x_2, \ldots, x_n$ a *x*, dada una métrica de distancia previamente definida.
2. Dado *K* vecinos más cercanos al punto *x*, se hace una votación y se asigna el punto *x* a la clase que obtenga la votación más alta.

Híbridos

- La **combinación de diferentes algoritmos de clasificación** para resolver algunos problemas de MD,

- Los **criterios para combinar** clasificadores son:
  - **Votación por mayoría**
  - **Votación ponderada**:



- **Existen tres esquemas:**
  - **Bagging:** usa diferentes clasificadores, y cada clasificador es entrenado con un conjunto diferente de entrenamiento. Cuando la fase de entrenamiento concluye, cada clasificador da a conocer las clases a las cuales pertenece cada ejemplo. Es aquí donde se utiliza un criterio de votación por mayoría.
  - **Bosting** usa *N* clasificadores iguales, entrenados con un conjunto diferente de entrenamiento, y cada uno de ellos tiene un peso según su efectividad (calidad de entrenamiento). El proceso es similar que el utilizado en bagging, pero en este caso cada voto por ponderación
  - **Stacking** usa un conjunto diferente de clasificadores, entrenados con el mismo conjunto de entrenamiento. Seguidamente se utiliza un criterio de votación por por ponderación según calidad del entrenamiento

Métricas para evaluar un algoritmo de clasificación

Clasificacion	Clasificados positivos	Clasificados negativos
Pos	Verdaderos positivos (tp)	Falsos negativos (fn)
Neg	Falsos positivos (fp)	Verdaderos negativos (tn)

Métrica	Formula
Tasa de error	$\frac{\sum_{i=1}^{n} \frac{tp_i + tn_i}{tp_i + tn_i + fp_i +}}{n}$
Precisión	$\frac{\sum_{i=1}^{n} tp_i}{\sum_{i=1}^{n} (tp_i + fp_i)}$
$Recall_{\mu}$	$\frac{\sum_{i=1}^{n} tp_i}{\sum_{i=1}^{n} (tp_i + fn_i)}$

ASOCIACION

ASOCIACION

Es el descubrimiento de relaciones entre las características (atributos) que conforman la base de datos,

Dichas asociaciones es el conocimiento

REGLAS DE ASOCIACION

Técnica no supervisada que permite predecir patrones de comportamientos futuros **basado en las ocurrencias simultaneas** de valores de variables.

Una asociación entre dos atributos ocurre cuando la **frecuencia con la que se dan dos o más valores determinados de cada uno conjuntamente es relativamente alta**.

Las reglas de asociación intentan descubrir asociaciones o conexiones entre objetos.

$$Consecuencia \Leftarrow Antecedente_1\ Antecedente_2 \ldots Antecedente_m.$$

Ejemplo, en un supermercado se analiza si los pañales y las compotas se compran conjuntamente.

REGLAS DE ASOCIACION: ejemplo

Gestión Estantes del supermercado.

- **Objetivo:** Identificar los elementos que compran juntos muchos clientes.
- **Enfoque:** encontrar dependencias entre elementos.
- **Un ejemplo de regla:**
  - Si un cliente compra pañales y leche, entonces es muy probable que compre compotas.

Reglas de Asociación

Reglas que implican relaciones

Sombreado: parado (standing)
No sombreado: acostado (lying)

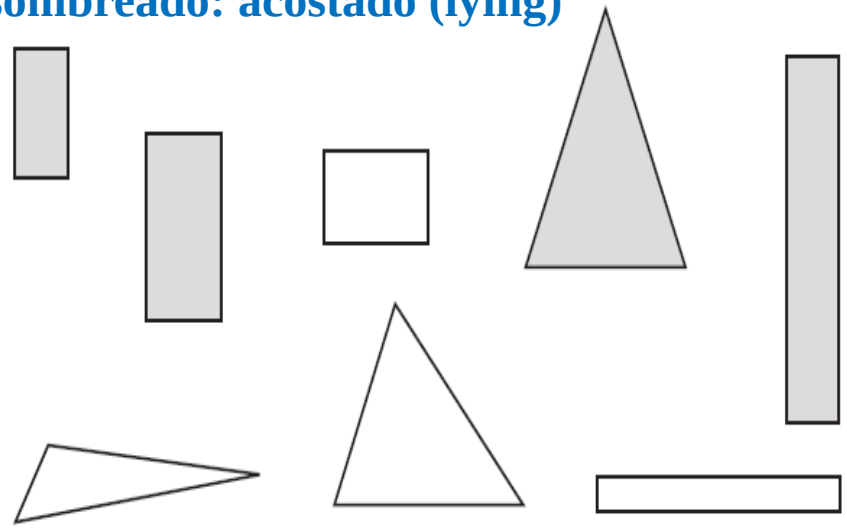


Tabla con datos de entrenamiento

Width	Height	Sides	Class
2	4	4	standing
3	6	4	standing
4	3	4	lying
7	8	3	standing
7	6	3	lying
2	9	4	standing
9	1	4	lying
10	2	3	lying

Reglas

if width ≥ 3.5 and height < 7.0 then lying
if height ≥ 3.5 then standing

Reglas de Asociación

- Pueden predecir cualquier atributo, o combinaciones de atributos.
- **La *cobertura*** de una regla de asociación es el número de instancias para las cuales ella predice correctamente (***soporte***).
- **La *precisión* (*confianza*)** es el número de instancias que predice correctamente, expresado como una proporción de todas las instancias a las que se aplica.

Pronostico	Temperatura	Humedad	Viento	Jugar
soleado	caliente	alta	falso	no
soleado	caliente	alta	verdadero	no
nublado	caliente	alta	falso	si
lluvioso	templado	alta	falso	si
lluvioso	fresco	normal	falso	si
lluvioso	fresco	normal	falso	si
nublado	fresco	normal	verdadero	si
soleado	templado	alta	falso	no
soleado	fresco	normal	falso	si
lluvioso	templado	normal	falso	si
soleado	templado	normal	verdadero	si
nublado	templado	alta	verdadero	si
nublado	caliente	normal	falso	si
lluvioso	templado	alta	verdadero	no

Se utilizan para descubrir **hechos que ocurren en común** dentro de un determinado conjunto de datos

Por ejemplo, en la tabla anterior las reglas:

- **Si temperatura = fria entonces humedad = normal**
- **Si viendo = falso y jugar = no entonces pronostico = soleado y humedad = alta**

Reglas de Asociación

Reglas con diferentes antecedentes y valores de cobertura (>1)

	One-item sets	Two-item sets	Three-item sets	Four-item sets
1	outlook = sunny (5)	outlook = sunny temperature = mild (2)	outlook = sunny temperature = hot humidity = high (2)	outlook = sunny temperature = hot humidity = high play = no (2)
2	outlook = overcast (4)	outlook = sunny temperature = hot (2)	outlook = sunny temperature = hot play = no (2)	outlook = sunny humidity = high windy = false play = no (2)
3	outlook = rainy (5)	outlook = sunny humidity = normal (2)	outlook = sunny humidity = normal play = yes (2)	outlook = overcast temperature = hot windy = false play = yes (2)
4	temperature = cool (4)	outlook = sunny humidity = high (3)	outlook = sunny humidity = high windy = false (2)	outlook = rainy temperature = mild windy = false play = yes (2)
5	temperature = mild (6)	outlook = sunny windy = true (2)	outlook = sunny humidity = high play = no (3)	outlook = rainy humidity = normal windy = false play = yes (2)
	…	…	…	…

	One-item sets	Two-item sets	Three-item sets	Four-item sets
…		…	…	
38		humidity = normal windy = false (4)	humidity = normal windy = false play = yes (4)	
39		humidity = normal play = yes (6)	humidity = high windy = false play = no (2)	
40		humidity = high windy = true (3)		
…		…		
47		windy = false play = no (2)		

Reglas de Asociación

- **Las reglas se obtienen a partir de valores de las variables**

  humidity = normal, windy = false, play = yes

- **Esto nos lleva a las 7 reglas potenciales:**

  If humidity = normal and windy = false → play = yes 4/4

  If humidity = normal and play = yes → windy = false 4/6

  If windy = false and play = yes → humidity = normal 4/7

  If humidity = normal → windy = false and play = yes 4/6

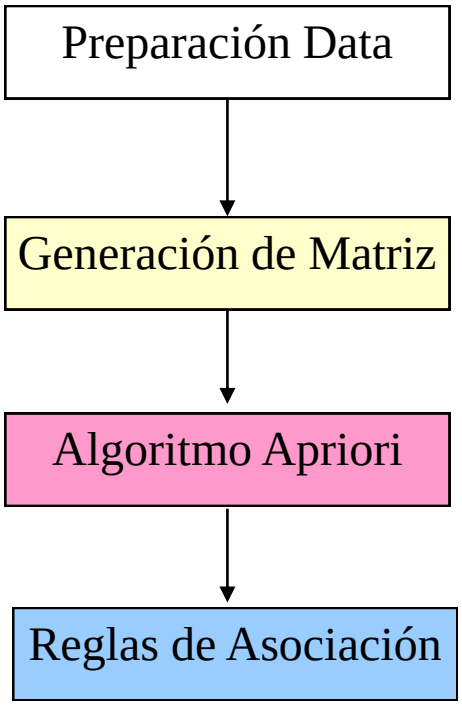
  If windy = false → humidity = normal and play = yes 4/8

  If play = yes → humidity = normal and windy = false 4/9

  If → humidity=normal and windy=false and play=yes 4/12

Método para determinar Reglas de Asociación (Algoritmo Apriori)

Encontrar las asociaciones que se producen entre los diferentes sitios de la página Web cuando los usuarios acceden a ésta.



Reglas de Asociación

Preparación de Data

Registro_Log

id	Id_se...	id_user	ip	Solicitud	fecha	bytes
8	2	11	200.110.86.82	/loginError.jsp	2006-02-26 00:03:00	3641
13	2	11	200.110.86.82	/private/mycourses/	2006-02-26 00:04:00	4785
16	2	11	200.110.86.82	/private/mycourses/website/...	2006-02-26 00:05:00	5717
19	2	11	200.110.86.82	/private/download/1048/3676...	2006-02-26 00:09:00	50688
24	2	11	200.110.86.82	/private/mycourses/website/...	2006-02-26 00:10:00	0
25	2	11	200.110.86.82	/private/mycourses/website/...	2006-02-26 00:10:00	4100
44	2	11	200.110.86.82	/private/mycourses/website/...	2006-02-26 00:19:00	5717
53	2	11	200.110.86.82	/js/tiny_mce/plugins/previe...	2006-02-26 00:21:00	0
110	4	11	200.110.86.82	/js/util.js	2006-02-26 01:03:00	0
176	4	11	200.110.86.82	/private/mycourses/website/...	2006-02-26 01:08:00	8778

Registro_Sesion

id_sesion	id_user	ip	hora_Inicio	hora_fin	num_pag..
3	31	201.2...	2006-02-26 00:54:00	2006-02-26 01:24:00	10
14	30	201.2...	2006-02-26 11:20:00	2006-02-26 11:27:00	9
30	23	200.6...	2006-02-26 16:41:00	2006-02-26 16:43:00	2
38	17	200.2...	2006-02-26 18:46:00	2006-02-26 18:46:00	0
1	6	200.1...	2006-02-26 00:01:00	2006-02-26 00:02:00	1
2	11	200.1...	2006-02-26 00:01:00	2006-02-26 00:29:00	42
7	11	200.1...	2006-02-26 01:36:00	2006-02-26 01:44:00	14
10	3	200.1...	2006-02-26 10:17:00	2006-02-26 10:23:00	3
11	32	201.2...	2006-02-26 10:33:00	2006-02-26 10:33:00	2
13	1	200.1...	2006-02-26 11:14:00	2006-02-26 11:15:00	4

Registro_Paginas

id_pagina	url
2	/index.jsp
7	/private/mybriefc...
16	/private/mycourse...
20	/private/mycourse...
22	/private/mycourse...
26	/private/mycourse...
30	/private/mycourse...
32	/private/myprofil...
35	/public/findUsers...
36	/public/portalDoc...

Reglas de Asociación

Generación Matriz

Sesión / Página	1	2	3	4	5	…..	# sesiones
1	0	1	0	1	0	…..	2
2	1	0	1	1	0	…..	3
3	1	1	0	1	0	…..	3
4	0	1	1	1	0	…..	3
5	1	0	0	0	0	…..	1
6	0	1	0	0	1	…..	2
:	:	:	:	:	:	…..	0
:	:	:	:	:	:	…..	0
# páginas	3	3	2	4	1	…..	

S1= (0+1+1+0+1+0+…+0)/# páginas

Reglas de Asociación

$X \rightarrow Y$

[/public/about.jsp]---->/public/team.jsp

Métricas

Soporte:

Soporte ($X \rightarrow Y$) = Probabilidad ($X \cup Y$)

Confianza:

Confianza ($X \rightarrow Y$) = Probabilidad (X / Y)

prob. condicional

Reglas de Asociación

Algoritmo Apriori (matriz , soporte, confianza)

Usa *conocimiento a priori* de las propiedades de los ítems (páginas) frecuentes que ya se han encontrado.

Premisa: *"Si un conjunto no pasa un test, todos sus súper conjuntos tampoco pasarán el mismo test"*

1

Candidatos antecedentes
(Sop > Soporte)

2

Sesión / Página	1	2	3	4	5	.....	# páginas
1	0	1	0	1	0	.....	0
2	1	0	1	1	0	.....	0
3	1	1	0	1	0	.....	0
4	0	1	1	1	0	.....	0
5	1	0	0	0	0	.....	0
6	0	1	0	0	1	.....	0
:	:	:	:	:	:	.....	0
:	:	:	:	:	:	.....	0
# sesiones	0	1	0	1	0	.....	0

S1 S2 ……S5 ……Sn

3

1→2 Solo si conf(1→2) > confianza confianza=Prob (X / Y)

4

Confianza: hallar esa prob. condicional desde los nodos soportes que pasen un umbral

Reglas de Asociación

Nº	Regla	Soporte	Confianza
1	[/public/about.jsp]---->/public/team.jsp	15,21%	57,14%
2	[/public/findUsers.jsp]---->/public/portalDocument.jsp	13,04%	83,33%
3	[/public/findUsers.jsp]---->/public/team.jsp	13,04%	83,33%
4	[/public/portalDocument.jsp]---->/index.jsp	15,21%	57,14%
5	[/public/portalDocument.jsp]---->/public/team.jsp	15,21%	71,42%
6	[/index.jsp]---->/public/team.jsp	17,39%	62,5%
7	[/loginError.jsp]---->/private/mycourses/index.jsp	10,86%	80%

Modelos de Agrupamiento (Segmentación)

Agrupamiento (Clustering)

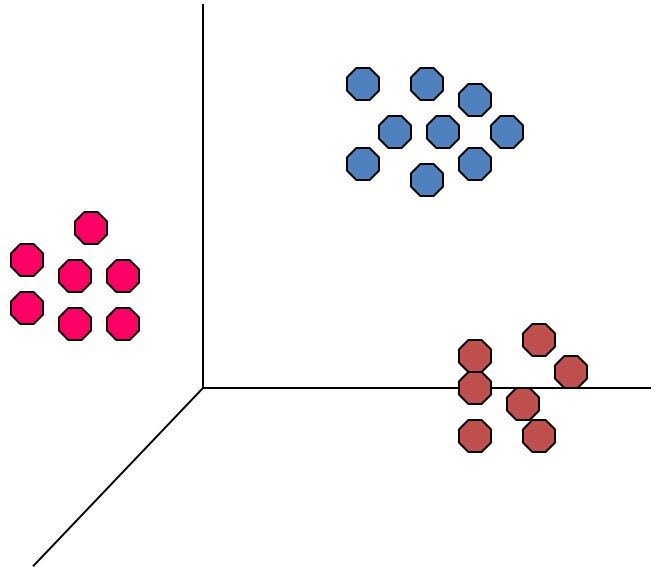
Dado un conjunto de datos, cada una con un conjunto de atributos, y una medida de similitud entre ellos, **encontrar grupos** de tal manera que:

- Los puntos de datos en un clúster son los **más similares** entre sí.
- Los puntos de datos en grupos separados son **menos similares** entre sí.

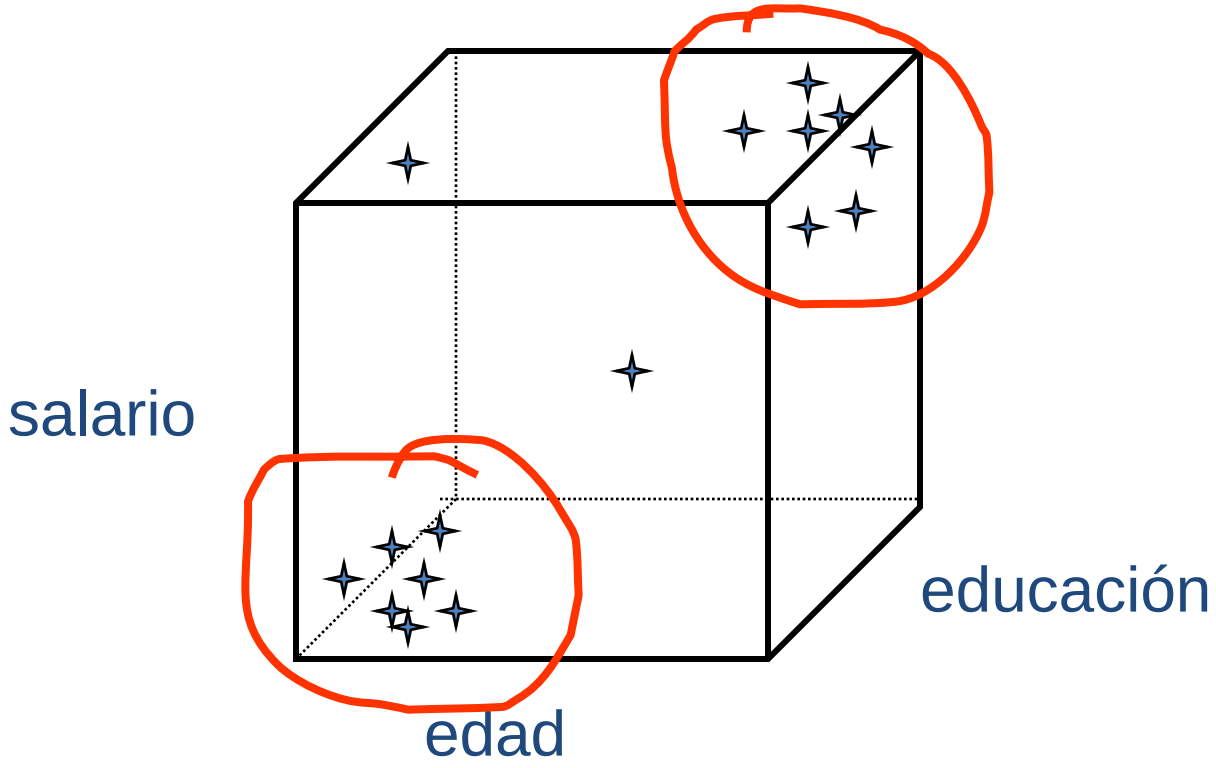
Agrupamiento (Clustering)

Distancias Intracluster son minimizadas

Distancias Intercluster son maximizadas

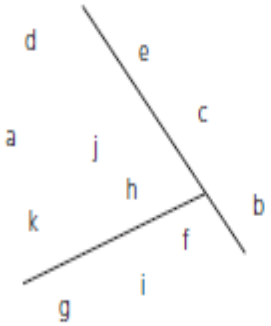


Agrupamiento (Clustering)

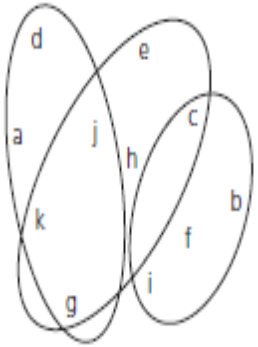


Clusters

Técnica muy usadas son inferir un **árbol de decisión** o un **conjunto de reglas** que asignan a cada instancia **al grupo** al que pertenece



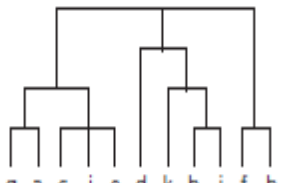
(a)



(b)

	1	2	3
a	0.4	0.1	0.5
b	0.1	0.8	0.1
c	0.3	0.3	0.4
d	0.1	0.1	0.8
e	0.4	0.2	0.4
f	0.1	0.4	0.5
g	0.7	0.2	0.1
h	0.5	0.4	0.1

(c)



(d)

No hay etiquetas de por medio

Ejemplo de Clustering

Agrupación de documento:

- **Objetivo:** encontrar grupos de documentos que son similares entre sí sobre la base de los términos importantes que aparecen en ellos.

- **Enfoque:** Identificar términos que aparecen con frecuencia en cada documento. Formar una medida de similitud basada en las frecuencias de los diferentes términos.

Agrupación de documento

- 3204 Artículos de un periódico.
- **Medida Similitud:** ¿Cuántas palabras son comunes en estos documentos (después de algún tipo de filtrado de palabras).

