Minería de datos: Modelos de Predicción, Modelos de Descripción, y usos en IN

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

Modelos de Predicción: clasificación, regresión, series temporales, etc.

Clasificación

- Dada una colección de registros (conjunto de entrenamiento)
  - Cada registro contiene un conjunto de atributos, uno de los atributos es la clase.
- Encontrar un modelo para cada clase de atributo en función de los valores de otros atributos.

- Meta: registros deben ser asignados a una clase con la mayor precisión posible.
- Un conjunto de prueba se utiliza para determinar la precisión del modelo.

- Por lo general, el conjunto de datos dado se divide en conjunto de entrenamiento y de prueba (utilizado para construir el modelo y para validarlo, respectivamente).

Clasificación

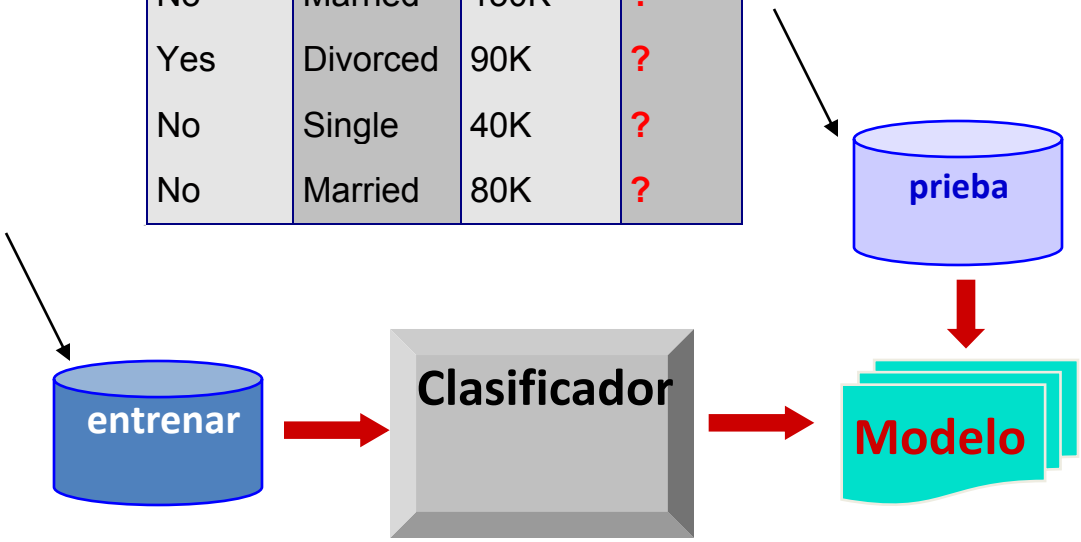
Email: Spam / No es Spam?
Transacciones en línea: Fraudulento (Si / No)?
Tumor: Maligno / Benigno ?

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

Clasificación

	categoría	categoría	continuo	clase
Tid	**Refund**	**Marital Status**	**Taxable Income**	**Cheat**
1	Yes	Single	125K	**No**
2	No	Married	100K	**No**
3	No	Single	70K	**No**
4	Yes	Married	120K	**No**
5	No	Divorced	95K	**Yes**
6	No	Married	60K	**No**
7	Yes	Divorced	220K	**No**
8	No	Single	85K	**Yes**
9	No	Married	75K	**No**
10	No	Single	90K	**Yes**

Refund	Marital Status	Taxable Income	Cheat
No	Single	75K	?
Yes	Married	50K	?
No	Married	150K	?
Yes	Divorced	90K	?
No	Single	40K	?
No	Married	80K	?



Clasificación

Detección de Fraude

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

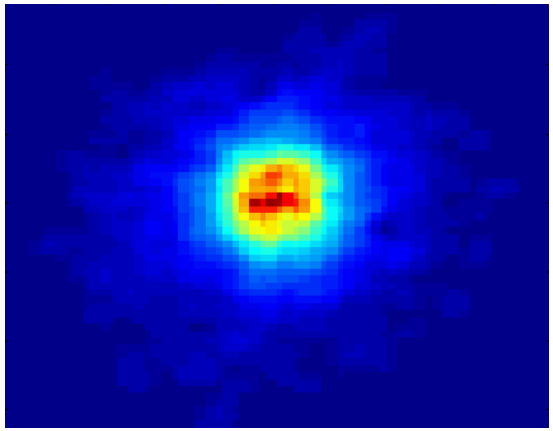
Enfoque:

- Utilice las transacciones con tarjeta de crédito y la información sobre su cuenta de titular como atributos.
  - ¿Cuándo comprar un cliente?, ¿qué quiere comprar?, ¿ con qué frecuencia se paga a tiempo?, etc.
- Etiquetar transacciones pasadas, fraude o transacciones justas. Esto forma el atributo de clase.
- Aprende un modelo para la clase de las transacciones.
- Utilice este modelo para detectar el fraude mediante la observación de las transacciones de tarjetas de crédito en una cuenta.