Categoría	*Total Articulos*	*Grupos*
Financiero	555	36
Extranjero	341	20
Nacional	273	6
Ciudad	943	76
Deportes	738	73
Entretenimiento	354	28

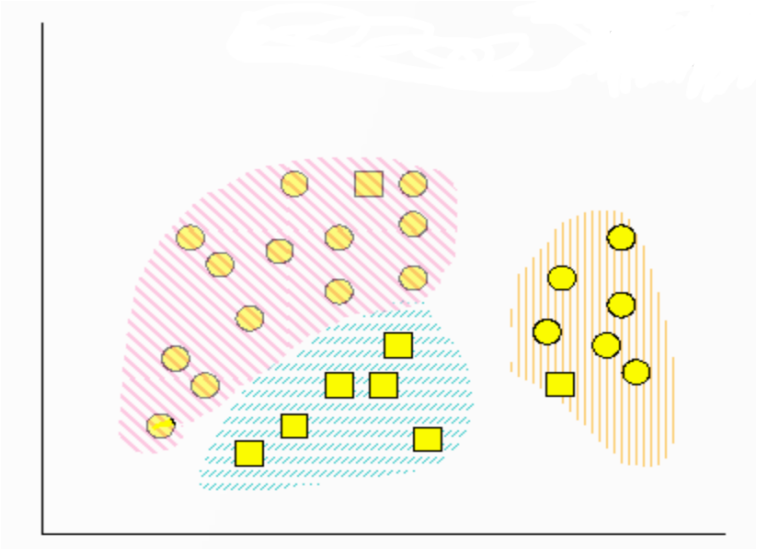
Tipos de clustering

- **Clustering particional**
  - Partición de los objetos en grupos o clusters. Todos los objetos pertenecen a alguno de los k clusters, los cuales son disjuntos. **Problema => elección de k**

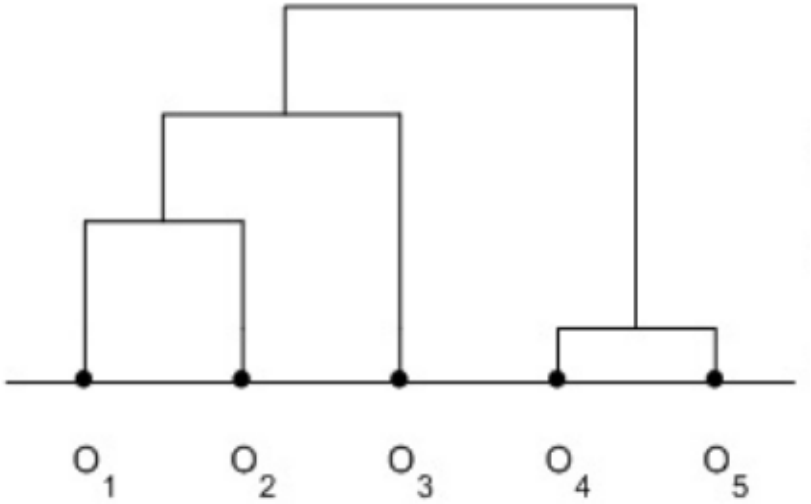
- **Clustering ascendente jerárquico**
  - Crear un dendograma, es decir, crear un conjunto de agrupaciones anidadas hasta construir **un árbol jerárquico**

Clusterización

Dados unos **datos sin etiquetar**, el objetivo es encontrar grupos naturales de instancias



a) Particional



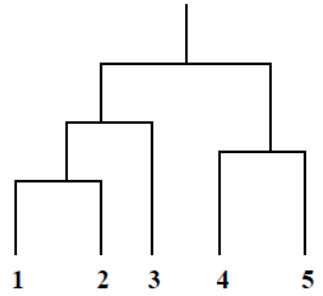
b) Jerárquico

Tipos de clustering

Métodos Jerárquicos

- **Los métodos aglomerativos**: Comienzan con la creación de un cluster para cada uno de los ejemplos individuales, seguidamente se van mezclando en pares los clusters más cercanos hasta que queden K clusters.
- **Los métodos divisivos**: Comienza con la creación de uno o dos cluster que contienen todos los ejemplos, seguidamente se va dividiendo hasta que se crearon K clusters

	I1	I2	I3	I4	I5
I1	1.00	0.90	0.10	0.65	0.20
I2	0.90	1.00	0.70	0.60	0.50
I3	0.10	0.70	1.00	0.40	0.30
I4	0.65	0.60	0.40	1.00	0.80
I5	0.20	0.50	0.30	0.80	1.00



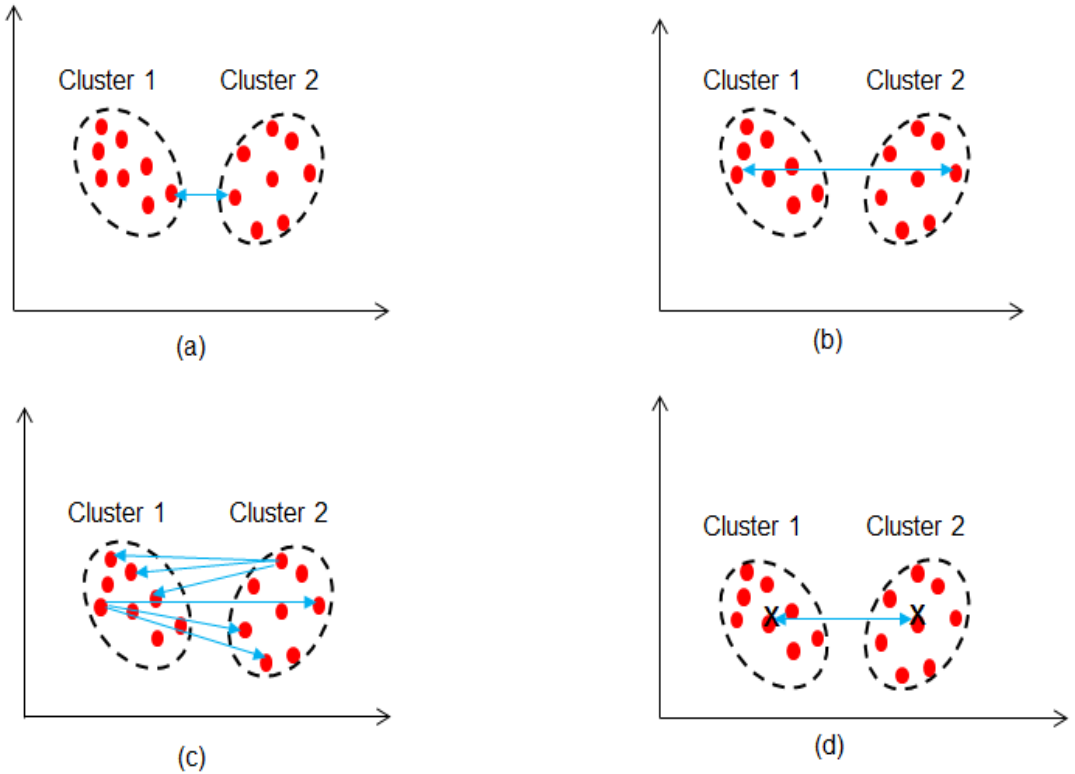
Tipos de clustering: basados en distancia

(a) Distancia mínima

(b) Distancia máxima

(c) Distancia de promedio del grupo

(d) Distancia con respecto al centroide

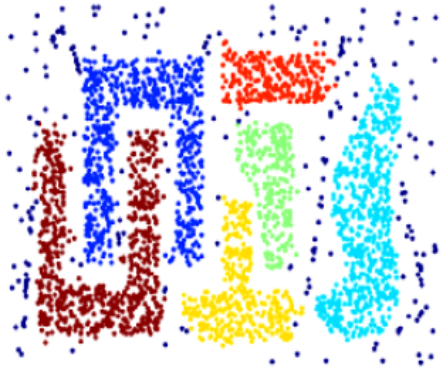


Tipos de clustering

Basados en densidad

Los algoritmos basados en densidad, tratan de formar agrupaciones en áreas con altas densidades de ejemplos

a) Datos originales



b) Datos después de clustering

Tipos de clustering

Algoritmo de DBSCAN

```
current_cluster_label ← 1
for all core points do
  if the core point has no cluster label then
    current_cluster_label ← current_cluster_label + 1
    Label the current core point with cluster label current_cluster_label
  end if
  for all points in the Eps-neighborhood, except i^th the point itself do
    if the point does not have a cluster label then
      Label the point with cluster label current_cluster_label
    end if
  end for
end for
```

1. **Comienza eliminando los puntos de ruido, es decir que no tienen una densidad mayor a un umbral en un radio previamente definido,**
2. **seguidamente se procede a realizar el agrupamiento de los puntos restantes**

Tipos de clustering

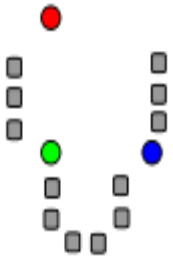
Basados en prototipos

- Un cluster se define por un conjunto de objetos, donde cada objeto esta más cerca (o es más similar) al **prototipo** que define al cluster donde fue asignado que a cualquier otro prototipo de otro cluster existente.

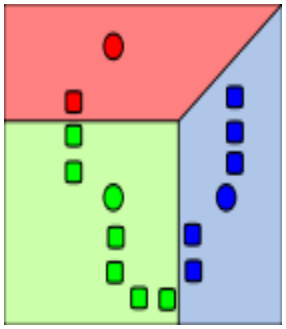
- El prototipo que define al cluster normalmente **denota sus características principales**, por ejemplo, para atributos solo numéricos el prototipo del cluster a menudo se representa como el centroide.

Algoritmo K-medias

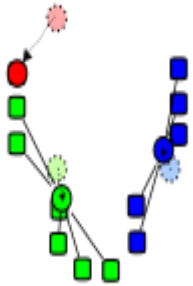
es un método de agrupamiento, que tiene como objetivo la partición de un conjunto (n) en k grupos en el que **cada ejemplo pertenece al grupo más cercano a la media**



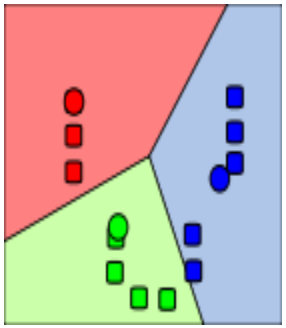
1) *k* centroides iniciales generados aleatoreamente (en este caso *k*=3)



2) *k* grupos son generados asociándole el punto



3) El centroide de cada uno de los *k* grupos se recalcula



4) Pasos 2 y 3 se repiten hasta que se logre la convergencia.

Algoritmo K-medias

1. **Método más utilizado** de clustering particional

2. La idea es situar **los prototipos o centros en el espacio**, de forma que los datos pertenecientes al mismo prototipo tengan características similares

3. Los datos se **asignan a cada centro según la menor distancia**, normalmente usando la distancia euclídea

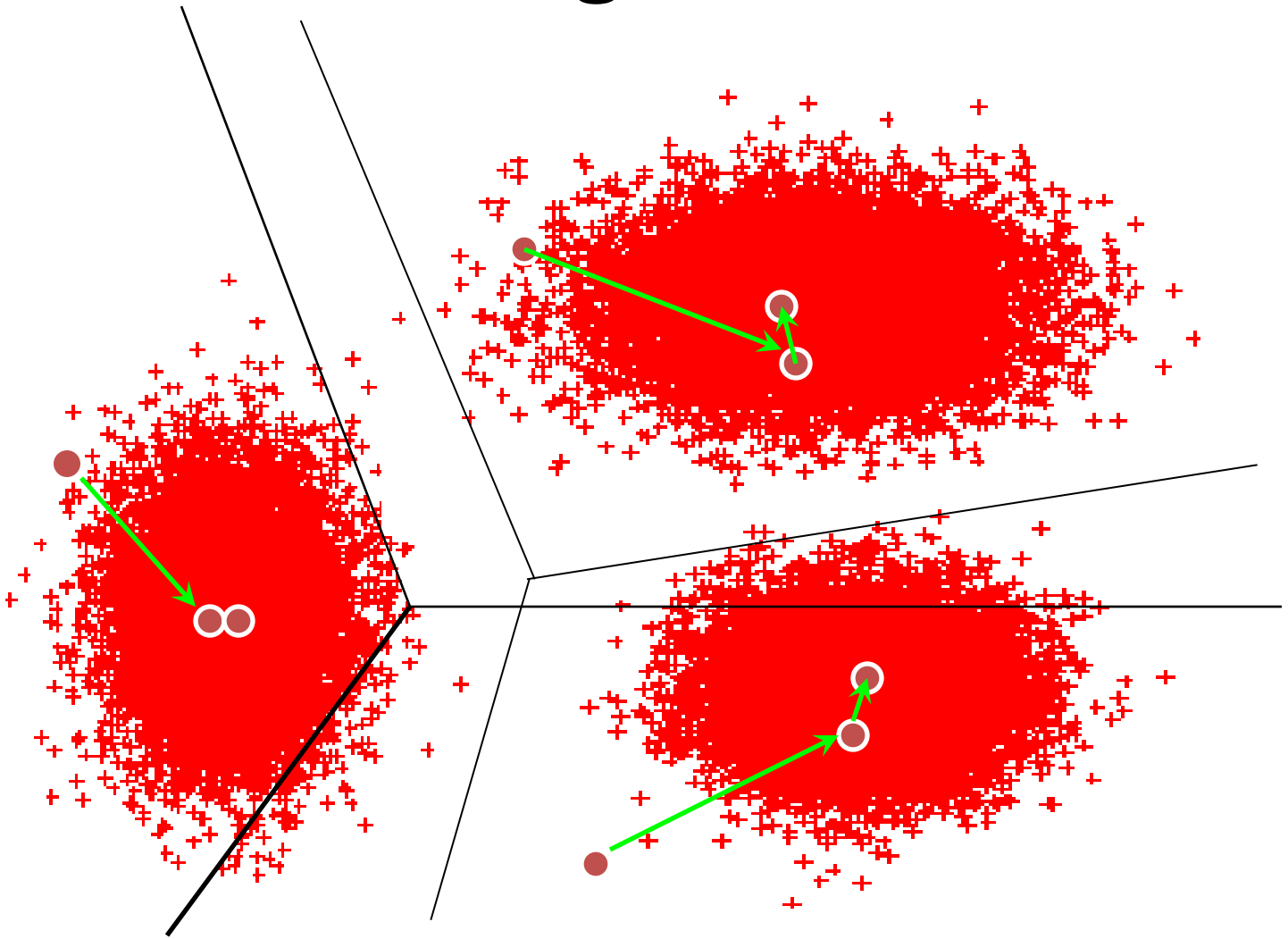
4. Una vez introducidos todos los datos, se **desplazan los prototipos hasta el centro de masas** de su nuevo conjunto, esto se repite hasta que no se desplazan más.

Clusterización: Algoritmo K-Medias

- Seleccionar centroides aleatorios
- Asignar cada objeto al grupo cuyo centroide sea el más cercano al objeto.
- Cuando todos los objetos hayan sido asignados, recalcular la posición de los k centroides.
- Repetir los pasos 2 y 3 hasta que los centroides no varíen

Distancia Euclídea $\delta^2 E(Xi, Xj) = ||Xi - Xj||^2$

Clusterización: Algoritmo K-Medias



Corrida en frio K-means

Input:

- K (number of clusters)
- Training set $\{x^{(1)}, x^{(2)}, \ldots, x^{(m)}\}$

Randomly initialize K cluster centroids $\mu_1, \mu_2, \ldots, \mu_K \in \mathbb{R}^n$

Repeat {

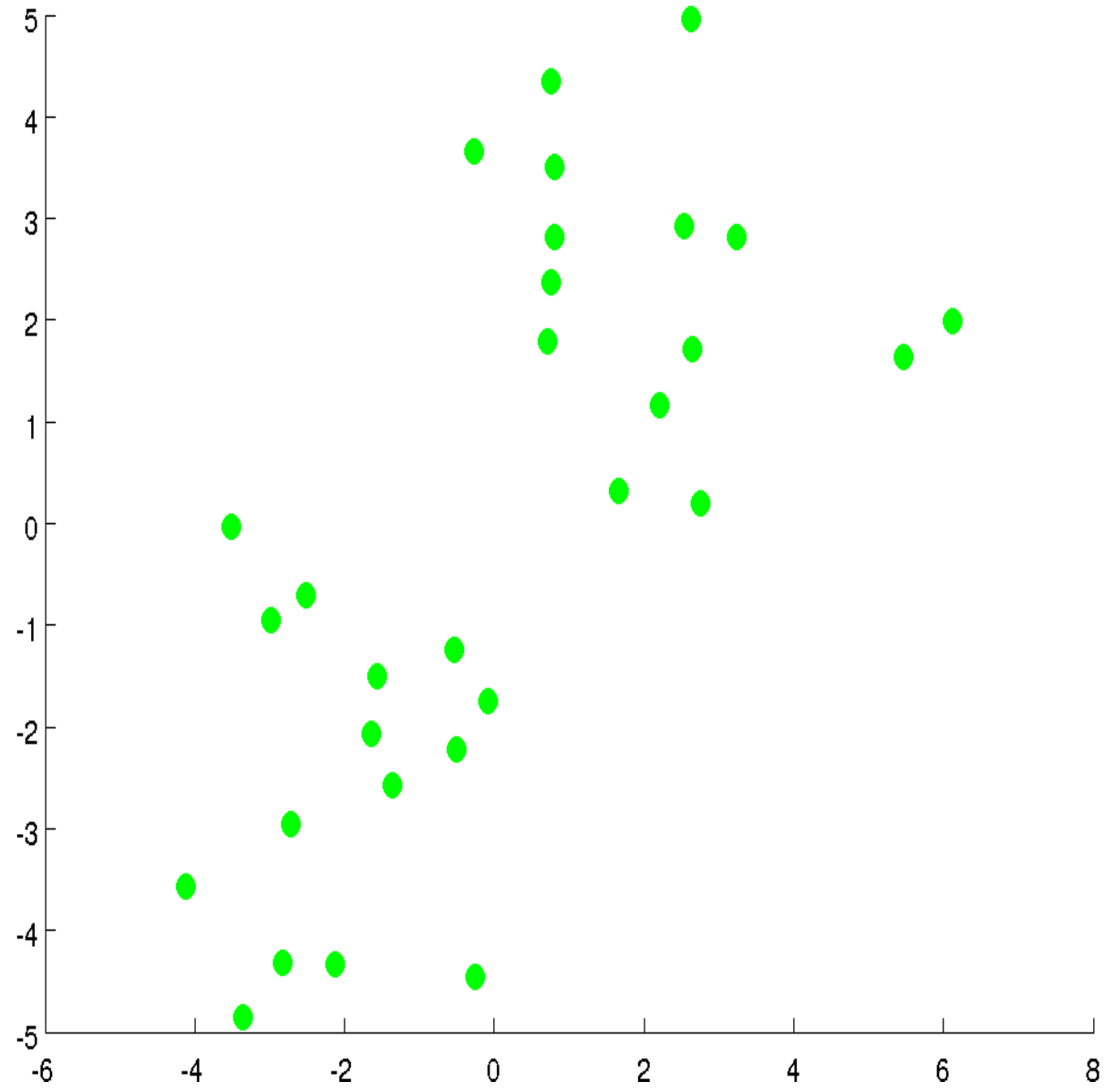
for i =1 to m

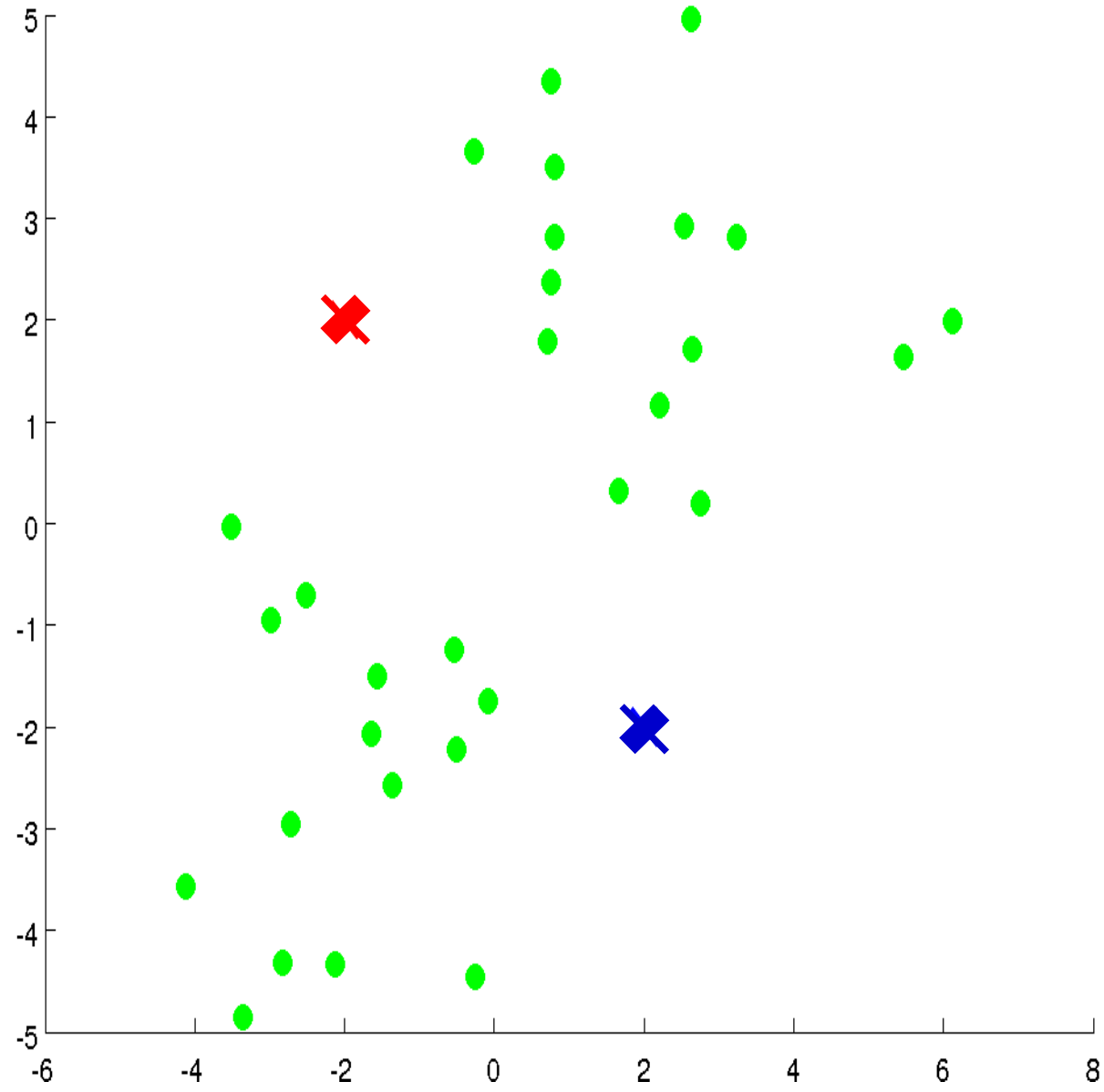
$c^{(i)}$:= index (from 1 to K) of cluster centroid closest to $x^{(i)}$

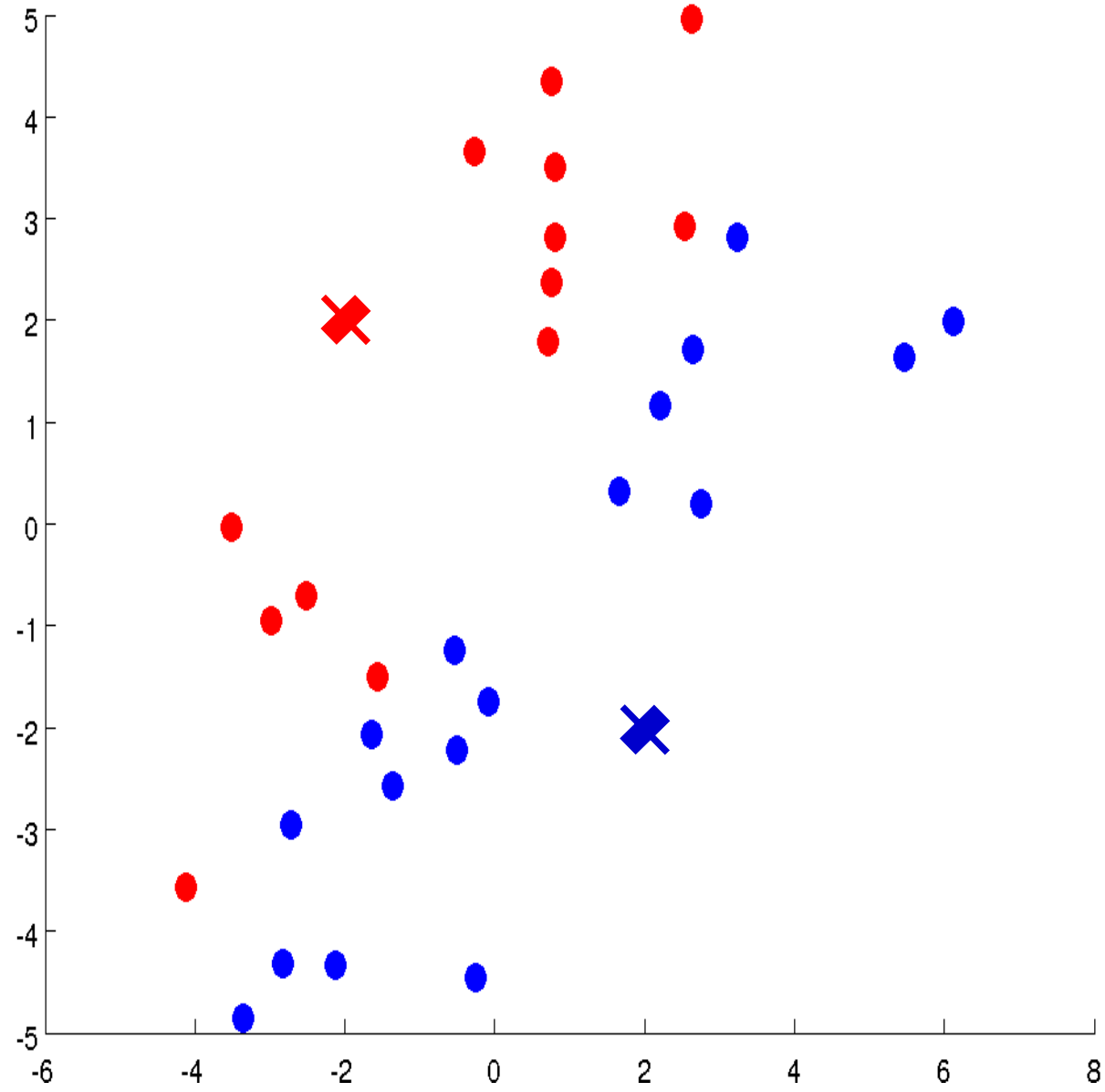
for k = 1 to K

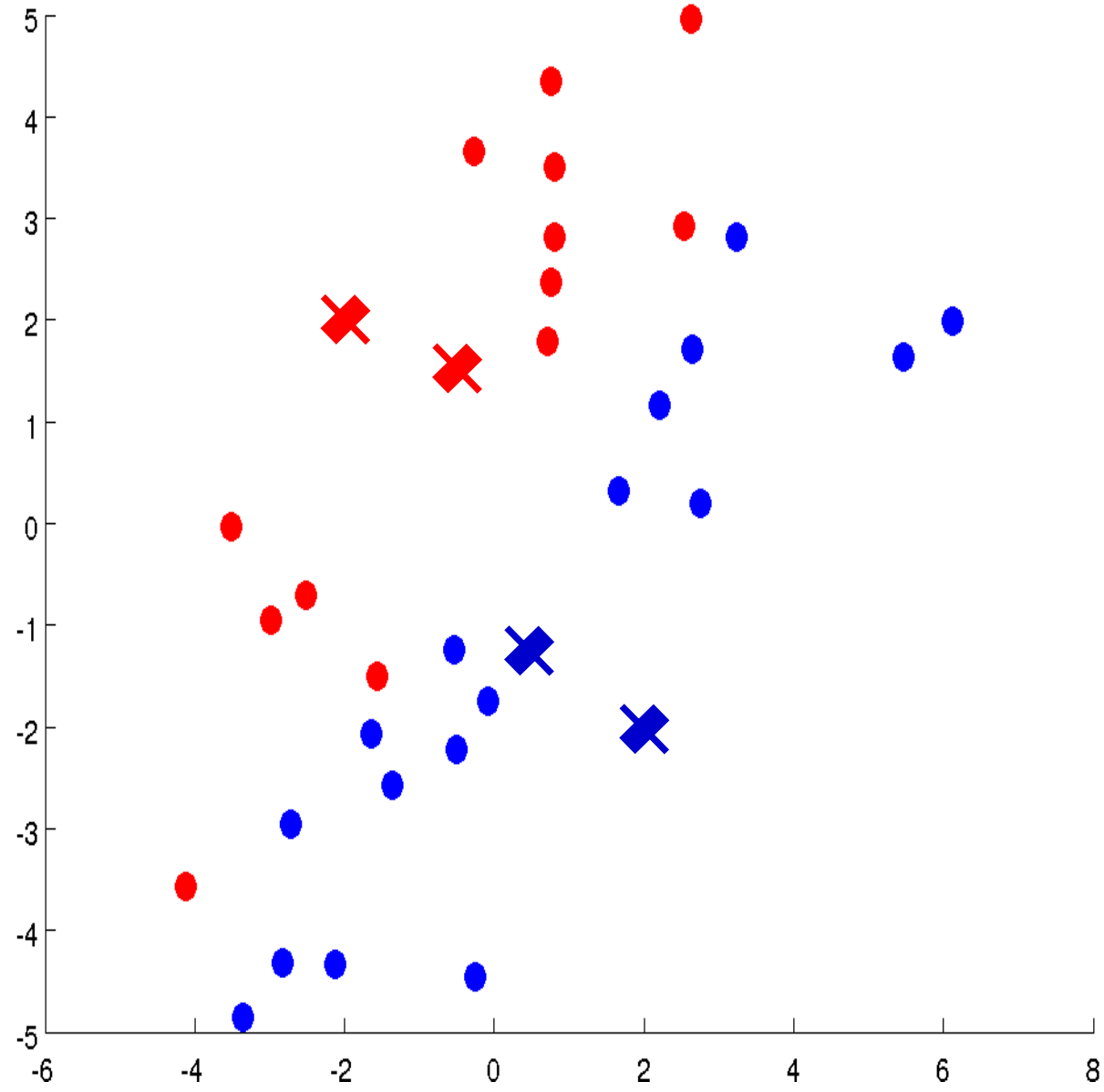
μ_k := average (mean) of points assigned to cluster