Clasificación

http://aps.umn.edu

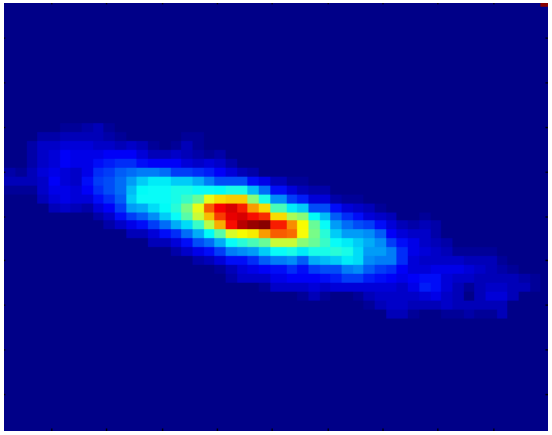
Temprano



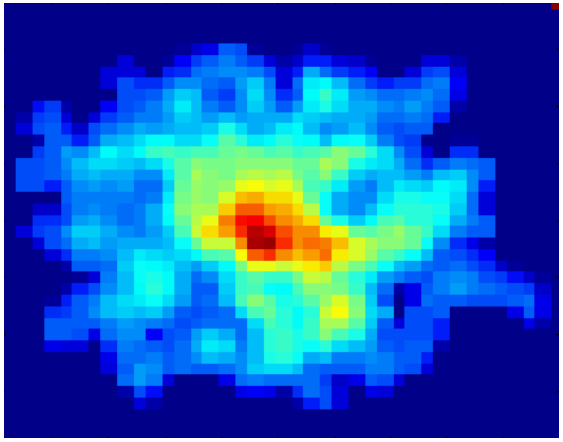
Clase:
- **Edo Formación**

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

Intermediate

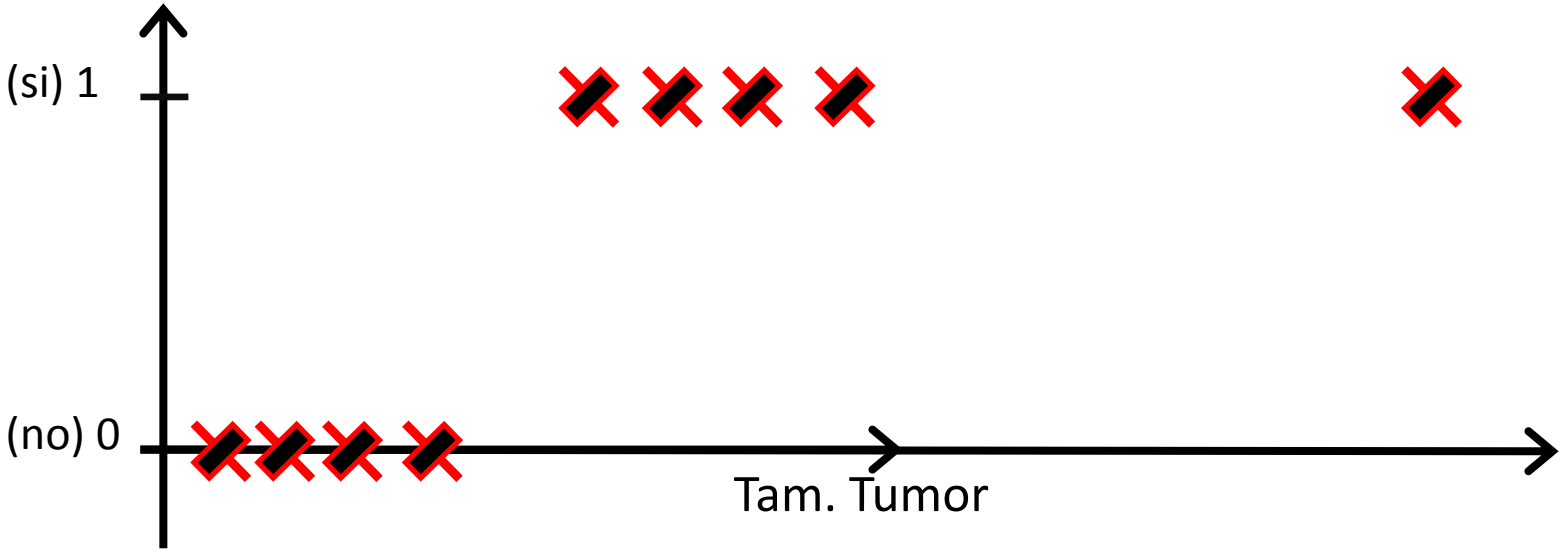


Tarde



Tam. datos:
- **72 millones estrellas, 20 millones galaxias**
- **BD Objectivo: 9 GB**
- **BD Imagen: 150 GB**

Clasificación



Umbral clasificador $h_\theta(x)$ = 0.5:

Si [redacted], predice “y = 1”

Si [redacted], predice “y = 0”

Regresión

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

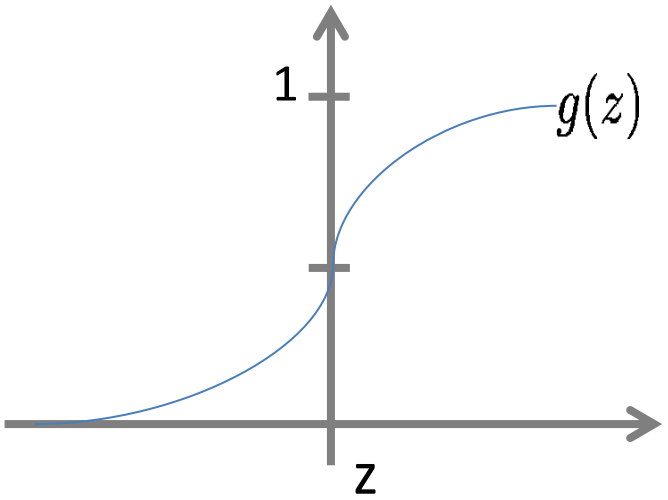
- Ejemplos:
  - Predecir las ventas de nuevos productos basados en gastos de publicidad.
  - La predicción de la velocidad del viento como una función de la temperatura, humedad, presión de aire, etc.
  - Predicción de series de tiempo de los índices bursátiles.

Clasificación y Regresión

Regresión Lógica : 

Regresión Lógica

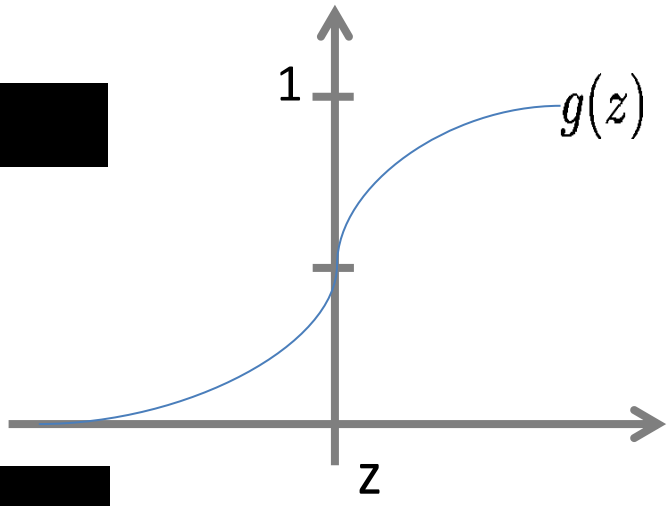
[redacted]