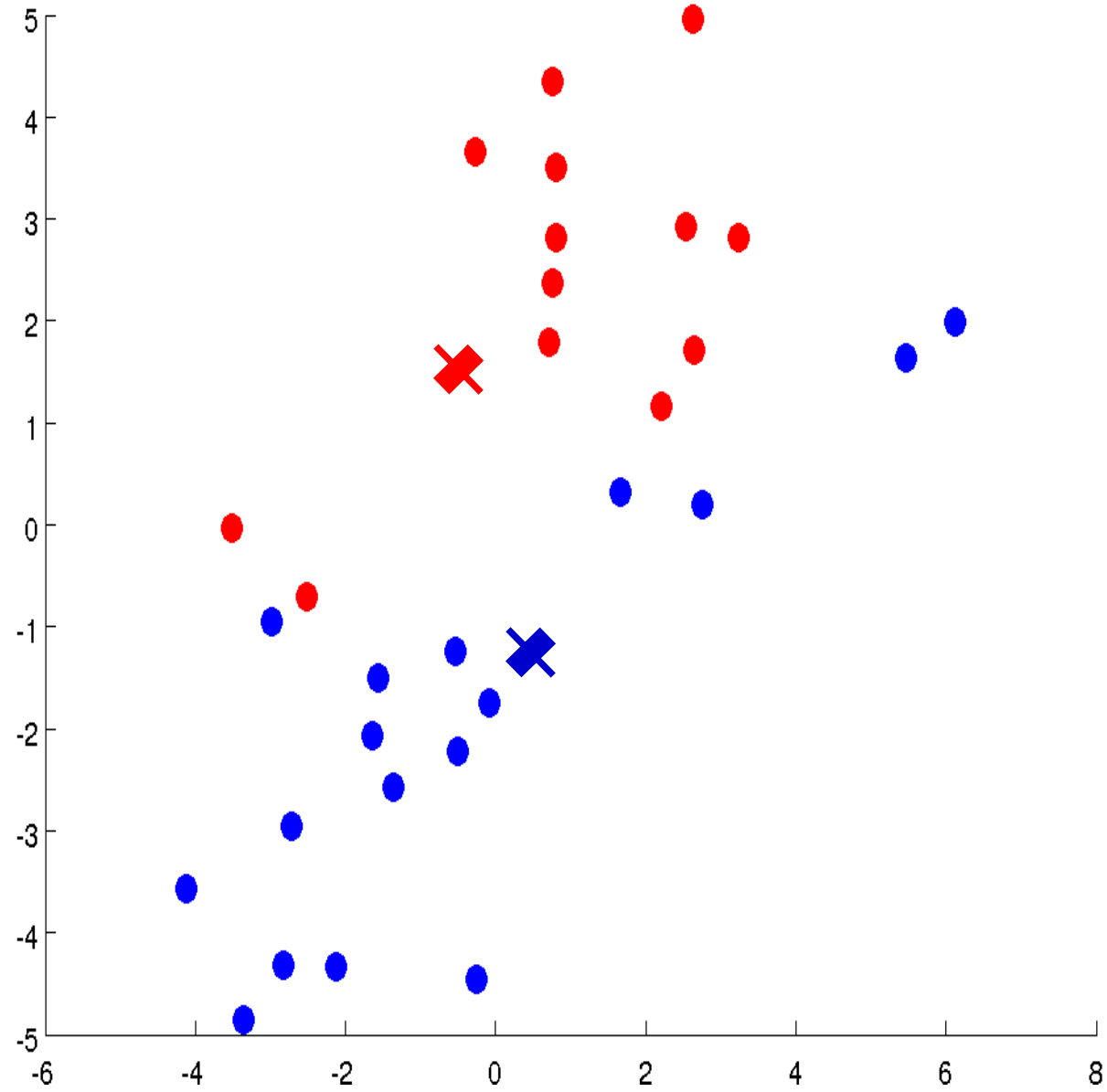
} until convergence criteria is met

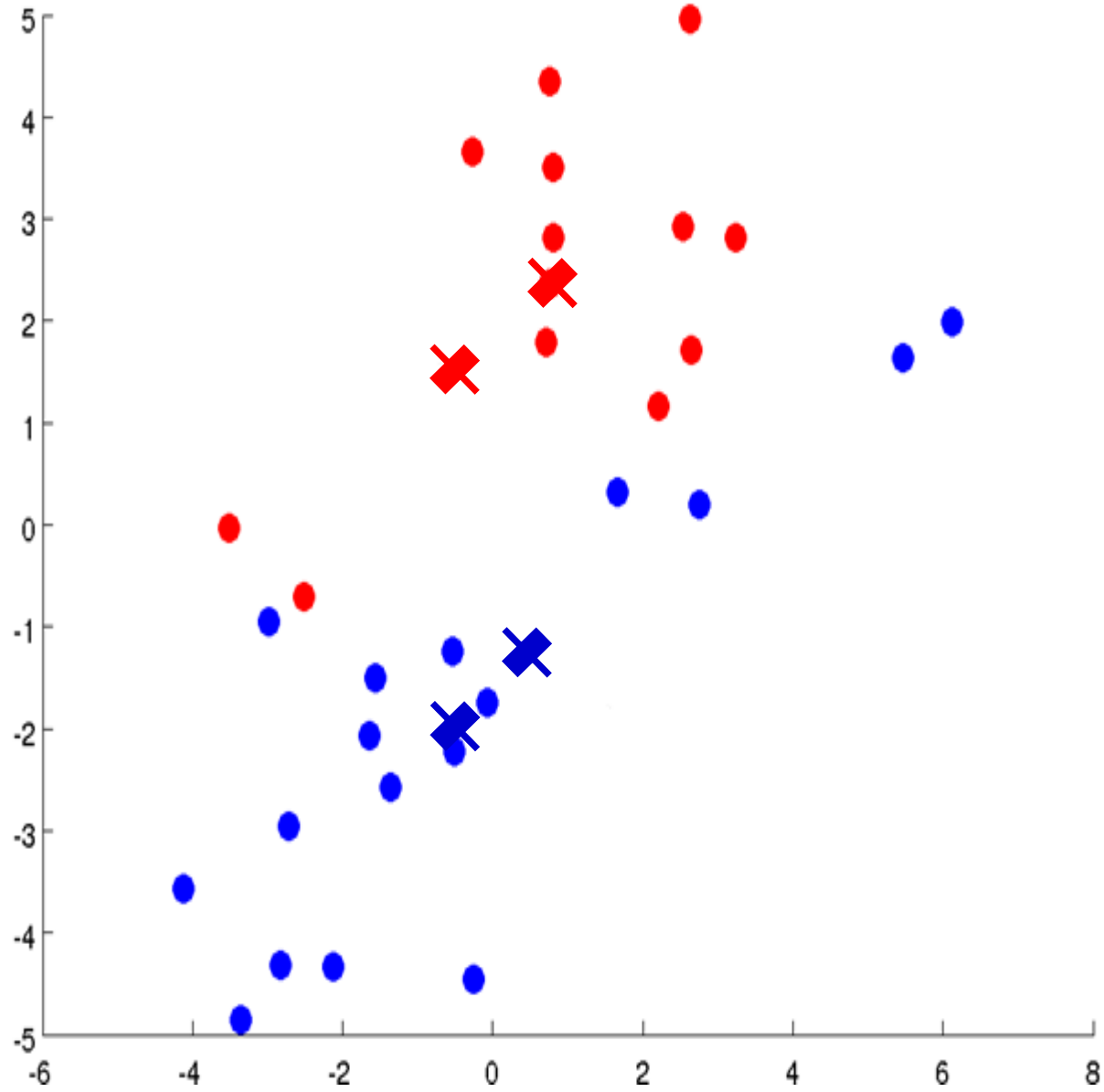

5
4
3
2
1
0
-1
-2
-3
-4
-5
-6
-4
-2
0
2
4
6
8

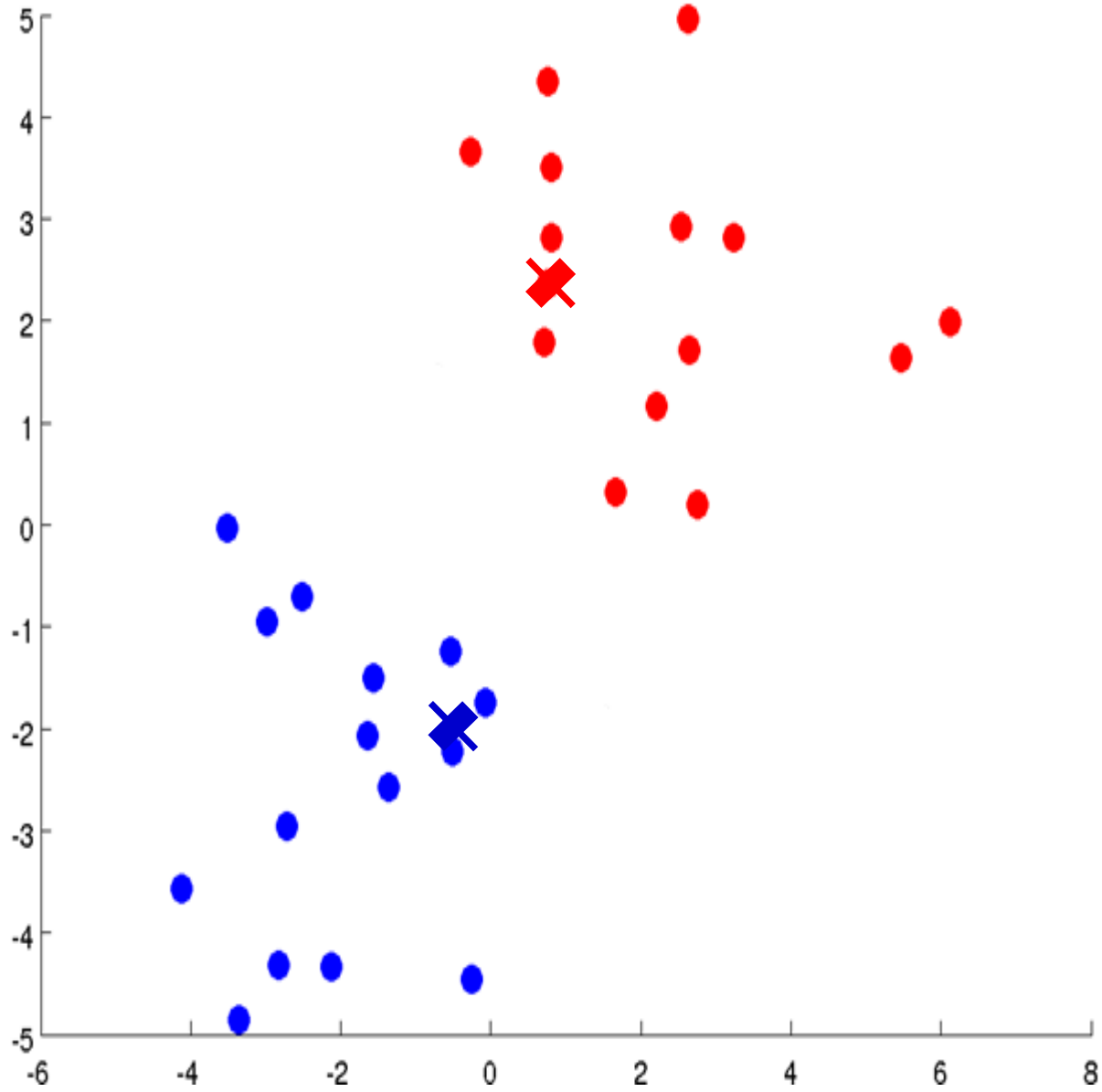

5
4
3
2
1
0
-1
-2
-3
-4
-5
-6
-4
-2
0
2
4
6
8

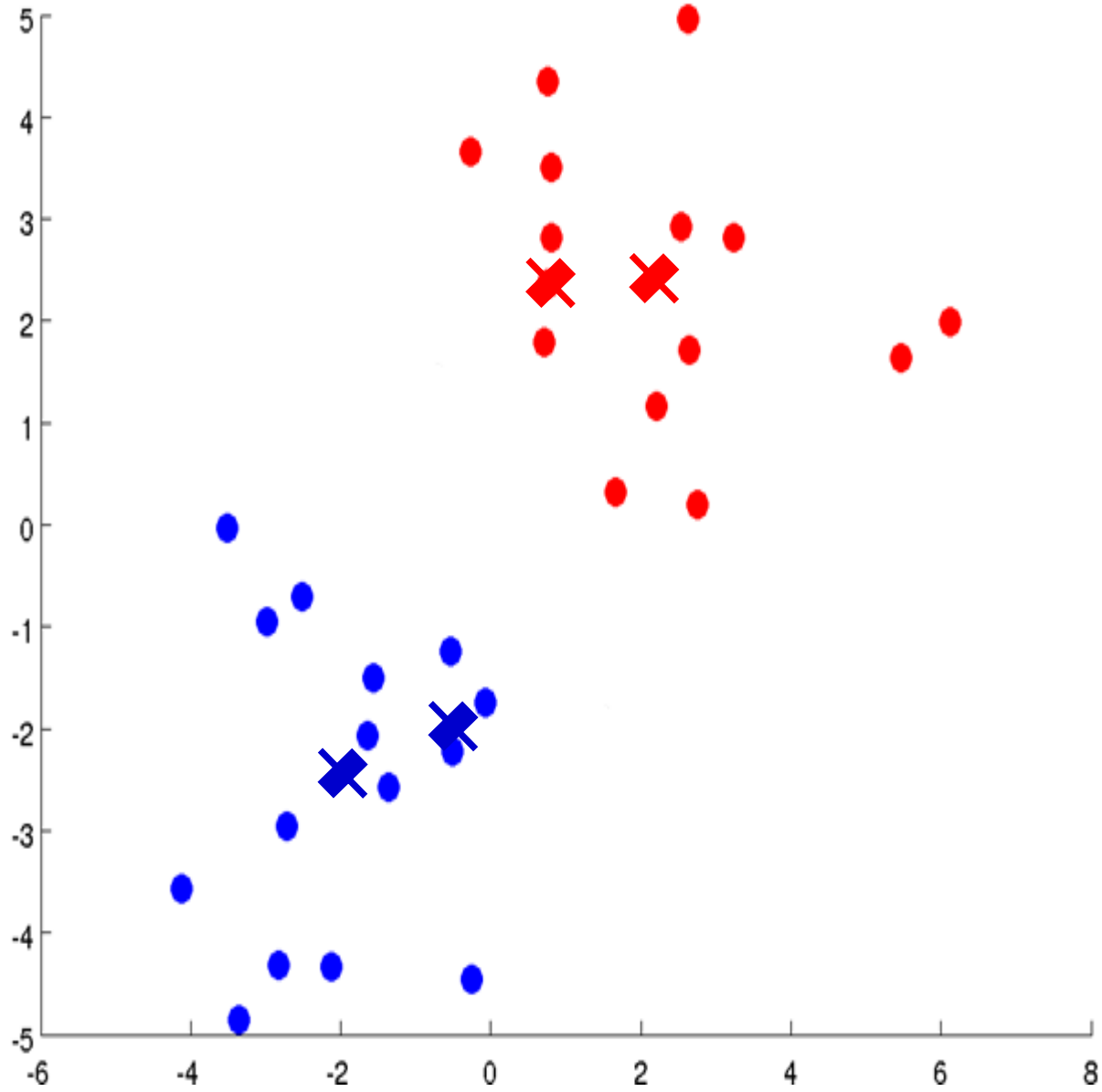

5
4
3
2
1
0
-1
-2
-3
-4
-5
-6
-4
-2
0
2
4
6
8

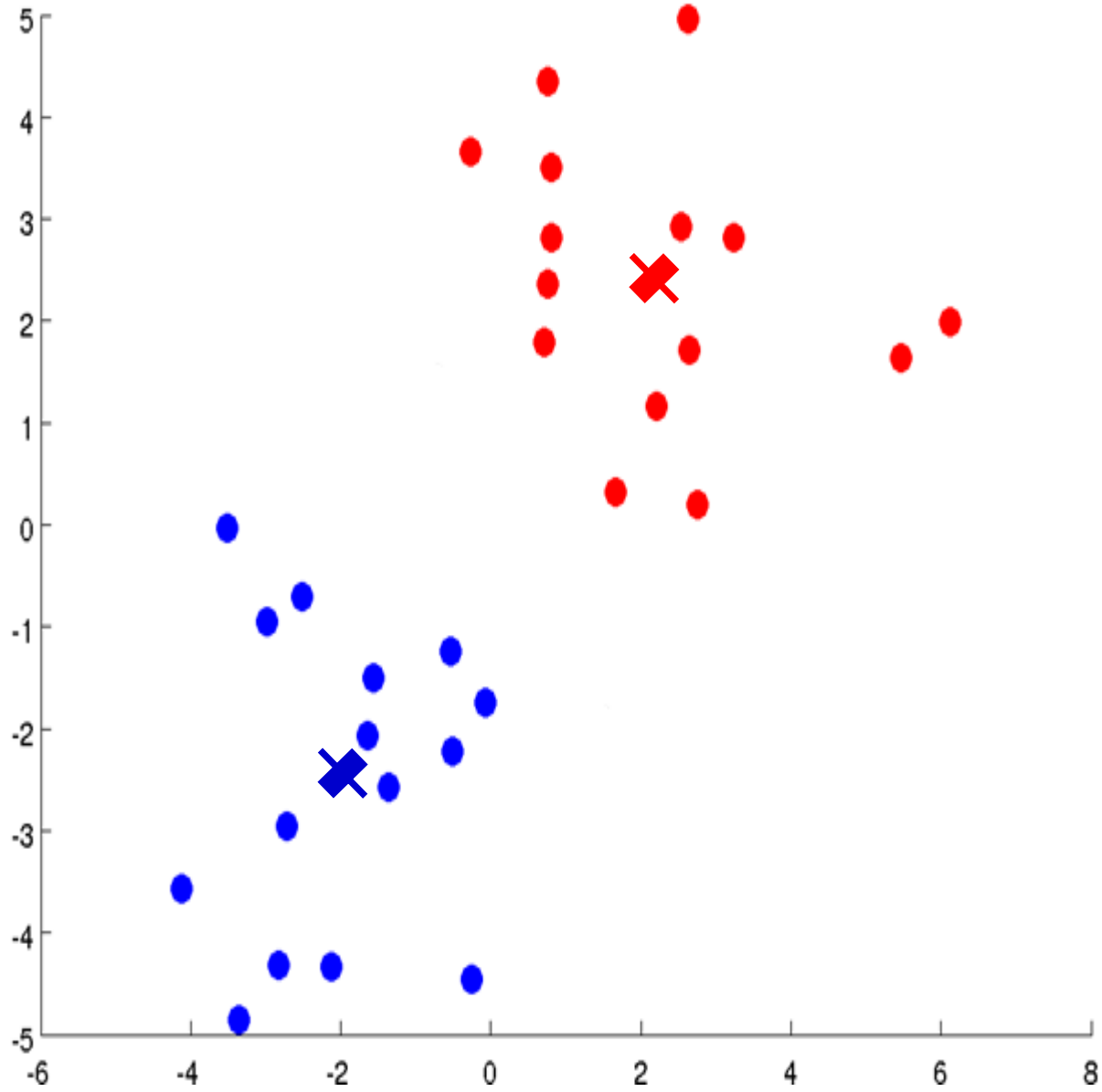

5
4
3
2
1
0
-1
-2
-3
-4
-5
-6
-4
-2
0
2
4
6
8


5
4
3
2
1
0
-1
-2
-3
-4
-5
-6
-4
-2
0
2
4
6
8

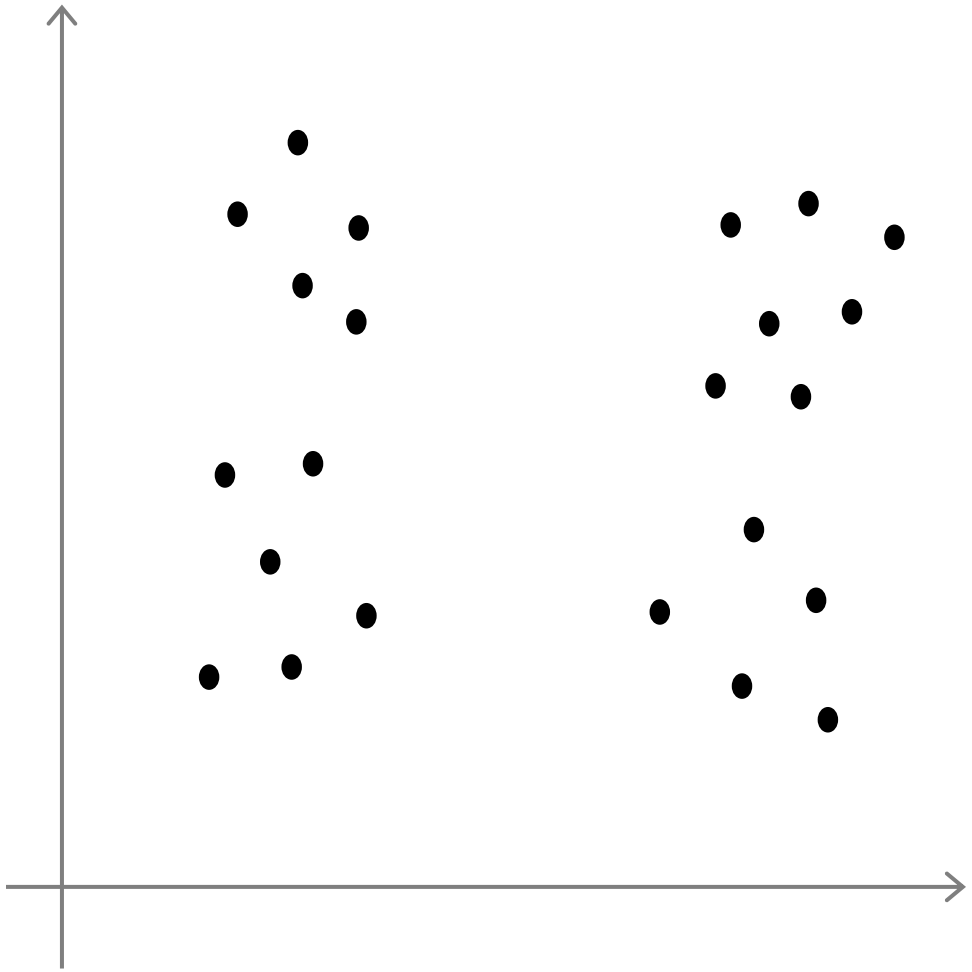

5
4
3
2
1
0
-1
-2
-3
-4
-5
-6
-4
-2
0
2
4
6
8


5
4
3
2
1
0
-1
-2
-3
-4
-5
-6
-4
-2
0
2
4
6
8


5
4
3
2
1
0
-1
-2
-3
-4
-5
-6
-4
-2
0
2
4
6
8

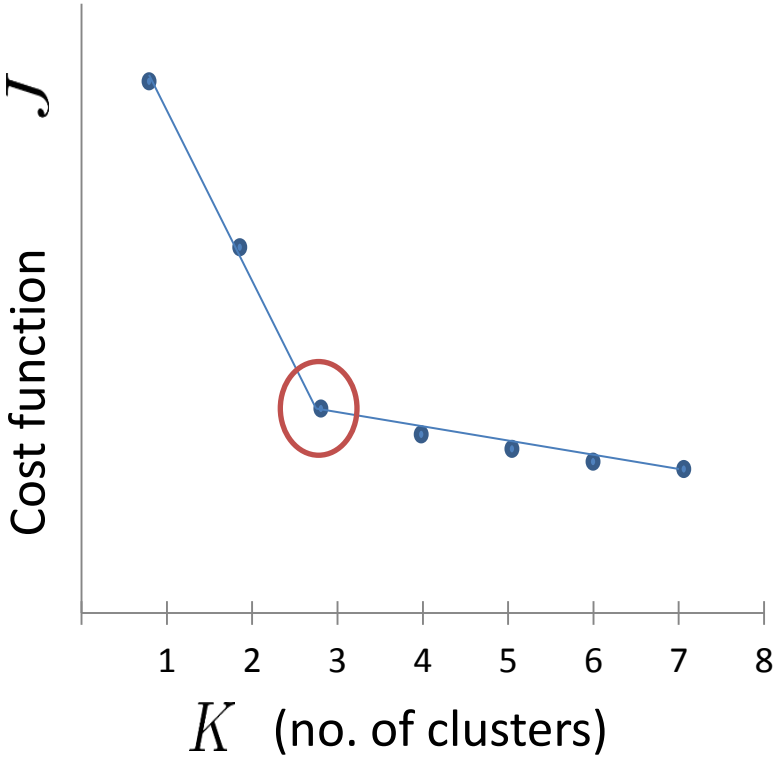

5
4
3
2
1
0
-1
-2
-3
-4
-5
-6
-4
-2
0
2
4
6
8

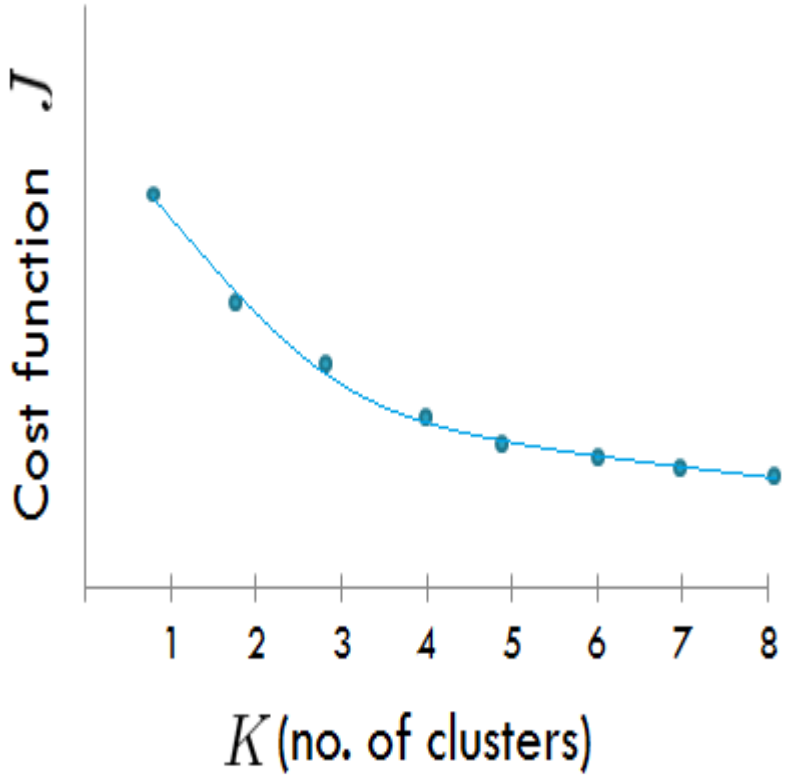
Cual es el correcto valor de K?



Escogiendo el valor de K

Método del codo:





Clustering Incremental

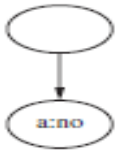
Datos de tiempo

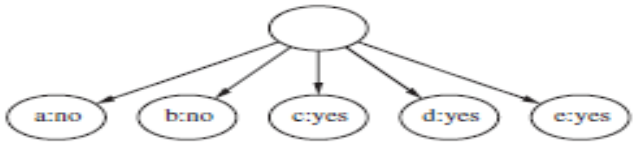
ID code	Outlook	Temperature	Humidity	Windy	Play
a	sunny	hot	high	false	no
b	sunny	hot	high	true	no
c	overcast	hot	high	false	yes
d	rainy	mild	high	false	yes
e	rainy	cool	normal	false	yes
f	rainy	cool	normal	true	no
g	overcast	cool	normal	true	yes
h	sunny	mild	high	false	no
i	sunny	cool	normal	false	yes
j	rainy	mild	normal	false	yes
k	sunny	mild	normal	true	yes
l	overcast	mild	high	true	yes
m	overcast	hot	normal	false	yes
n	rainy	mild	high	true	no

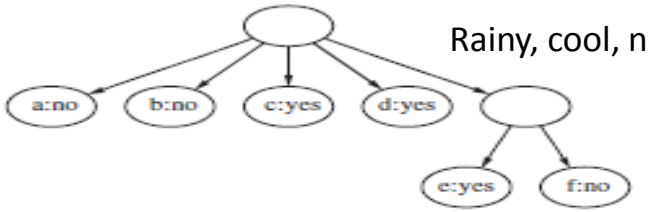
Clustering Incremental

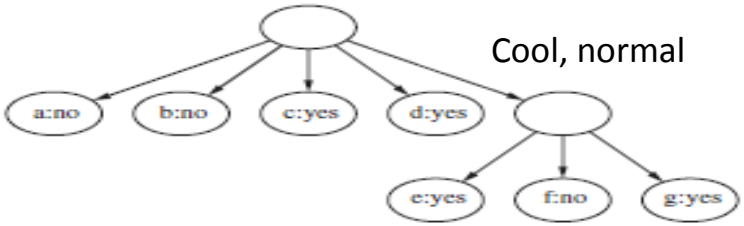
Table 4.6 The weather data with identification codes.

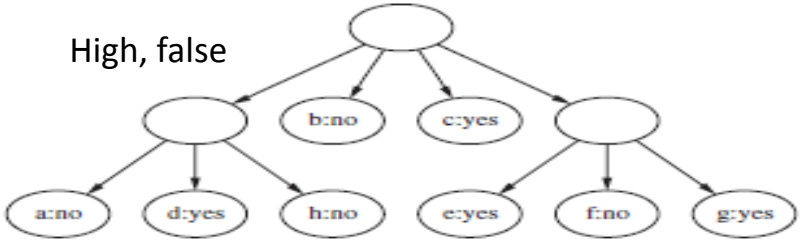
ID code	Outlook	Temperature	Humidity	Windy	Play
a	sunny	hot	high	false	no
b	sunny	hot	high	true	no
c	overcast	hot	high	false	yes
d	rainy	mild	high	false	yes
e	rainy	cool	normal	false	yes
f	rainy	cool	normal	true	no
g	overcast	cool	normal	true	yes
h	sunny	mild	high	false	no
i	sunny	cool	normal	false	yes
j	rainy	mild	normal	false	yes
k	sunny	mild	normal	true	yes
l	overcast	mild	high	true	yes
m	overcast	hot	normal	false	yes
n	rainy	mild	high	true	no

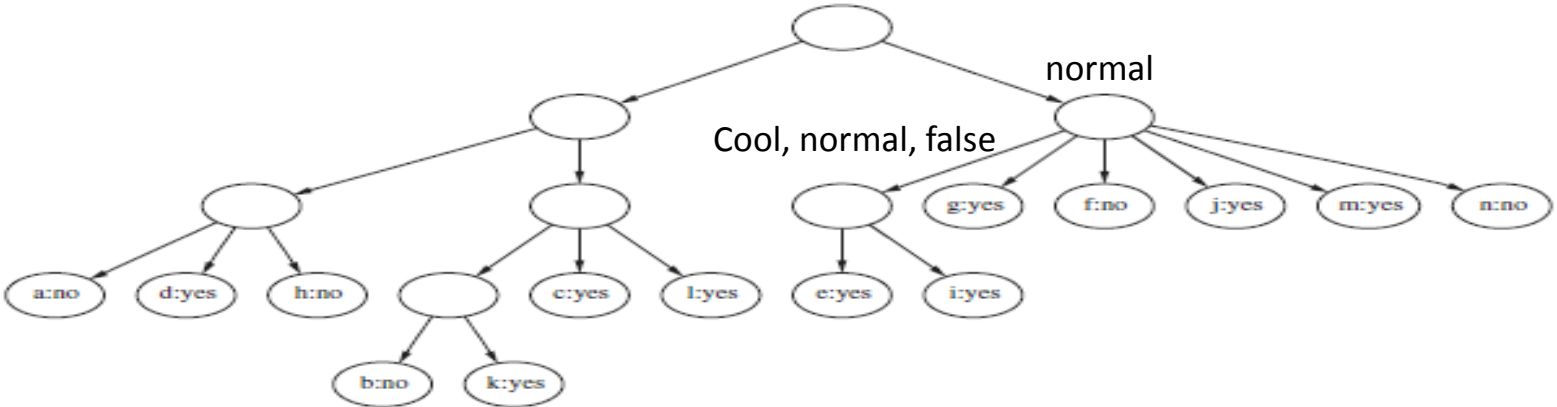








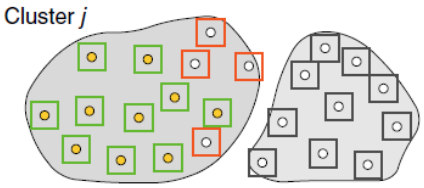




Métricas para evaluar un algoritmo de agrupamiento

Índice Externo: usado para medir el grado en que las etiquetas de un cluster coinciden con etiquetas de clases externas

- **F-mesure:** esta métrica está basada en la precisión y recall,



		Truth	
		P	N
Hypothesis	P	TP	FP
	N	FN	TN

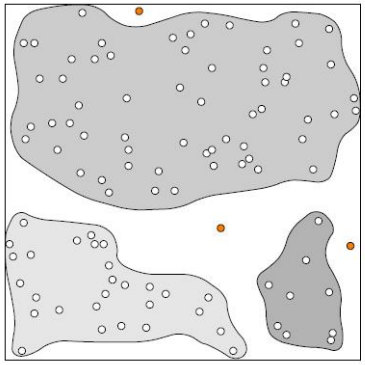
- **Entropía**: Es el grado de coincidencia de los clusters a las clases ya definidas de los datos originales.

$$e_i = -\sum_{j=1}^{C} p_{ij} log_2 p_{ij} \qquad (2.8)$$

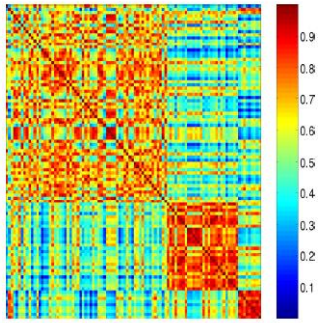
Métricas para evaluar un algoritmo de agrupamiento

Índice Interno: usado para medir la “bondad” de un estructura agrupada sin tener información de referencia externa

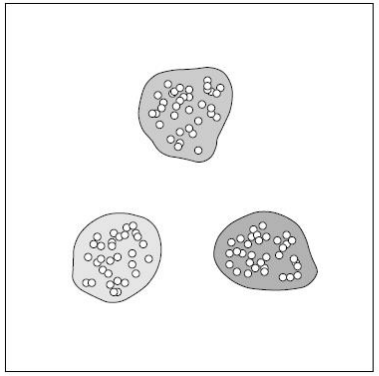
- Coeficiente de correlación:

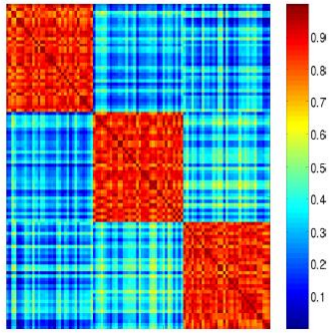


DBSCAN at random data.

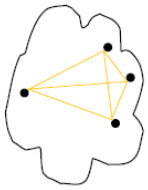


Similarity matrix sorted by cluster label.

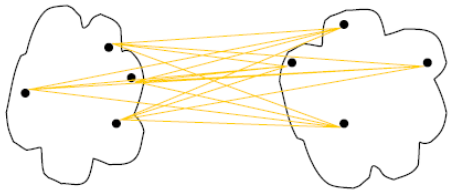




- Cohesión y separación

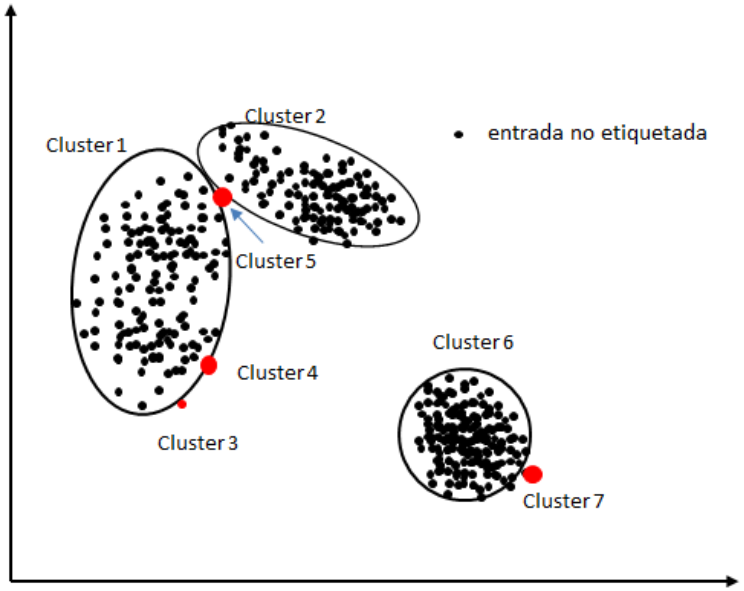


cohesión

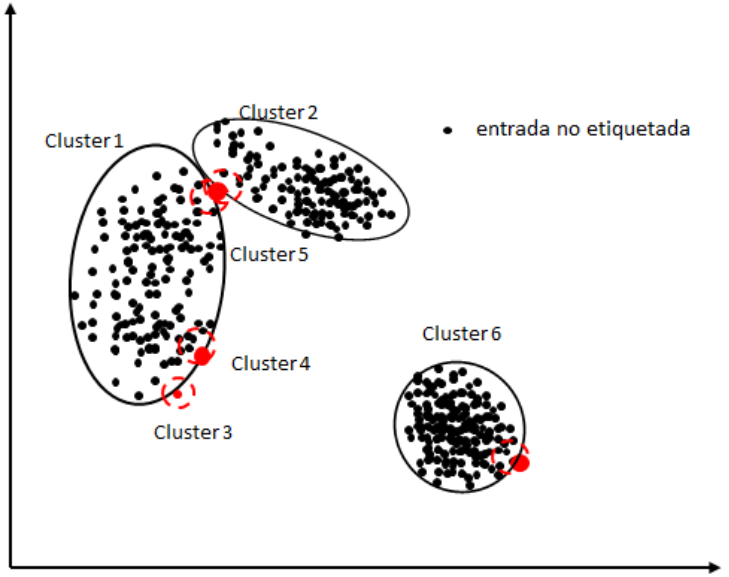


separación

Agrupamiento

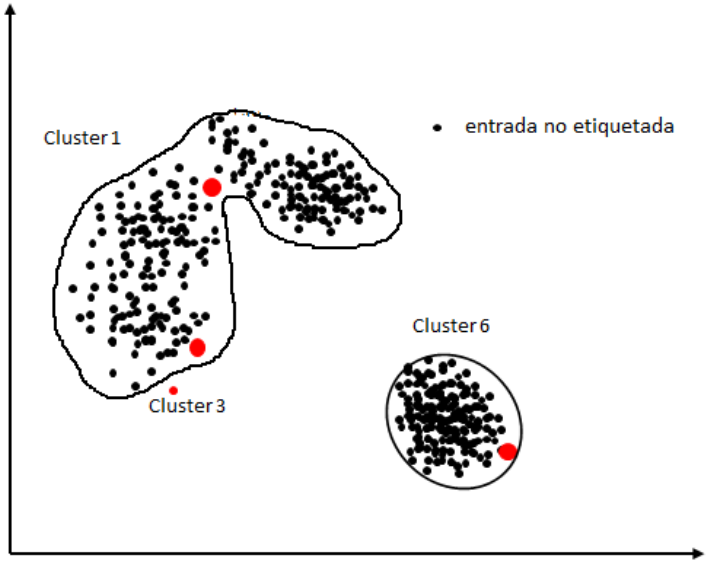


Inicial



Detección de los grupos candidatos para ser mezclados

Mezcla



Predicción

Predicción

- Predice un valor de una variable dada, sobre la base de los valores de otras variables, suponiendo un modelo lineal o no lineal de dependencia.

- **Ejemplos:**
  - Predecir las ventas de nuevos productos basados en gastos de publicidad.
  - Predecir la velocidad del viento como una función de la temperatura, humedad, presión de aire, etc.
  - Predecir series de tiempo de los índices bursátiles.

Regresión

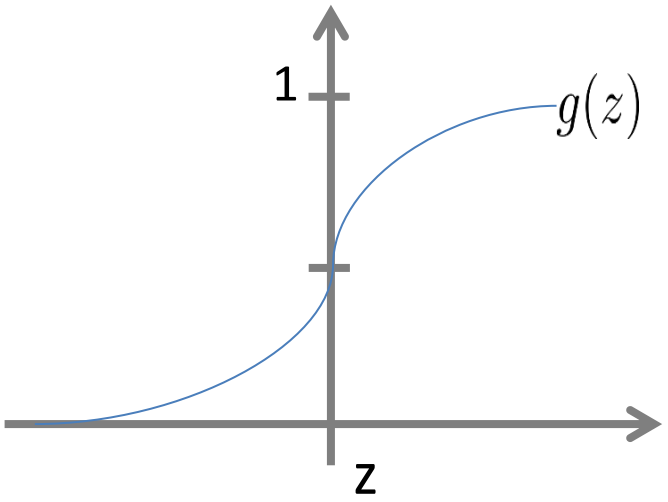
Regresión Lógica : $0 \leq h_\theta(x) \leq 1$

Cercano a clasificación

Regresión Lógica

$$h_\theta(x) = g(\theta^T x)$$

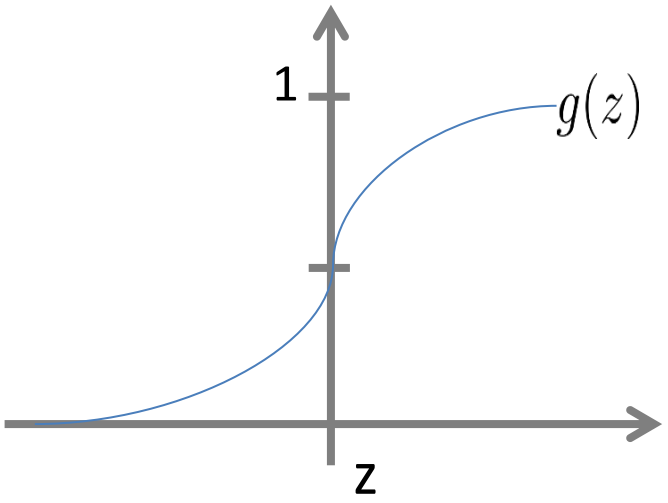
$$g(z) = \frac{1}{1+e^{-z}}$$



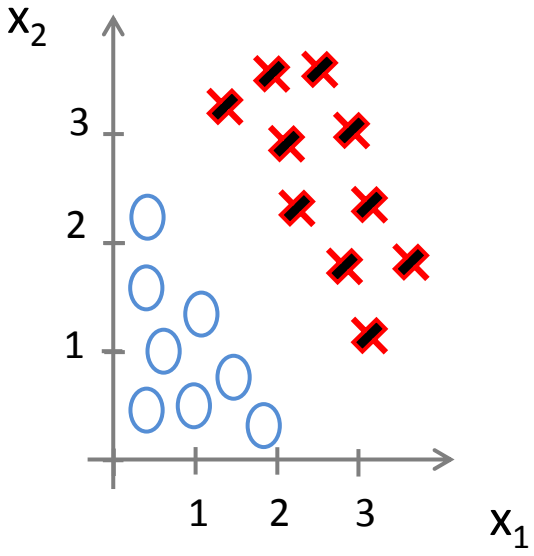
Regresión Lógica:
Barrera de decisión

predice “y=1“ si $h_\theta(x) \geq 0.5$

predice “y=0“ si $h_\theta(x) < 0.5$

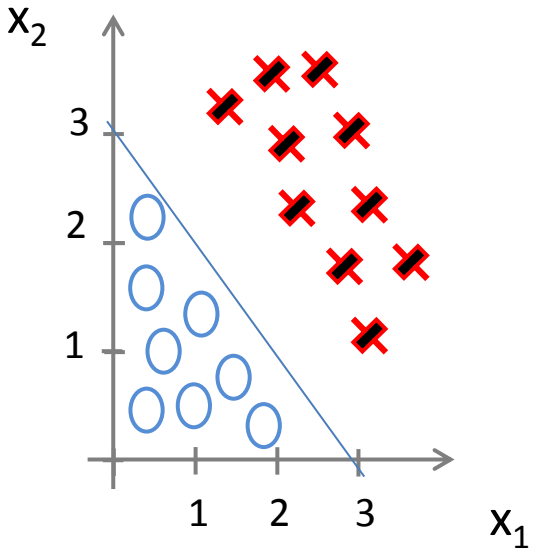


Regresión Lógica:
Barrera de decisión



$$h_\theta(x) = g(\theta_0 + \theta_1 x_1 + \theta_2 x_2)$$

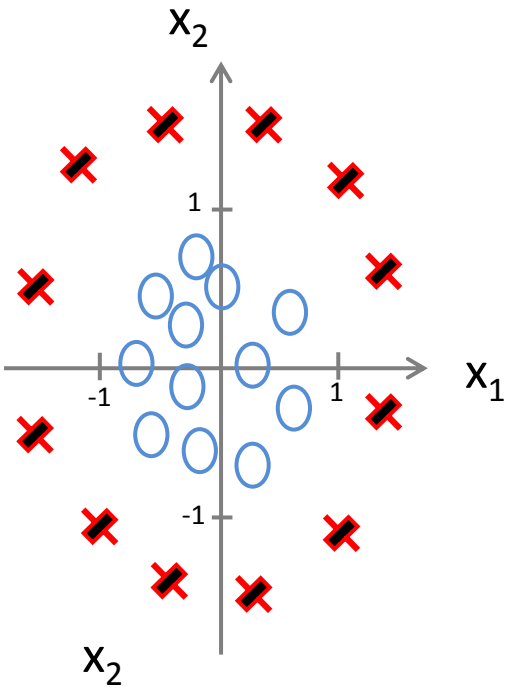
Regresión Lógica: Barrera de decisión



$$h_\theta(x) = g(\theta_0 + \theta_1 x_1 + \theta_2 x_2)$$

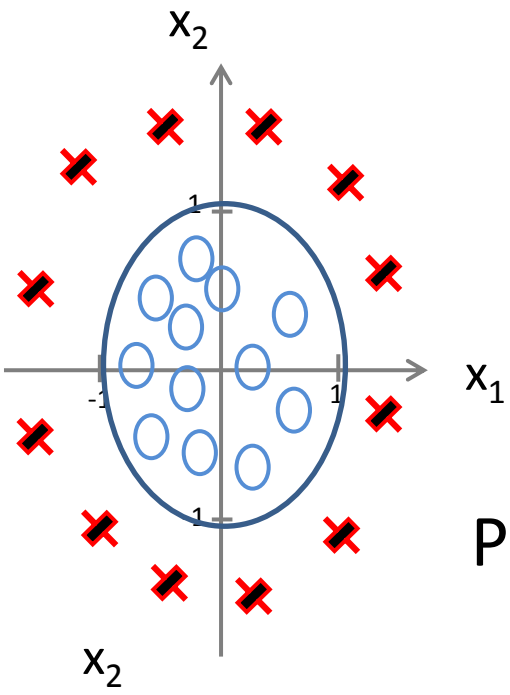
Predice “y=1“ si $-3 + x_1 + x_2 \geq 0$

Regresión Lógica:
Barrera de decisión



$$h_\theta(x) = g(\theta_0 + \theta_1 x_1 + \theta_2 x_2 + \theta_3 x_1^2 + \theta_4 x_2^2)$$

Regresión Lógica:
Barrera de decisión



$$h_\theta(x) = g(\theta_0 + \theta_1 x_1 + \theta_2 x_2 + \theta_3 x_1^2 + \theta_4 x_2^2)$$

Predice " y=1" si $-1 + x_1^2 + x_2^2 \geq 0$

Regresión Lógica: Función de costos

Conjunto de entrenamiento: $\{(x^{(1)}, y^{(1)}), (x^{(2)}, y^{(2)}), \cdots, (x^{(m)}, y^{(m)})\}$

n ejemplos $x \in \begin{bmatrix} x_0 \\ x_1 \\ \cdots \\ x_n \end{bmatrix}$ $y \in \{0, 1\}$

$$h_\theta(x) = \frac{1}{1 + e^{-\theta^T x}}$$

Como escoger el parámetro θ ?

Regresión Lógica: Función de costos

$$J(\theta) = \frac{1}{m} \sum_{i=1}^{m} \text{Cost}(h_\theta(x^{(i)}), y^{(i)})$$

$$\text{Cost}(h_\theta(x), y) = \begin{cases} -\log(h_\theta(x)) & \text{if } y = 1 \\ -\log(1 - h_\theta(x)) & \text{if } y = 0 \end{cases}$$

$$\min_\theta J(\theta)$$

Gradiente descendiente

$$J(\theta) = -\frac{1}{m}[\sum_{i=1}^{m} y^{(i)} \log h_\theta(x^{(i)}) + (1 - y^{(i)}) \log (1 - h_\theta(x^{(i)}))]$$

Queremos $\min_\theta J(\theta)$:

Repeat $\{$

$$\theta_j := \theta_j - \alpha \frac{\partial}{\partial \theta_j} J(\theta)$$ (simultaneamente actualizar θ_j)

$\}$

Función de costos

Regresión lineal: $J(\theta) = \frac{1}{m} \sum_{i=1}^{m} \frac{1}{2} \left(h_\theta(x^{(i)}) - y^{(i)} \right)^2$

$$\text{Cost}(h_\theta(x^{(i)}), y^{(i)}) = \frac{1}{2} \left(h_\theta(x^{(i)}) - y^{(i)} \right)^2$$

Problema de Optimización!!!

Dado θ , queremos calcular

- $J(\theta)$
- $\frac{\partial}{\partial \theta_j} J(\theta)$

(for j=0, …, n) $\{$

$$\theta_j := \theta_j - \alpha \frac{\partial}{\partial \theta_j} J(\theta)$$

$\}$

Sobre-ajustamiento (Overfitting)

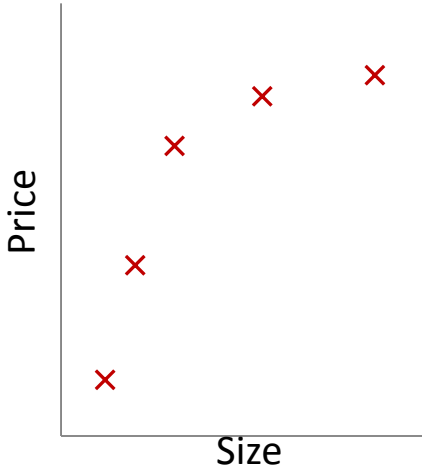
- Es el efecto de sobre-entrenar un algoritmo de aprendizaje con unos ciertos datos para los que se conoce el resultado deseado.

- El algoritmo de aprendizaje debe alcanzar un estado en el que sea capaz de predecir el resultado en otros casos a partir de lo aprendido con los datos de entrenamiento, **generalizando** para poder resolver situaciones distintas a las definidas en el entrenamiento.

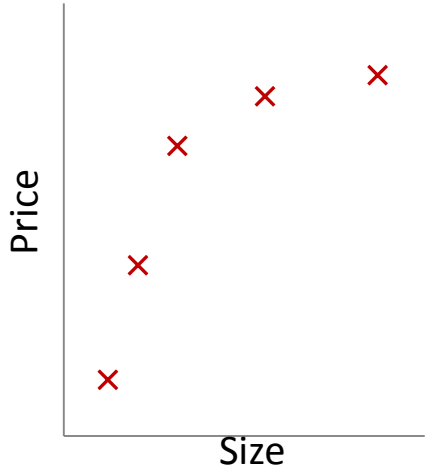
- Sin embargo, cuando un sistema se entrena demasiado (se sobre-entrena) o se entrena con datos extraños, el algoritmo de aprendizaje puede quedar ajustado a unas características muy específicas de los datos de entrenamiento.

Durante la fase de sobre-ajuste el éxito al responder las muestras de entrenamiento sigue incrementándose mientras que su actuación con muestras nuevas va empeorando.