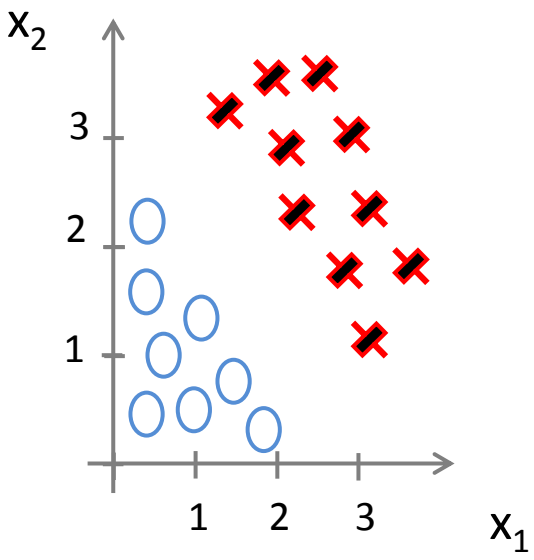
Regresión Lógica:
Barrera de decisión

predice "$y = 1$" si [redacted]

predice "$y = 0$" si [redacted]



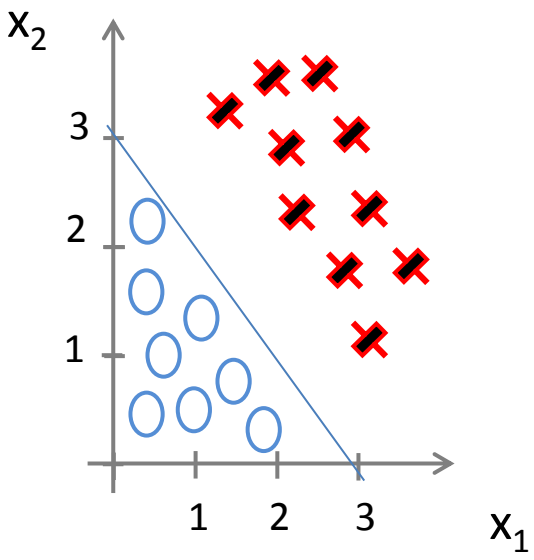
Regresión Lógica: Barrera de decisión



[redacted]

Predice "[redacted]" si [redacted]

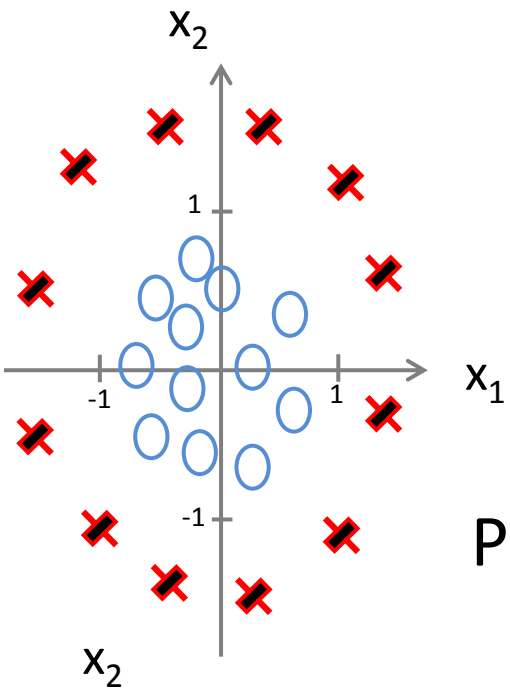
Regresión Lógica:
Barrera de decisión



[redacted]

Predice "[redacted]" si [redacted]

Regresión Lógica:
Barrera de decisión

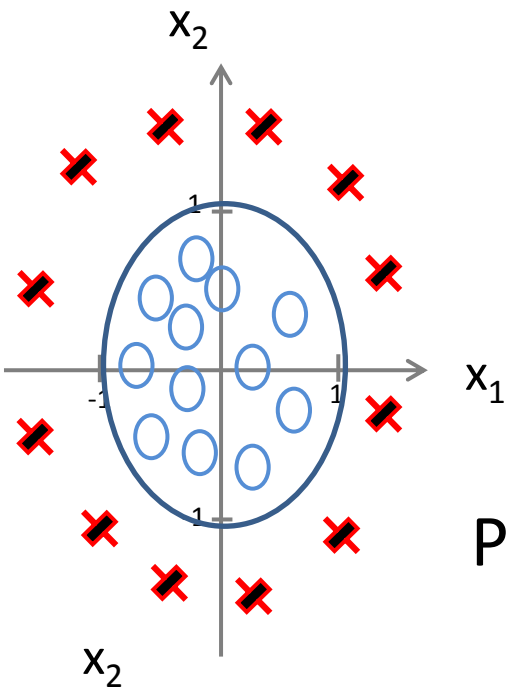


[redacted]

[redacted]

Predice "[redacted]" si [redacted]

Regresión Lógica:
Barrera de decisión



[redacted]

[redacted]

Predice "[redacted]" si [redacted]

Regresión Lógica: Función de costos

Conjunto de entrenamiento: [redacted]