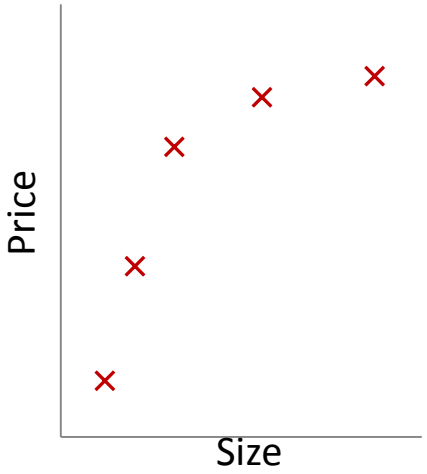
Sobre-ajustamiento (Overfitting)



$$\theta_0 + \theta_1 x$$

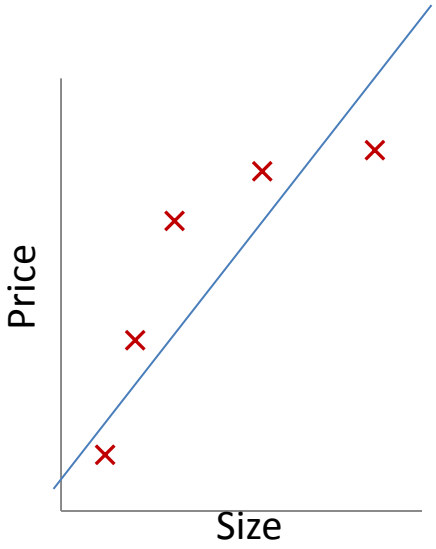


$$\theta_0 + \theta_1 x + \theta_2 x^2$$

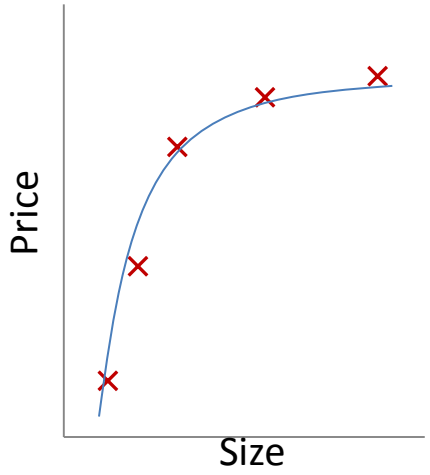


$$\theta_0 + \theta_1 x + \theta_2 x^2 + \theta_3 x^3 + \theta_4 x^4$$

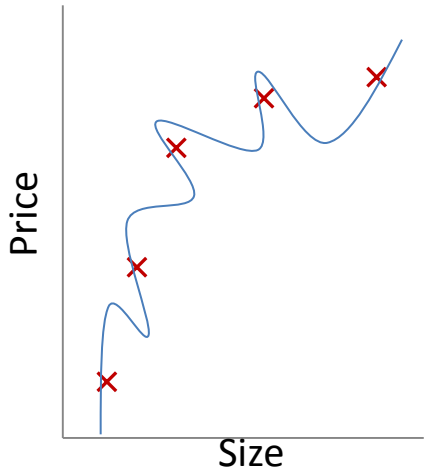
Sobre-ajustamiento (Overfitting)



$$\theta_0 + \theta_1 x$$

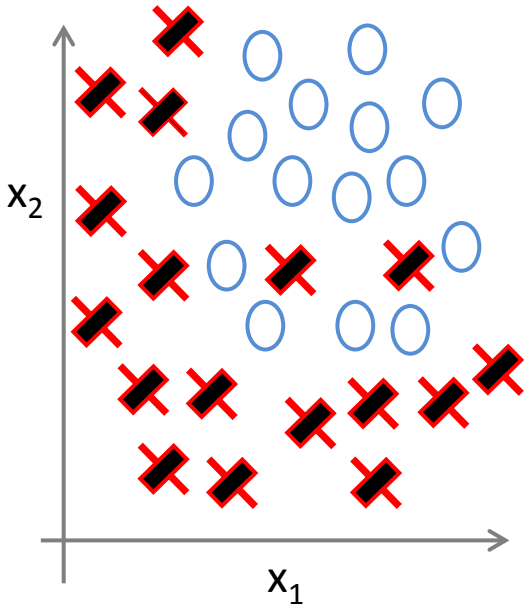


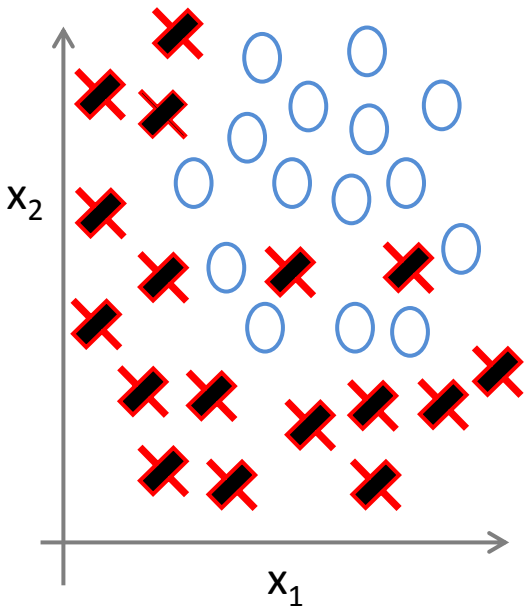
$$\theta_0 + \theta_1 x + \theta_2 x^2$$

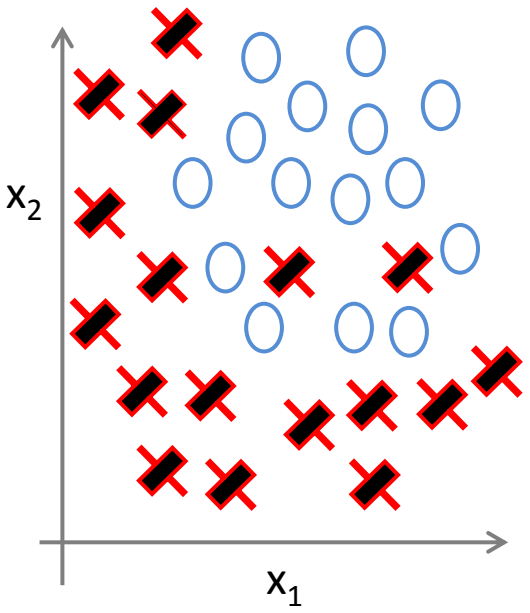


$$\theta_0 + \theta_1 x + \theta_2 x^2 + \theta_3 x^3 + \theta_4 x^4$$

Sobre-ajustamiento (Overfitting)







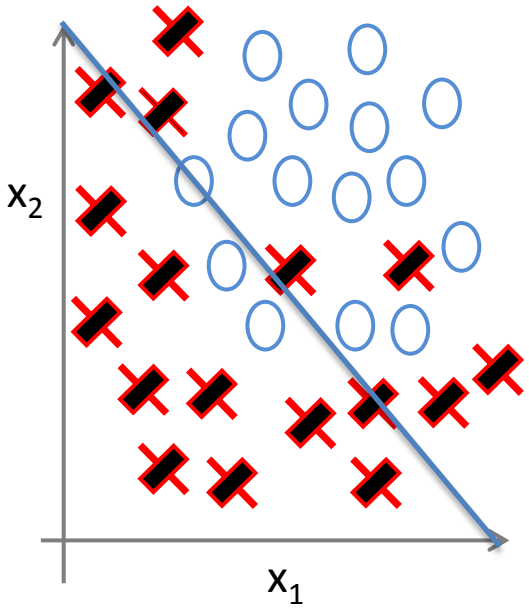
$$h_\theta(x) = g(\theta_0 + \theta_1 x_1 + \theta_2 x_2)$$

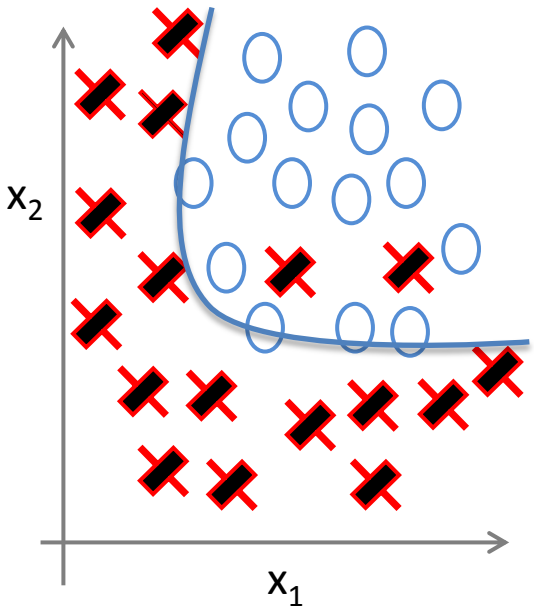
(g = función sigmoidal)

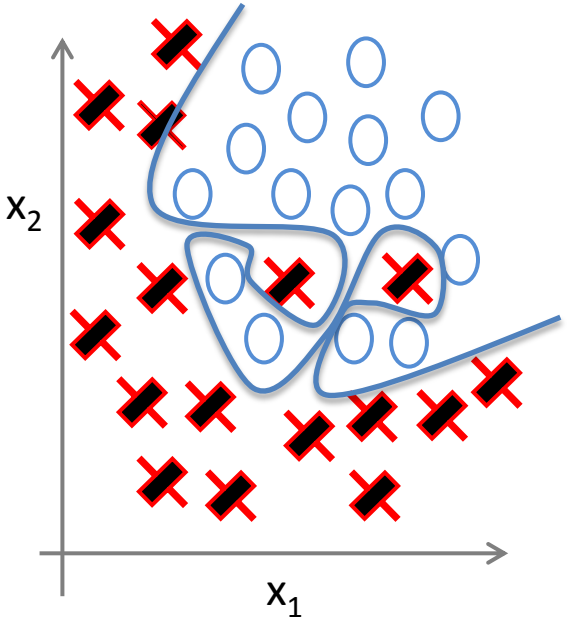
$$g(\theta_0 + \theta_1 x_1 + \theta_2 x_2 + \theta_3 x_1^2 + \theta_4 x_2^2 + \theta_5 x_1 x_2)$$

$$g(\theta_0 + \theta_1 x_1 + \theta_2 x_1^2 + \theta_3 x_1^2 x_2 + \theta_4 x_1^2 x_2^2 + \theta_5 x_1^2 x_2^3 + \theta_6 x_1^3 x_2 + \ldots)$$

sigmoidal





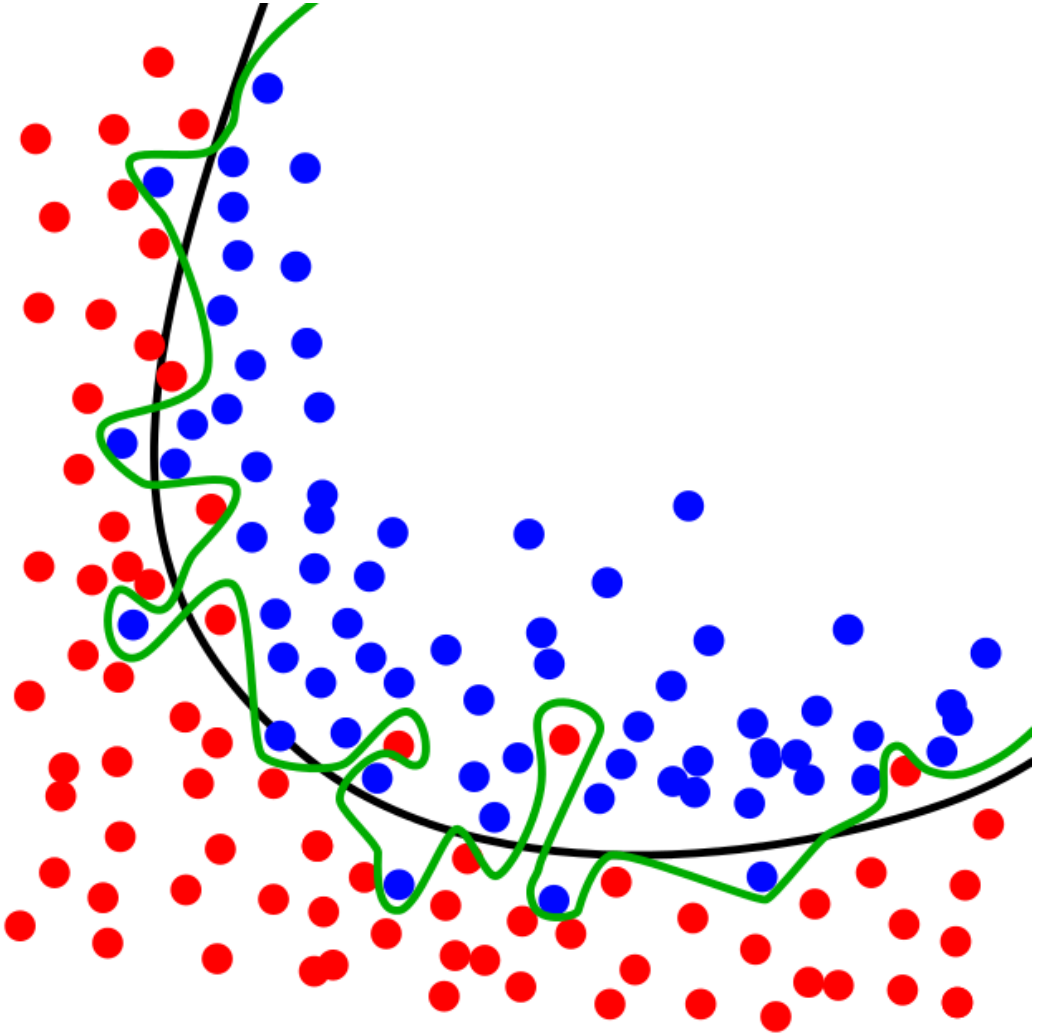


$$h_\theta(x) = g(\theta_0 + \theta_1 x_1 + \theta_2 x_2)$$

$$g(\theta_0 + \theta_1 x_1 + \theta_2 x_2 + \theta_3 x_1^2 + \theta_4 x_2^2 + \theta_5 x_1 x_2)$$

$$g(\theta_0 + \theta_1 x_1 + \theta_2 x_1^2 + \theta_3 x_1^2 x_2 + \theta_4 x_1^2 x_2^2 + \theta_5 x_1^2 x_2^3 + \theta_6 x_1^3 x_2 + \ldots)$$

Sobre-ajustamiento (Overfitting)



Emplear la **línea verde** como clasificador se adapta mejor a los datos con los que hemos entrenado al clasificador, pero está *demasiado* adaptada a ellos, de forma que ante nuevos datos probablemente arrojará más errores que la clasificación usando la **línea negra**.

Sobre-ajustamiento (Overfitting)

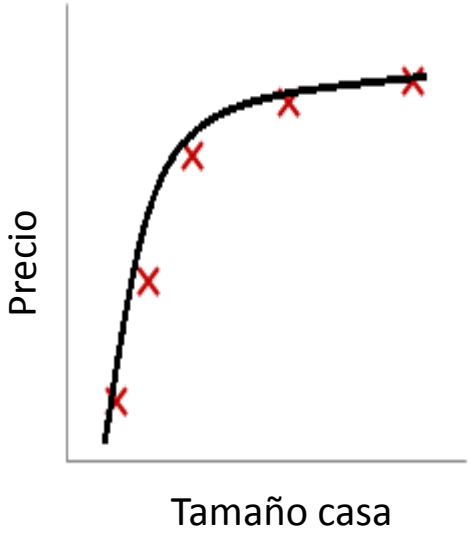
Opciones:

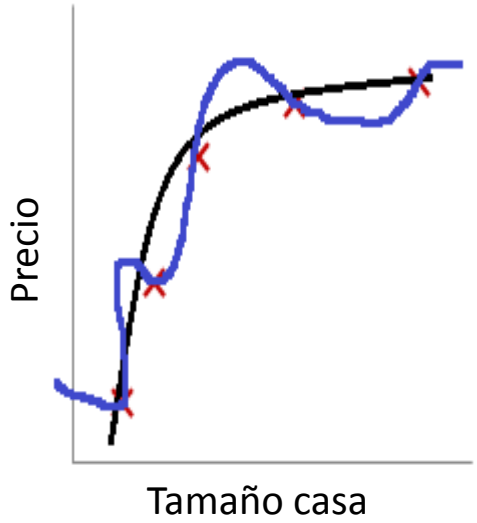
- **Reducir el número de características**.
  - Seleccionar manualmente las características que desea conservar.

- **Regularización.**
  - Mantener todas las características, pero reducir la magnitud/valores de los parámetros.
  - Funciona bien cuando tenemos una gran cantidad de características, y cada una contribuye un poco a la predicción.

.

Función de costo





$$\theta_0 + \theta_1 x + \theta_2 x^2$$

$$\theta_0 + \theta_1 x + \theta_2 x^2 + \theta_3 x^3 + \theta_4 x^4$$

Penalizar $\theta_3\ \theta_4$

$$\min_\theta \frac{1}{2m} \sum_{i=1}^{m} (h_\theta(x^{(i)}) - y^{(i)})^2$$

Patrones Secuenciales

- Descubrir patrones en los cuales la presencia de un conjunto de ítems es seguido por otro ítem en orden temporal.

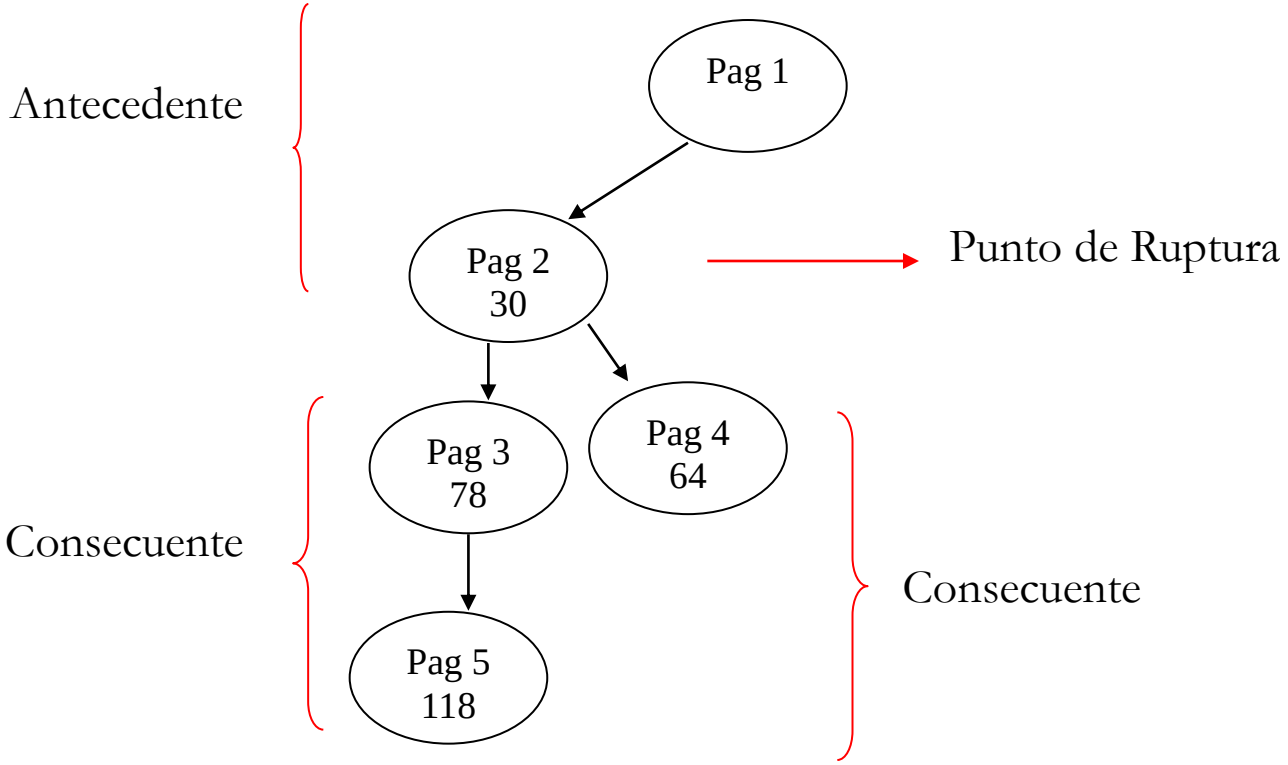
- Ejemplo: Encontrar y predecir el comportamiento de los visitantes de un sitio Web con respecto al tiempo.

$$[x1 \rightarrow x2 \rightarrow x3] \rightarrow [y1 \rightarrow y2] \text{ en t días}$$

[/public/team.jsp ->]---->/public/findUsers.jsp->
/private/mycourses/website/folders/assignment/assignment_view.jsp->
/public/portalDocument.js
en 2 dias

Patrones Secuenciales

Generación FBP-Árbol (Matriz FTM, Lista de Caminos)



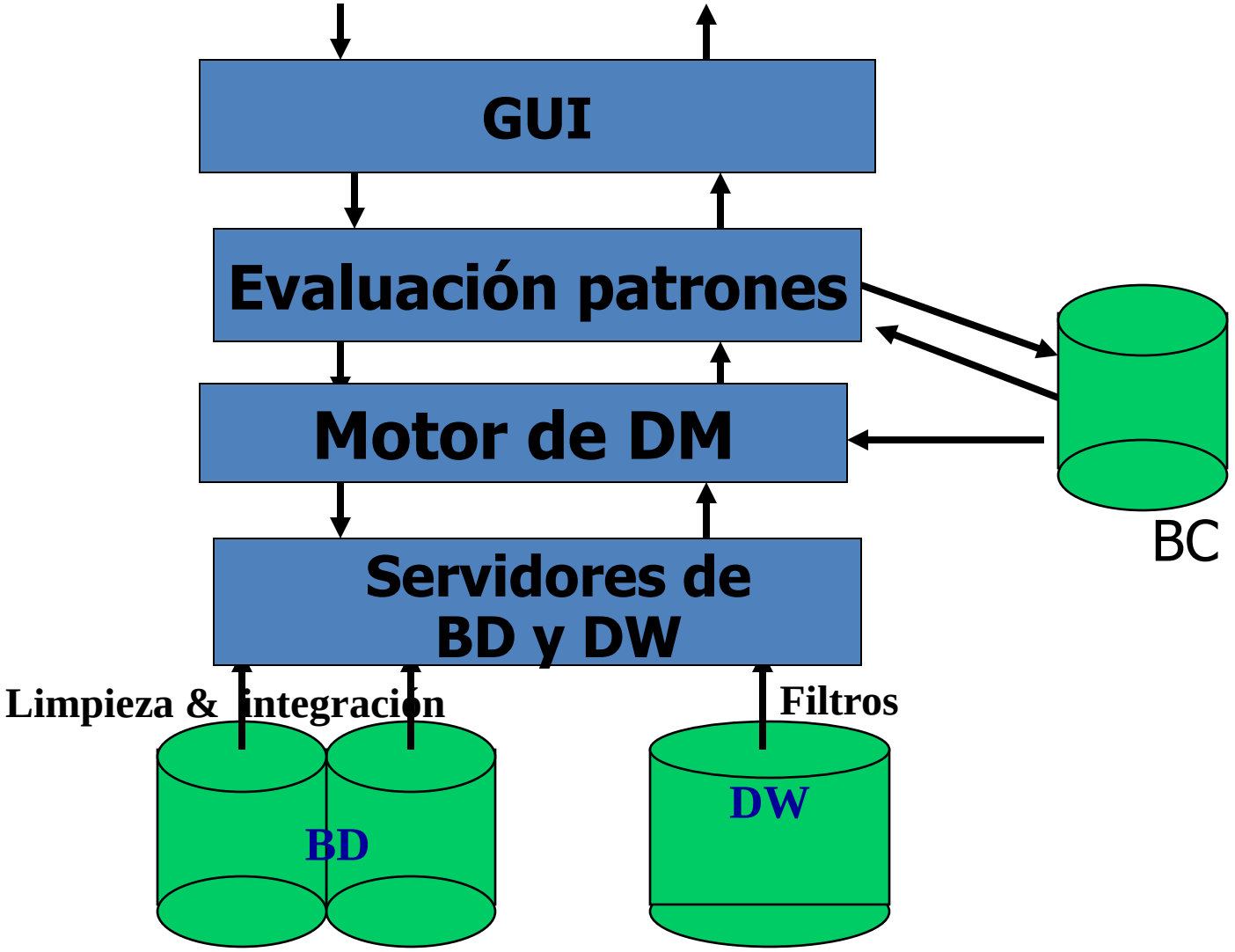
Patrones Secuenciales

Algoritmo Patrones (FBP-Arbol, soporte, confianza)

- La confianza de una *regla de comportamiento-frecuente* se representa como *conf*(PIND → PDEP) y define la probabilidad de recorrer el camino PDEP una vez se ha recorrido el camino PIND.