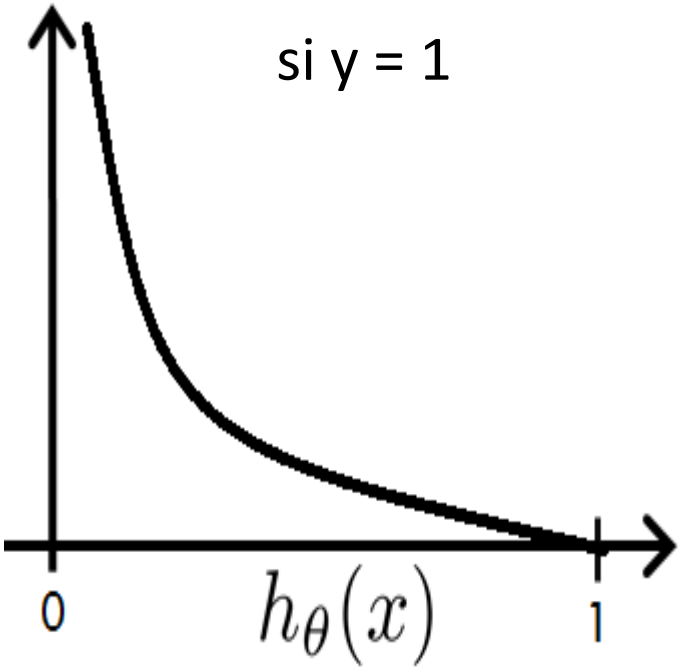
m ejemplos

$$x \in \begin{bmatrix} x_0 \\ x_1 \\ \dots \\ x_n \end{bmatrix} \qquad y \in \{0, 1\}$$

[redacted]

Como escoger los parámetros? [redacted]

Regresión Lógica: Función de costos



$Cost = 0$ si $y=1$, $h_\theta(x) = 1$

Pero cuando

$$h_\theta(x) \to 0$$

$$Cost \to \infty$$

(predice $P(y = 1 \mid x; \theta) = 0$)

Regresión Lógica:
Función de costos

[redacted]

[redacted]

[redacted]

Gradiente descendiente

[redacted]

Queremos [redacted] :

Repeat {

[redacted]

(simultaneously update all θ_j)

}

Función de costos

Regresión lineal: 

Optimización

$J(\theta)$ [redacted]

Dado θ , queremos calcular

- $J(\theta)$
- [redacted]

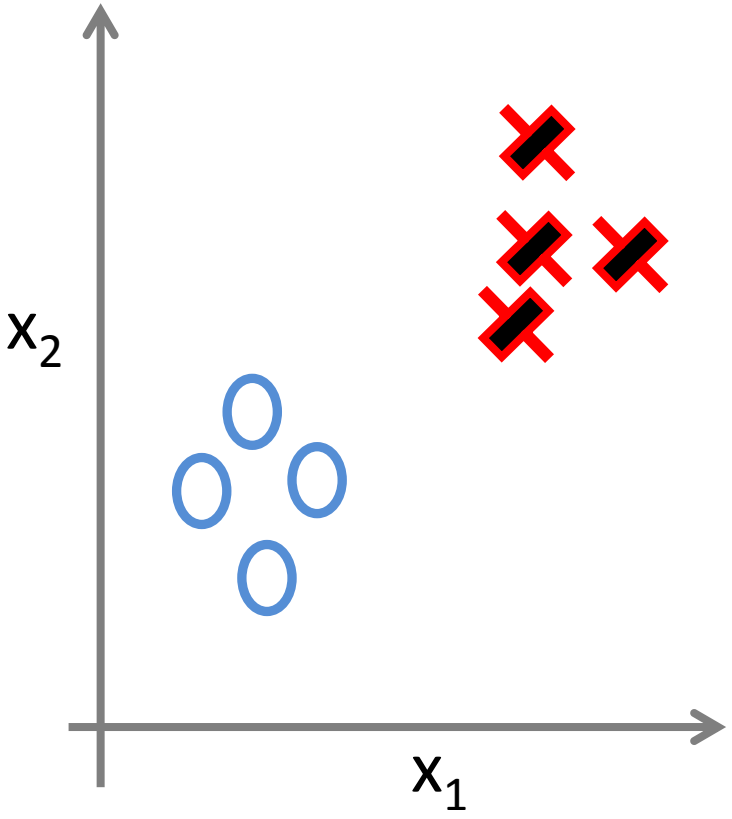
(for [redacted] {

[redacted]

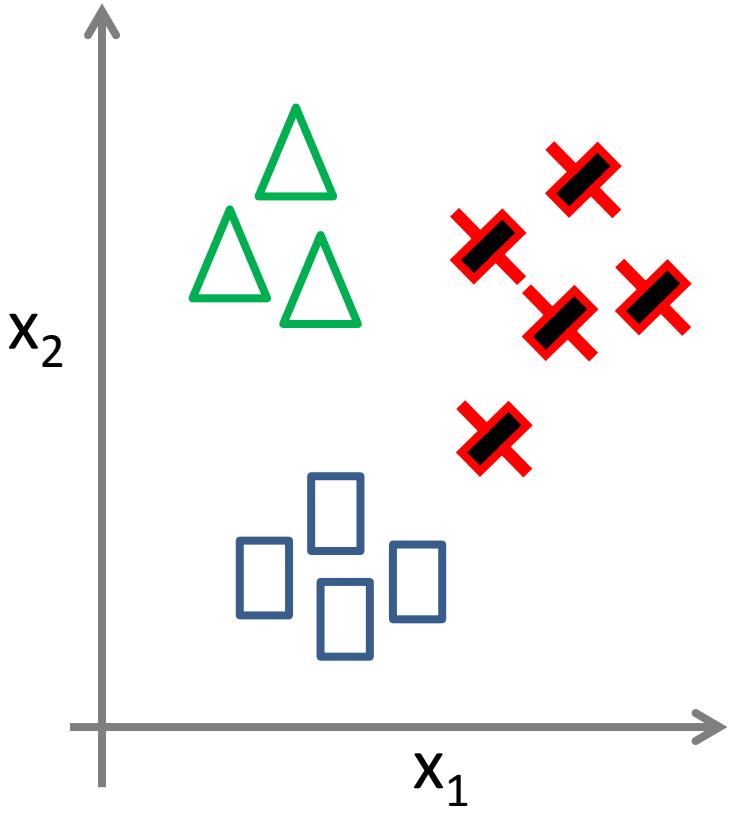
}

Clasificación multi-clases

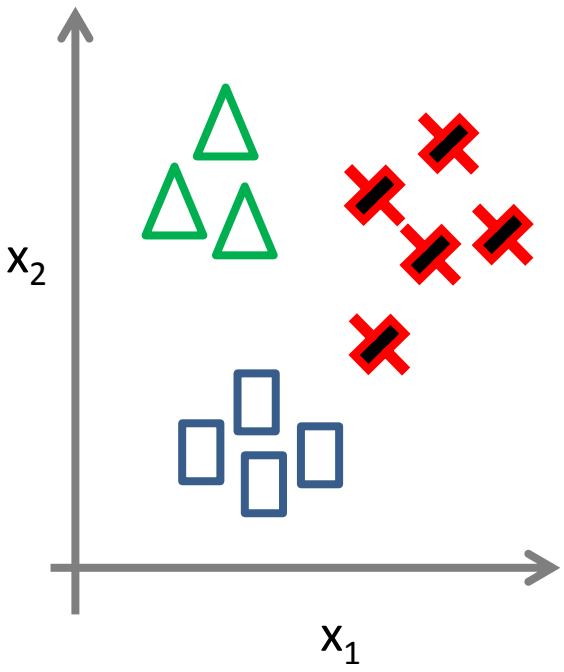
Clasificación Binaria

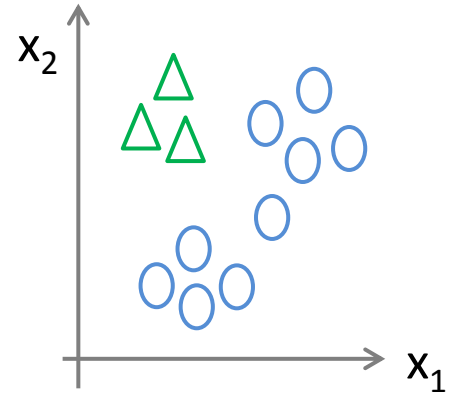


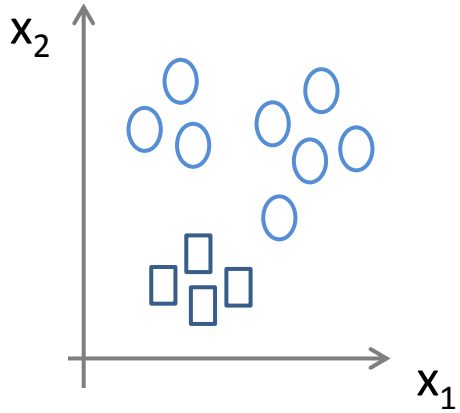
Clasificación Multi-clases

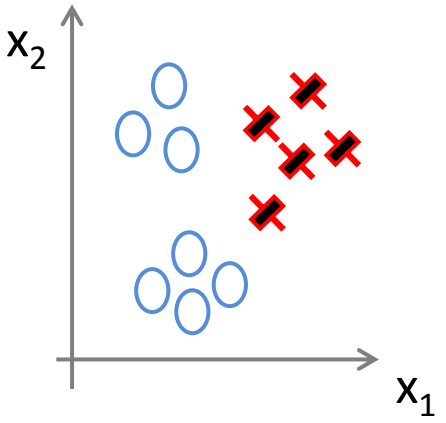


Una-vs-todos (uno-vs-resto):









Clase 1:
Clase 2:
Clase 3:

$(i = 1, 2, 3)$

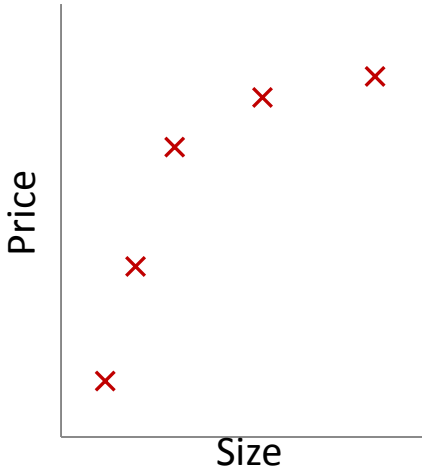
Uno vs Todo

Entrenar a un clasificador de regresión logistica $h_\theta^{(i)}(x)$ por cada clase [redacted] para predecir la probabilidad de [redacted]

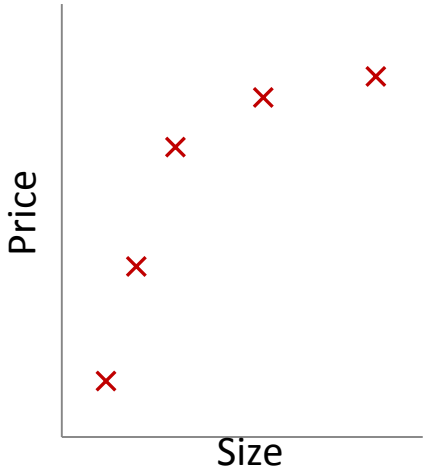
Con una nueva entrada [redacted], para hacer predicción se toma la clase [redacted] que maximice

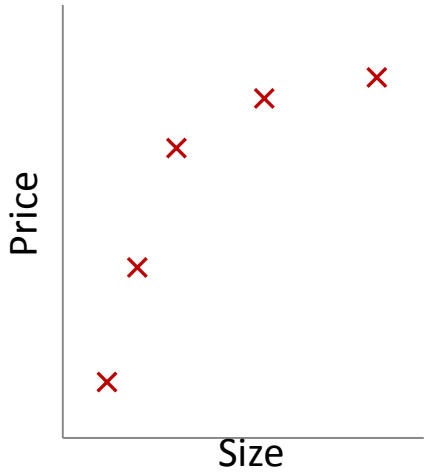
[redacted]

Sobre-ajustamiento (Overfitting)