- Se recorre el árbol desde las hojas al nodo raíz.
- Teniendo en cuenta el soporte de cada camino las reglas son calculados como sigue.
- Buscar en hojas el punto de ruptura.
  - Si la hoja no es Punto ruptura, ir a hoja anterior.
  - Si la hoja es Punto Ruptura, calcular confianza.
    - Si conf > confianza, genera Patrón
    - Si conf < confianza, podar rama de árbol.

Minería de Datos e Inteligencia de Negocio



Minería de Datos e Inteligencia de Negocio

- Financieras
- Comercio
- Seguros
- Educación
- Medicina
- Bioinformática
- Otras áreas

Minería de Datos e Inteligencia de Negocio

Agente comercial: ¿Debo conceder una hipoteca a un cliente?

Datos

cId	Credit-p (years)	Credit-a (euros)	Salary (euros)	Own House	Defaulter accounts	...	Returns-credit
101	15	60.000	2.200	yes	2	...	no
102	2	30.000	3.500	yes	0	...	yes
103	9	9.000	1.700	yes	1	...	no
104	15	18.000	1.900	no	0	...	yes
105	10	24.000	2.100	no	0	...	no
...	...	...	...	...	...	...	...

Minería de datos

Clasificación

Modelo generado

If Defaulter-accounts > 0 **then** Returns-credit = no

If Defaulter-accounts = 0 **and** [(Salary > 2500) **or** (Credit-p > 10)] **then** Returns-credit = yes

Minería de Datos e Inteligencia de Negocio

Supermercado: ¿Cuándo los clientes compran huevos, también compran aceite?

Datos:

BasketId	Eggs	Oil	Nappies	Wine	Milk	Butter	Salmon	Endive	...
1	yes	yes	no	yes	no	yes	yes	yes	...
2	no	yes	no	no	yes	no	no	yes	...
3	no	no	yes	no	yes	no	no	no	...
4	no	yes	yes	no	yes	no	no	no	...
5	yes	yes	no	no	no	yes	no	yes	...
6	yes	no	no	yes	yes	yes	yes	no	...
7	no	no	no	no	no	no	no	no	...
8	yes	yes	yes	yes	yes	yes	yes	no	...
...	...	...	...	...	...	...	...	...	...

Modelo generado:

Asociación

Eggs -> Oil: Confianza = 75%, Soporte = 37%

Minería de Datos e Inteligencia de Negocio

Gestión de personal de una empresa: ¿Qué tipos de empleados hay contratados?

Datos:

Id	Salary	Married	Car	Children	Rent/ Owner	Union	Off sick/year	Work years	Gender
1	10000	yes	no	0	Rent	no	7	15	M
2	20000	no	yes	1	Rent	yes	3	3	F
3	15000	yes	yes	2	Owner	yes	5	10	M
4	30000	yes	yes	1	Rent	no	15	7	F
5	10000	yes	yes	0	Owner	yes	1	6	M
6	40000	no	yes	0	Rent	yes	3	16	F
7	25000	no	no	0	Rent	yes	0	8	M
8	20000	no	yes	0	Owner	yes	2	6	F
15	8000	no	yes	0	Rent	no	3	2	M
...	...	...	...	...	...	...	...	...	...

Modelo generado:

Minería de datos

Clustering

Grupo 1: Sin niños y en una casa alquilada. Bajo número de uniones. Muchos días enfermos

Grupo 2: Sin niños y con coche. Alto número de uniones. Pocos días enfermos. Más mujeres y en una casa alquilada

Grupo 3: Con niños, casados y con coche. Más hombres y normalmente propietarios de casa. Bajo número de uniones

Minería de Datos e Inteligencia de Negocio

Tienda de TV: ¿Cuántas televisiones planas se venderán el próximo mes?

Datos:

PRODUCT	Month–12	...	Month–4	Month–3	Month–2	Month–1	Month
Flat TV 30'	20	...	52	14	139	74	?
Video-dvd-recorder	11	...	43	32	26	59	?
Discman	50	...	61	14	5	28	?
Five star fridge	3	...	21	27	1	49	?
Three star fridge	14	...	27	2	25	12	?
...	...	...	...	...	...	...	...

Modelo generado:

Minería de datos

Predicción

Modelo lineal: número de televisiones para el próximo mes

V(month)*flatTV* = 0.62 V(Month-1)*flat-TV* + 0.33 V(Month-2)*flat-TV* + 0.12 V(Month-1)*DVD-Recorder* – 0.05

Weka

Weka
(Data Mining Tool)

- Weka es una herramienta de minería de datos de código abierto desarrollado en Java.

- Se utiliza para la investigación, la educación, y las aplicaciones.

- Se puede ejecutar en Windows, Linux y Mac.

http://www.cs.waikato.ac.nz/ml/weka/

Weka

- Weka es una colección de algoritmos de aprendizaje automático para tareas de minería de datos.
- Los algoritmos bien se pueden aplicar directamente a un conjunto de datos (con interfaz gráfica de usuario) o llamados desde su propio código Java (usando biblioteca Weka de Java).
- Weka contiene herramientas para **pre-procesamiento de los datos , clasificación, regresión, clustering, reglas de asociación, selección de caracteristicas y la visualización.**

Weka

- Entrada de datos a Weka (Input)
- Weka para Data Mining
- Salida desde Weka (Output)

Entrada de datos a Weka (Input)

- El maps popular formato is “arff” (“arff” es la extensión del nombre del archivo).

FILE FORMAT

@relation RELATION_NAME

@attribute ATTRIBUTE_NAME ATTRIBUTE_TYPR
@attribute ATTRIBUTE_NAME ATTRIBUTE_TYPR
@attribute ATTRIBUTE_NAME ATTRIBUTE_TYPR
@attribute ATTRIBUTE_NAME ATTRIBUTE_TYPR

@data
DATAROW1
DATAROW2
DATAROW3

Entrada de datos a Weka (Input)

archivo “arff”

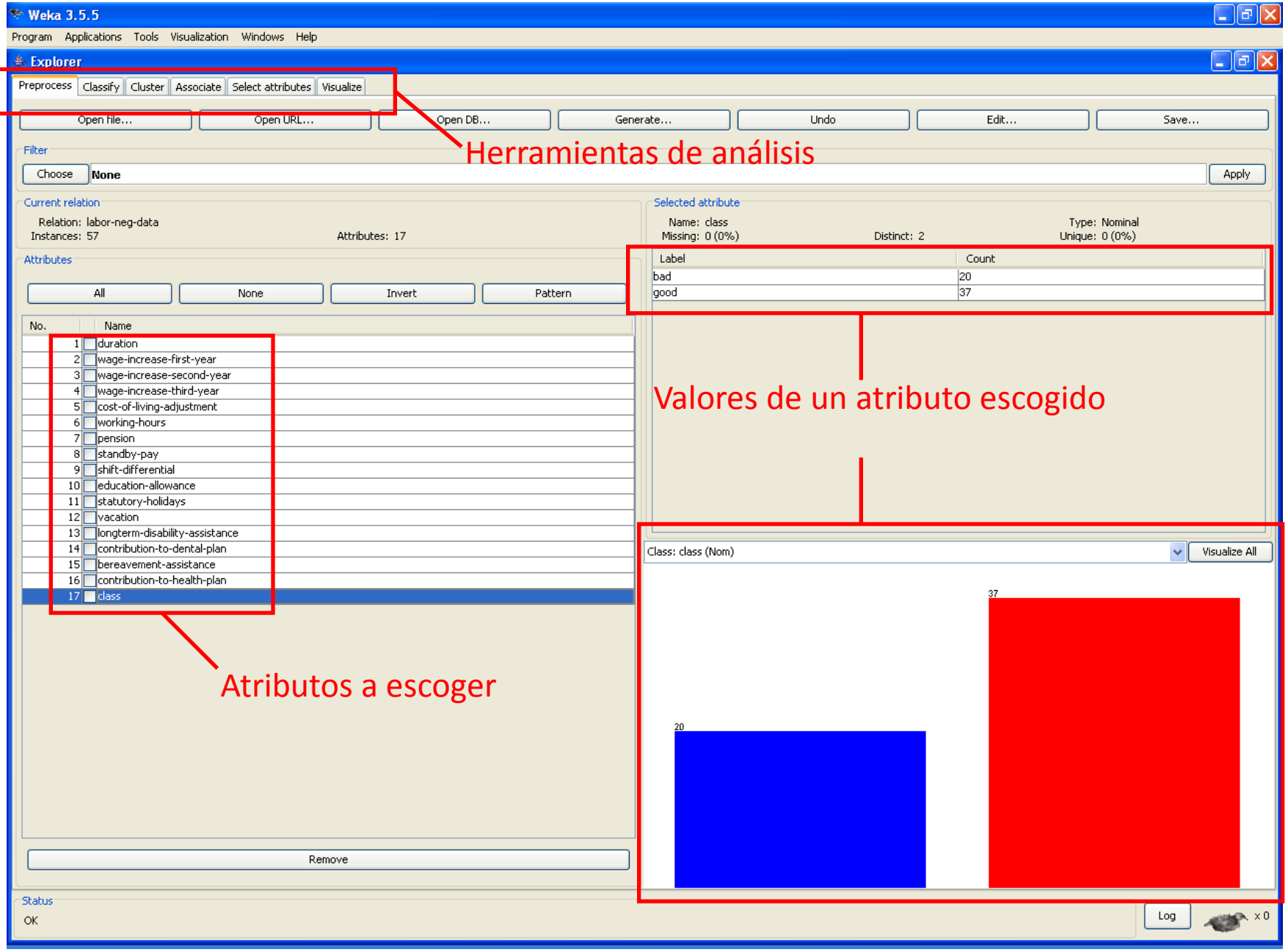

```
@relation heart-disease-simplified

@attribute age numeric
@attribute sex { female, male}
@attribute chest_pain_type { typ_angina, asympt, non_anginal, atyp_angina}
@attribute cholesterol numeric
@attribute exercise_induced_angina { no, yes}
@attribute class { present, not_present}

@data
63,male,typ_angina,233,no,not_present
67,male,asympt,286,yes,present
67,male,asympt,229,yes,present
38,female,non_anginal,?,no,not_present
...
```

Weka para Data Mining

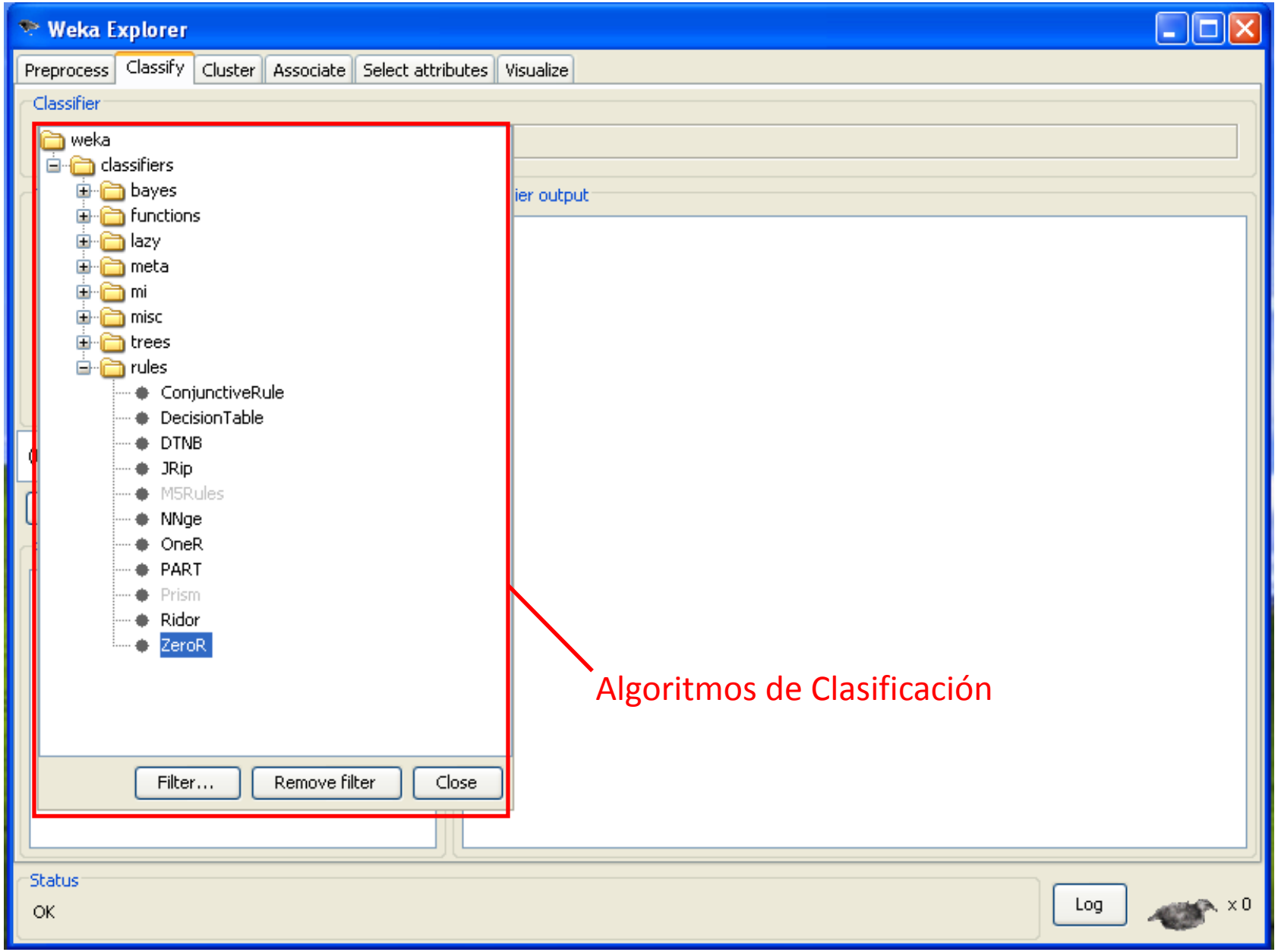
- Con interfaz gráfica de usuario

- O usando biblioteca Weka de Java

Weka GUI


Weka 3.5.5
Program Applications Tools Visualization Windows Help
Explorer
Preprocess Classify Cluster Associate Select attributes Visualize
Open file... Open URL... Open DB... Generate... Undo Edit... Save...
Herramientas de análisis
Filter
Choose None Apply
Current relation
Relation: labor-neg-data
Instances: 57
Attributes: 17
Selected attribute
Name: class
Missing: 0 (0%)
Distinct: 2
Type: Nominal
Unique: 0 (0%)
Label Count
bad 20
good 37
Valores de un atributo escogido
Attributes
All None Invert Pattern
No. Name
1 duration
2 wage-increase-first-year
3 wage-increase-second-year
4 wage-increase-third-year
5 cost-of-living-adjustment
6 working-hours
7 pension
8 standby-pay
9 shift-differential
10 education-allowance
11 statutory-holidays
12 vacation
13 longterm-disability-assistance
14 contribution-to-dental-plan
15 bereavement-assistance
16 contribution-to-health-plan
17 class
Atributos a escoger
Class: class (Nom)
Visualize All
37
20
Remove
Status
OK
Log
x 0

Weka GUI


Weka Explorer
Preprocess
Classify
Cluster
Associate
Select attributes
Visualize
Classifier
weka
classifiers
bayes
functions
lazy
meta
mi
misc
trees
rules
ConjunctiveRule
DecisionTable
DTNB
JRip
M5Rules
NNge
OneR
PART
Prism
Ridor
ZeroR
Filter...
Remove filter
Close
ier output
Status
OK
Log
x 0
Algoritmos de Clasificación

Weka Java library

- Clases para carga de datos
- Clases para los clasificadores
- Clases para la evaluación

Clases para Carga de Datos

- weka.core.Instances
- weka.core.Attribute

- Cada DataRow -> Instance, Every Attribute -> Attribute, Whole -> Instances

```
# Load a file as Instances
FileReader reader;
reader = new FileReader(path);
Instances instances = new Instances(reader);
```

Clases para Carga de Datos

- **Cómo recuperar un valor de una instancia?**

```
# Get Instance
Instance instance = instances.instance(index);
# Get Instance Count
int count = instances.numInstances();
```

- **Cómo recuperar un atributo?**

```
# Get Attribute Name
Attribute attribute = instances.attribute(index);
# Get Attribute Count
int count = instances.numAttributes();
```

Clases para Carga de Datos

- **Cómo recuperar el valor del atributo para cada Instancia?**

```
# Get value
instance.value(index);      or
instance.value(attrName);
```

- **Indice de Clase**

```
# Get Class Index
instances.classIndex();                    or
instances.classAttribute().index();
# Set Class Index
instances.setClass(attribute);      or
instances.setClassIndex(index);
```

Clases para los Clasificadores

- **Clases Weka para C4.5, Naïve Bayes, and SVM**
  - Clasificadores:
    - C4.5: weka.classifier.trees.J48
    - NaiveBayes: weka.classifiers.bayes.NaiveBayes
    - SVM: weka.classifiers.functions.SMO

- **Cómo construir un clasificador?**

```
# Build a C4.5 Classifier
Classifier c = new weka.classifier.trees.J48();
c.buildClassifier(trainingInstances);
# Build a SVM Classifier
Classifier e = weka.classifiers.functions.SVM();
e.buildClassifier(trainingInstances);
```

Clases para la Evaluación

- weka.classifiers.CostMatrix
- weka.classifiers.Evaluation

- **Cómo usarlas?**

```
# Use Classifier To Do Classification
CostMatrix costMatrix = null;
Evaluation eval = new Evaluation(testingInstances, costMatrix);

for (int i = 0; i < testingInstances.numInstances(); i++){
eval.evaluateModelOnceAndRecordPrediction(c,testingInstances.instance(i));
System.out.println(eval.toSummaryString(false));
System.out.println(eval.toClassDetailsString()) ;
System.out.println(eval.toMatrixString());
}
```

Clases para la evaluación

Validación Cruzada

- Divide un único conjunto de datos en N partes iguales.

- Toma la N-1 como un conjunto de datos de entrenamiento, el resto se utilizará como prueba de conjunto de datos.