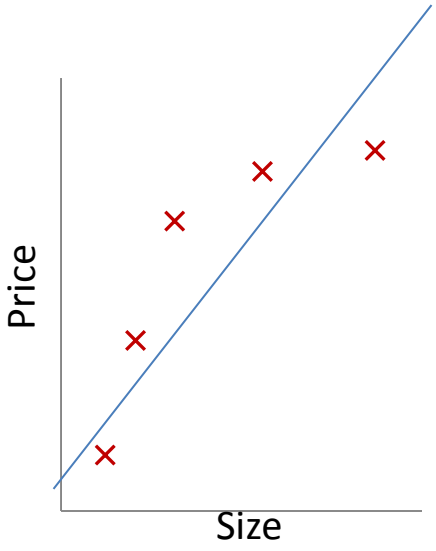


$\theta_0 + \theta_1 x$

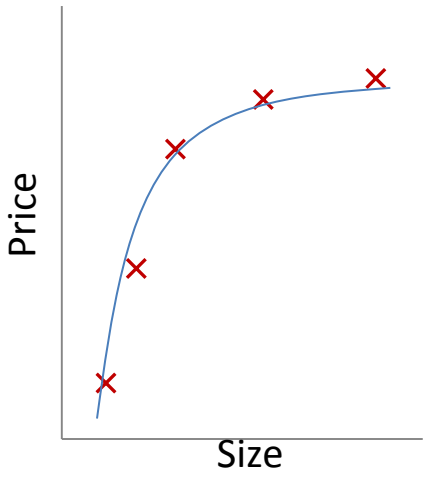




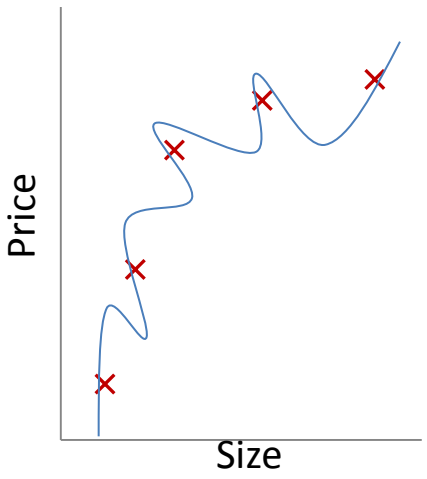
Sobre-ajustamiento (Overfitting)



$\theta_0 + \theta_1 x$

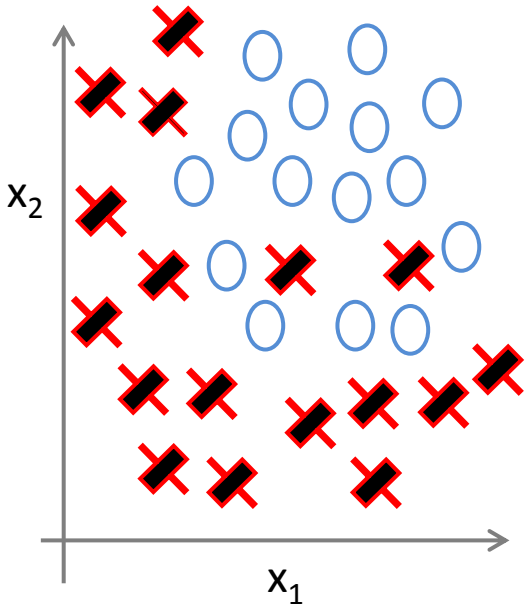


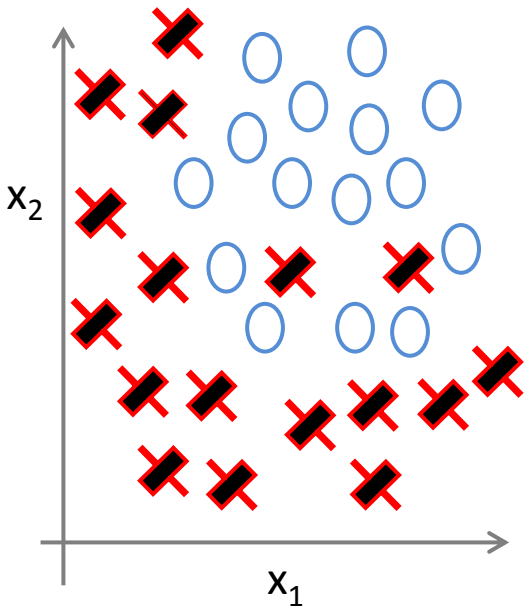
[redacted]

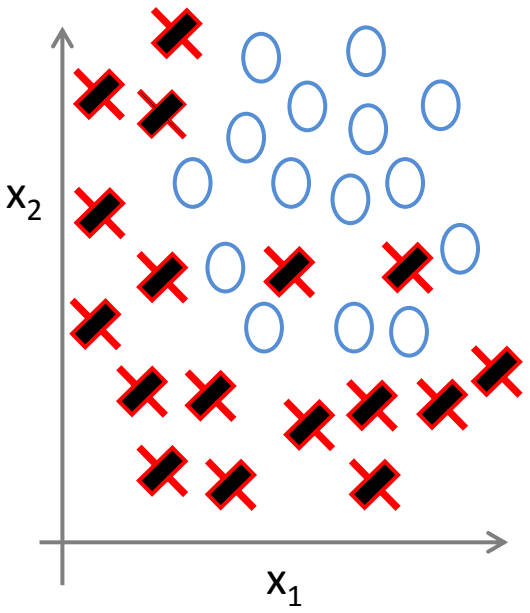


[redacted]

Sobre-ajustamiento (Overfitting)



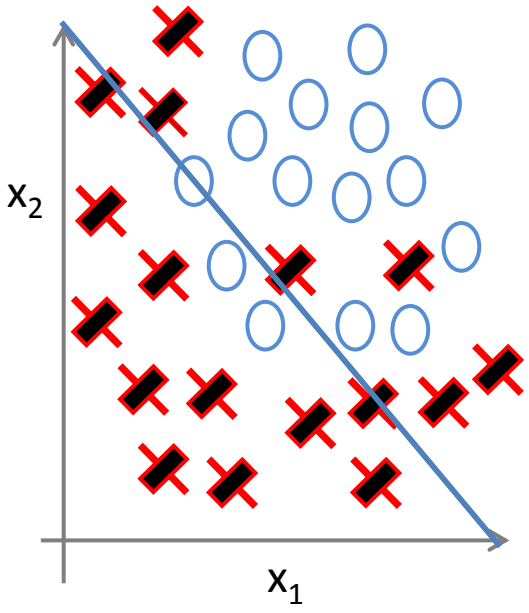


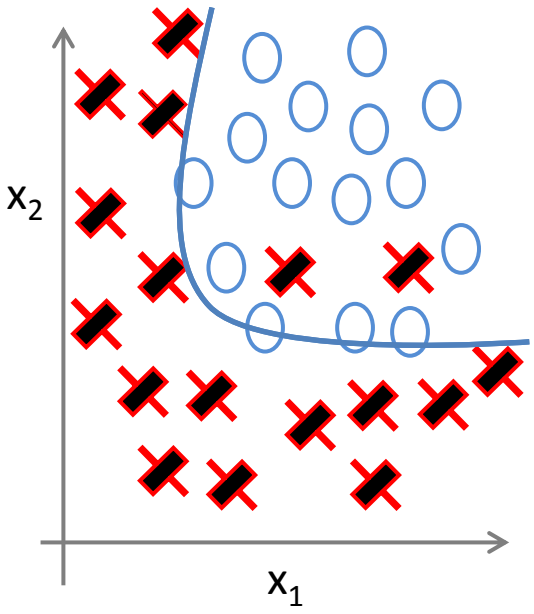


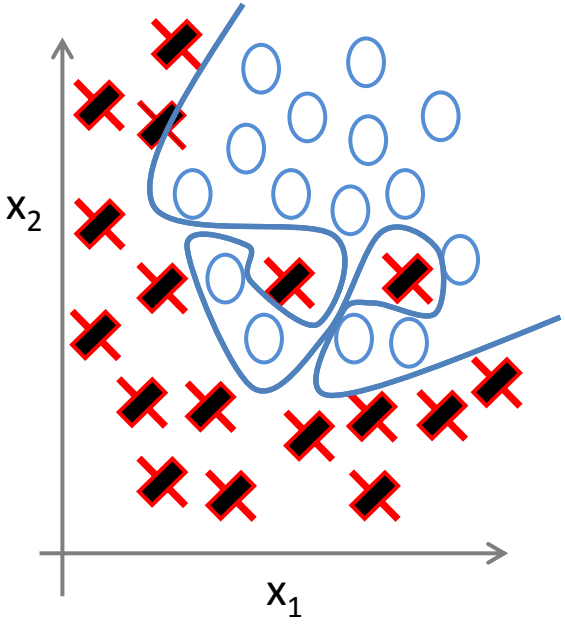
$$h_\theta(x) = g(\theta_0 + \theta_1 x_1 + \theta_2 x_2)$$

(g = función sigmoidal)

sigmoidal







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

Sobre-ajustamiento (Overfitting)

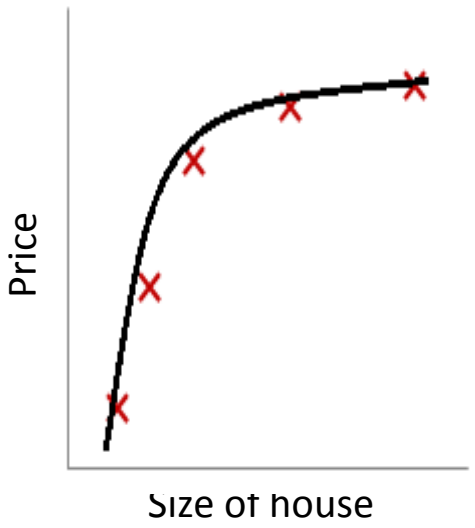
Opciones:

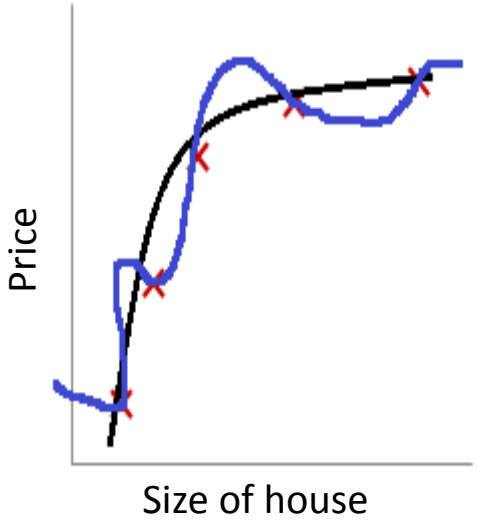
- Reducir el número de características.
  - Seleccionar manualmente las características que desea conservar.
  - Algoritmo de selección de modelo.

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

.

Función de costo





[redacted]

[redacted]

Supongamos que penalizamos $\theta_3\ \theta_4$

[redacted]

Patrones Secuenciales

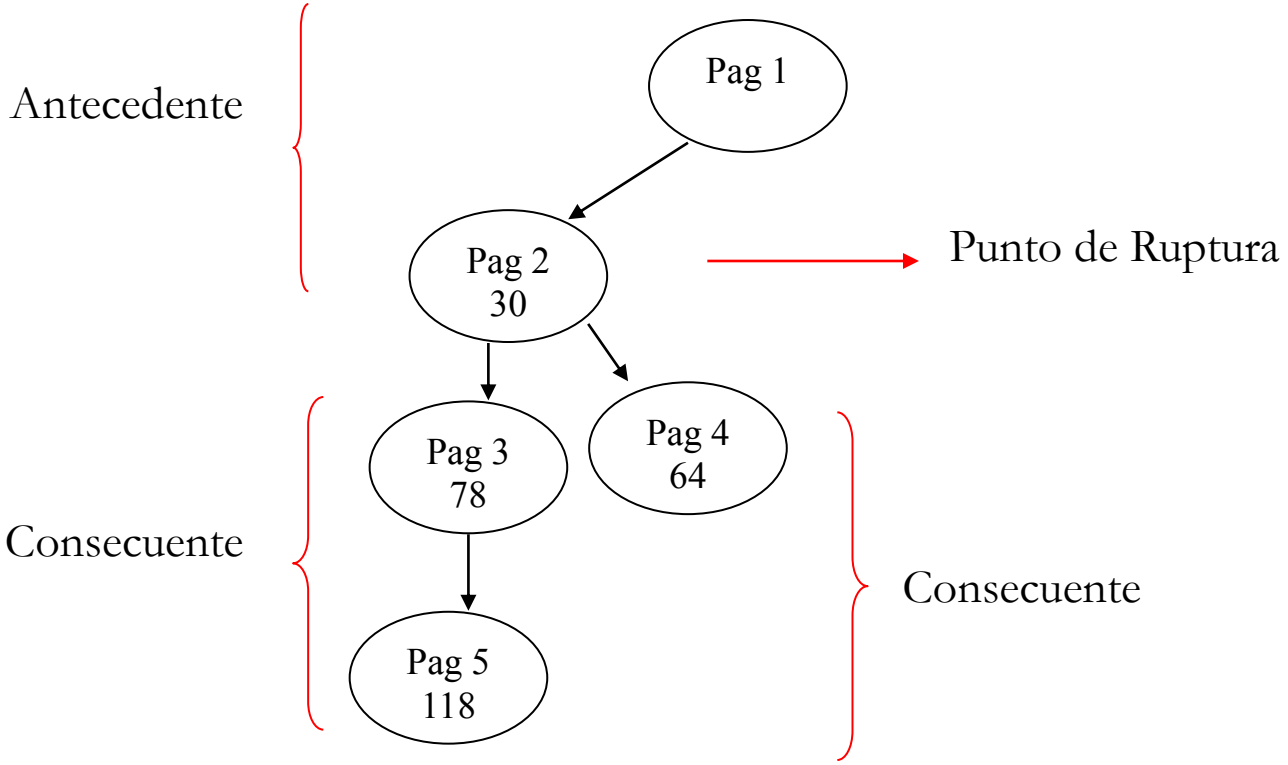
- Descubrir patrones en los cuales la presencia de un conjunto de ítems es seguido por otro ítem en orden temporal.
- Ejemplo: Encontrar y predecir el comportamiento de los visitantes de un sitio Web con respecto al tiempo.