Clases para la evaluación

- Obtención conjunto de entrenamiento

```
Random random = new Random(seed);
instances.randomize(random);
instances.stratify(N);

for (int i = 0; i < N; i++)
{
        Instances train = instances.trainCV(N, i , random);
        Instances test = instances.testCV(N, i , random);
}
```

Salida desde Weka (Output)

```
=== Summary ===

Correctly Classified Instances          46               65.7143 %
Incorrectly Classified Instances        24               34.2857 %
Kappa statistic                          0.2398
Mean absolute error                      0.3654
Root mean squared error                  0.5367
Relative absolute error                 75.2288 %
Root relative squared error            108.9601 %
Total Number of Instances               70

=== Detailed Accuracy By Class ===

               TP Rate   FP Rate   Precision   Recall  F-Measure   ROC Area  Class
                 0.878     0.655       0.655     0.878      0.75       0.632     Y
                 0.345     0.122       0.667     0.345      0.455      0.632     N
Weighted Avg.    0.657     0.434       0.66      0.657      0.628      0.632

=== Confusion Matrix ===

  a  b   <-- classified as
 36  5 |  a = Y
 19 10 |  b = N
```

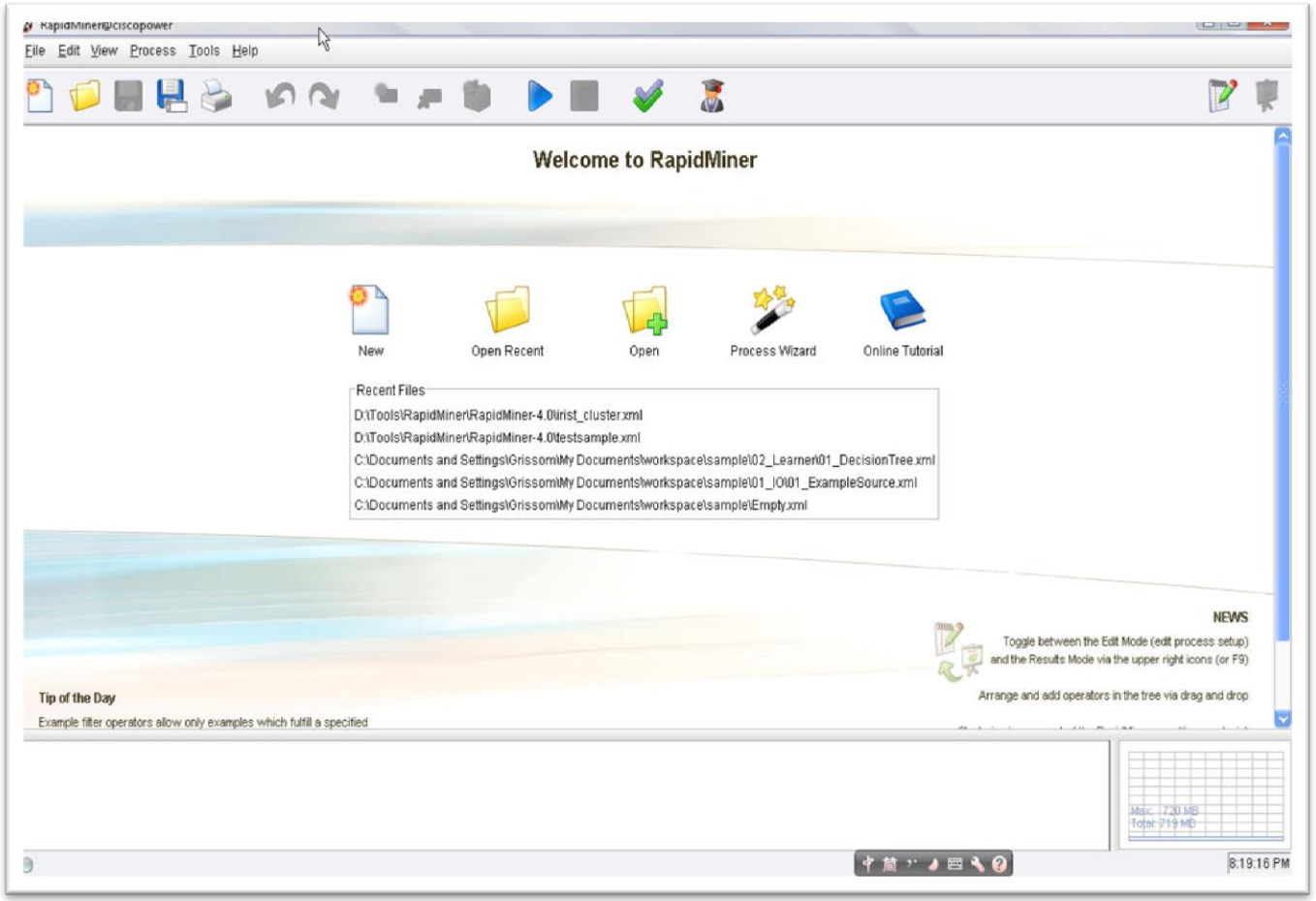
Tres pasos para el uso

- Asignar el archivo de datos
- Seleccionar funcionalidad
- Ejecutar Función utilizando RápidoMinero

Los conjuntos de datos pueden utilizar formato ARFF, pero otros formatos también

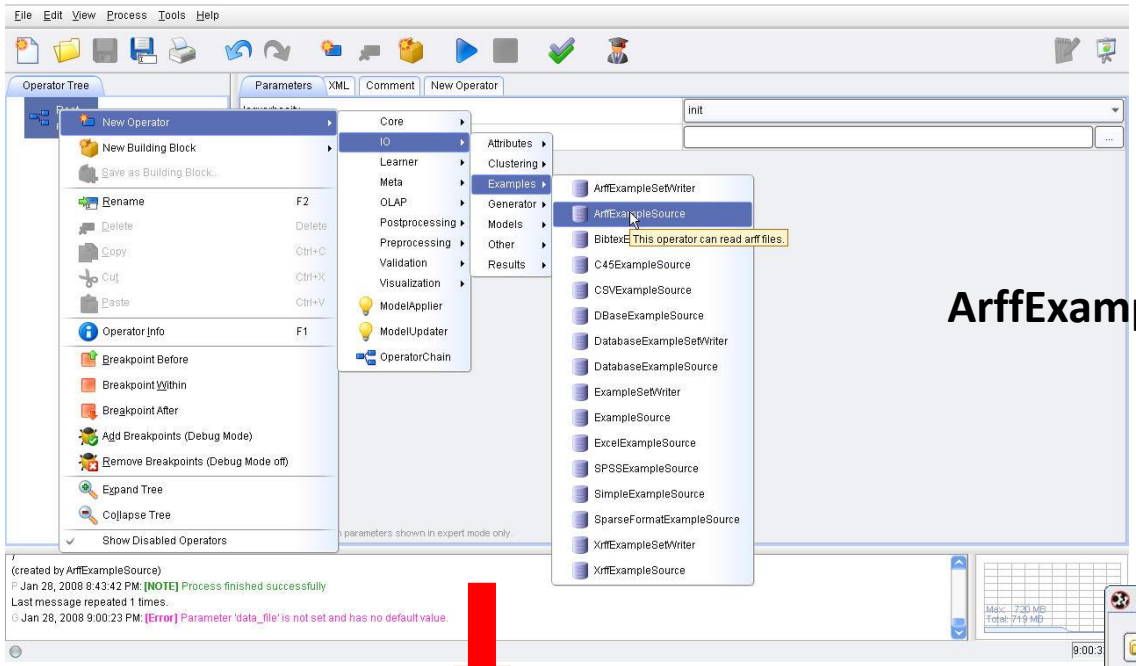
http://rapid-i.com/content/view/130/82/

- Crear un proyecto Rapid Miner (opción “new”)

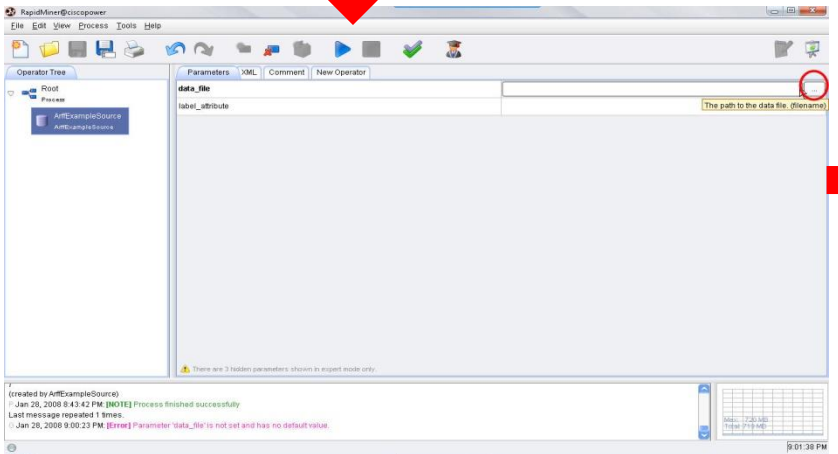


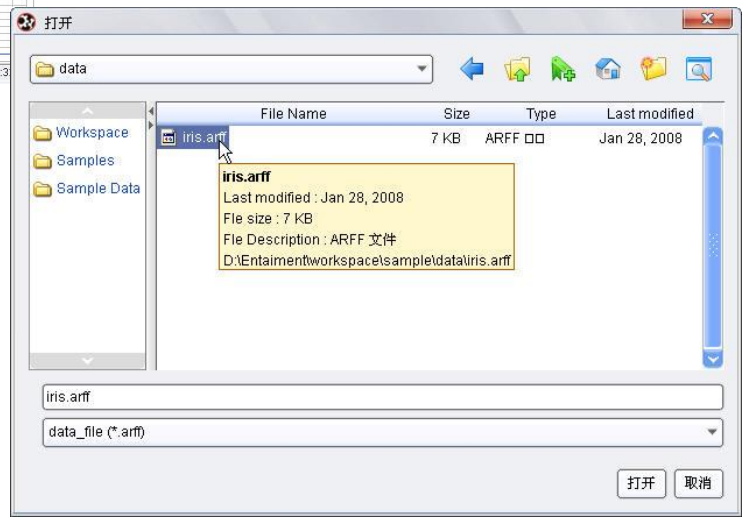
Establecer una fuente de datos para el proyecto

- Arff, excel , etc



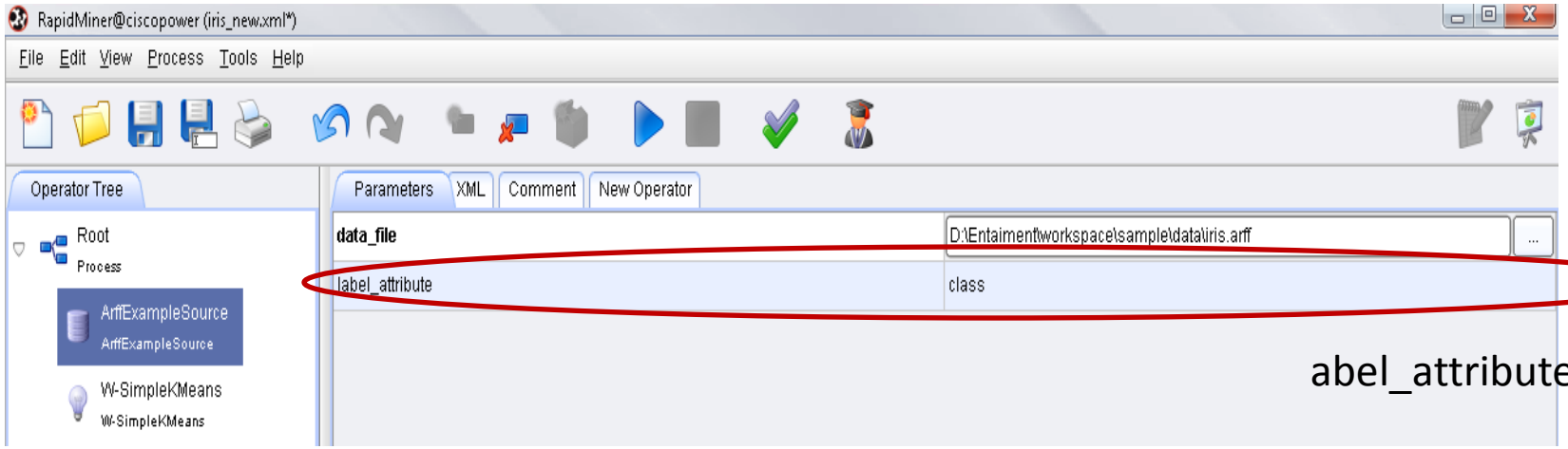
ArffExampleSource" menu





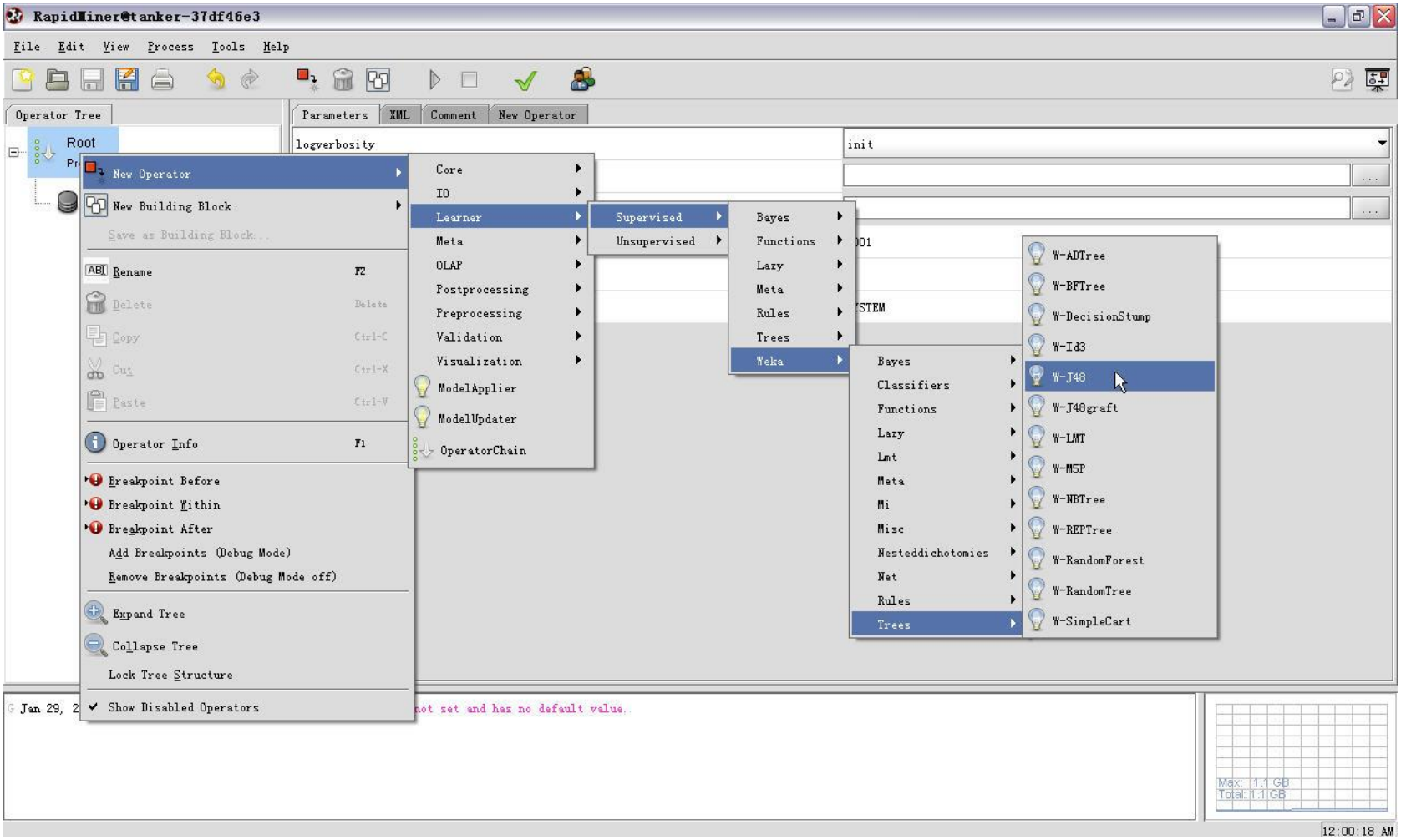
Escoger etiqueta atributo

- qué atributo en el origen de datos es el atributo de clase.

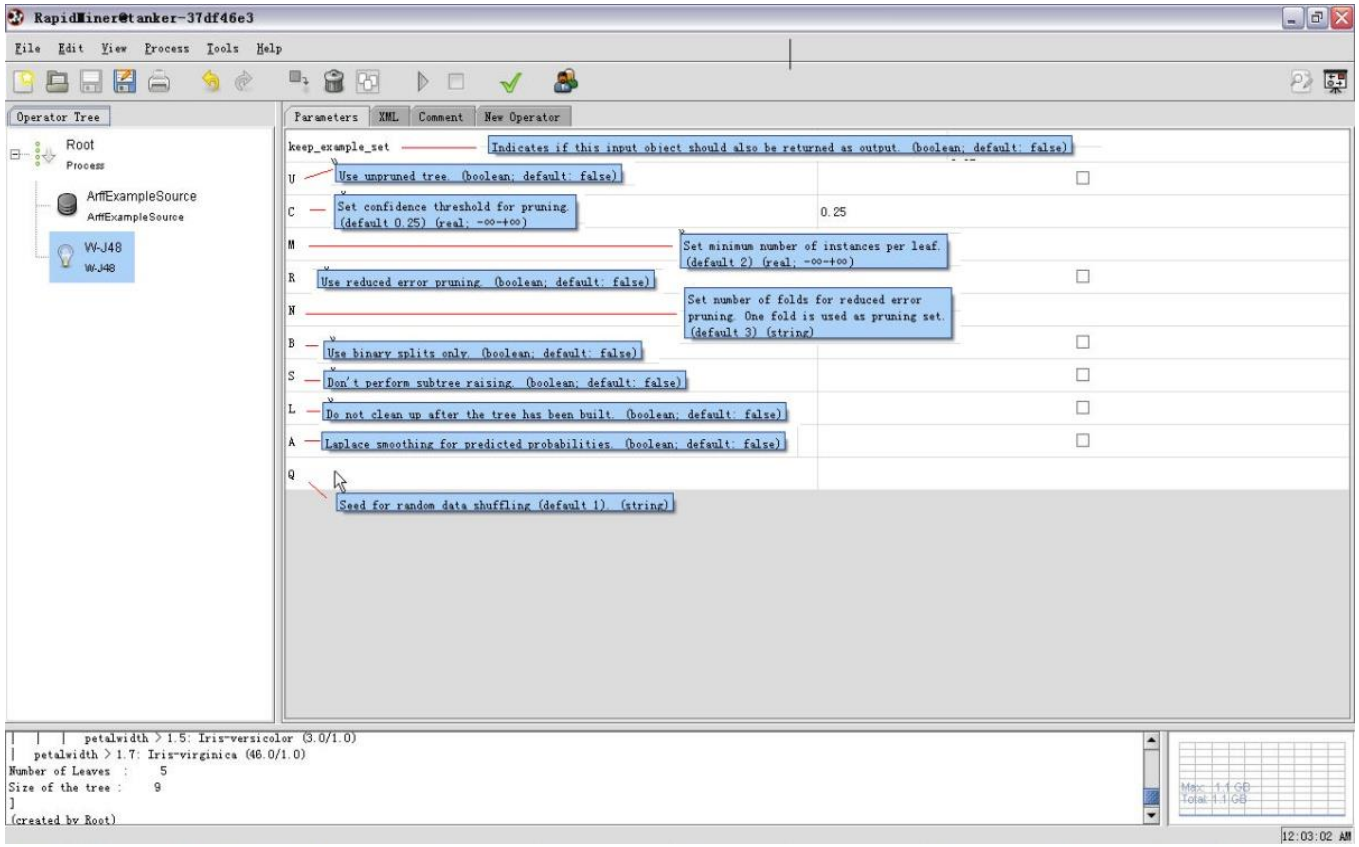


Seleccionar funcionalidad

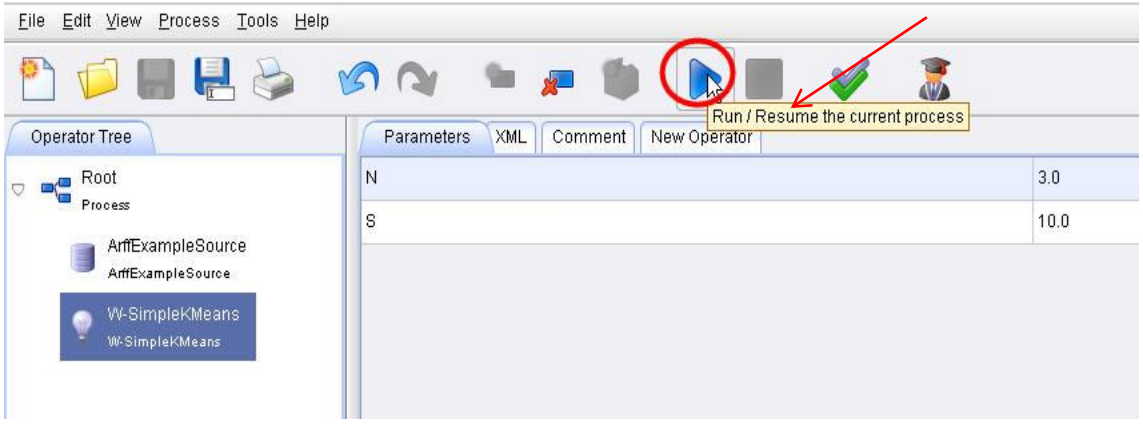
clustering, classification, decision tree, etc.



Escoger parámetros



Ejecutar



resultados