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

[/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.

Modelos de Agrupamiento, Segmentación y Asociación

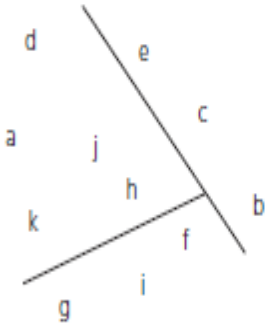
Agrupamiento (Clustering)

Dado un conjunto de puntos 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 más similares entre sí.
- Los puntos de datos en grupos separados son menos similares entre sí.

Clusters

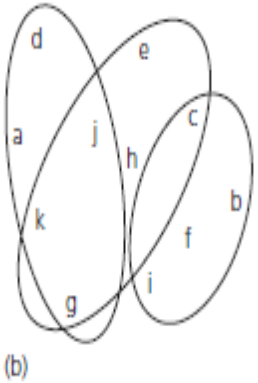
A menudo una técnica muy usada en la cual se infiere un **árbol de decisión o conjunto de reglas** que asigna a cada instancia al grupo al que pertenece



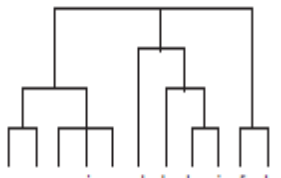
(a)

	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)

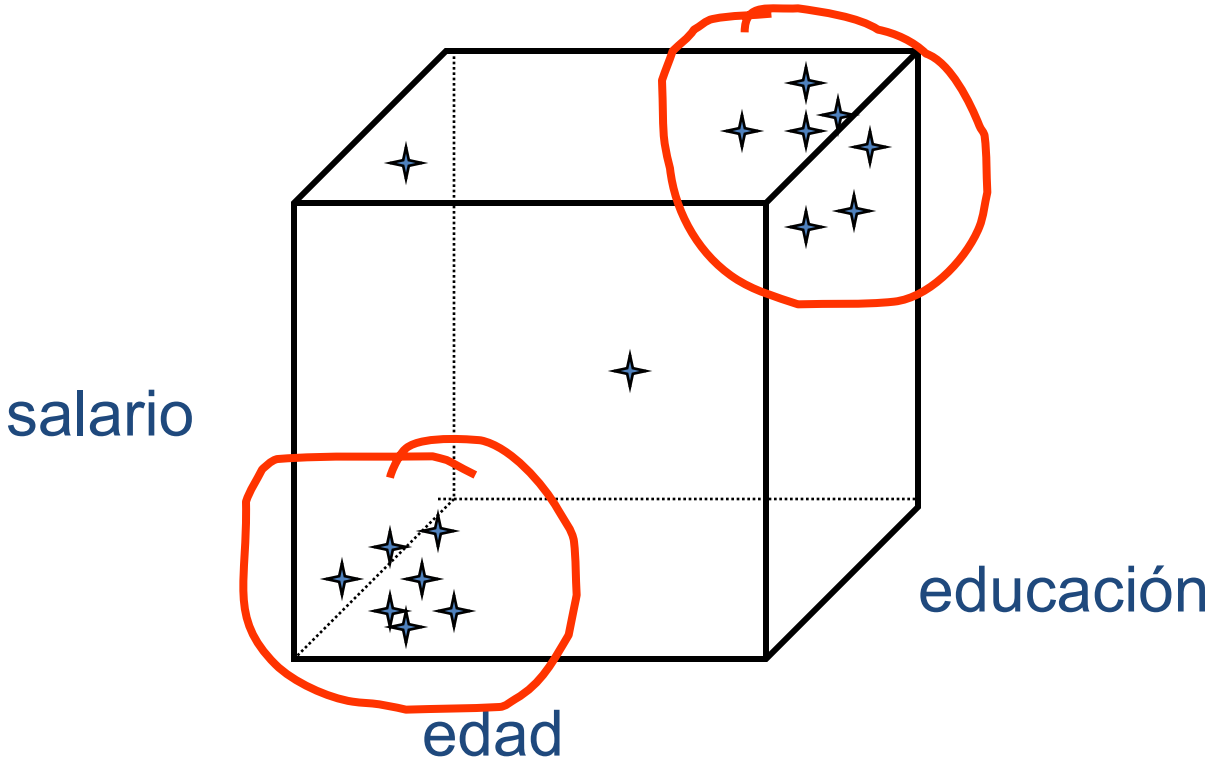


(b)



(d)

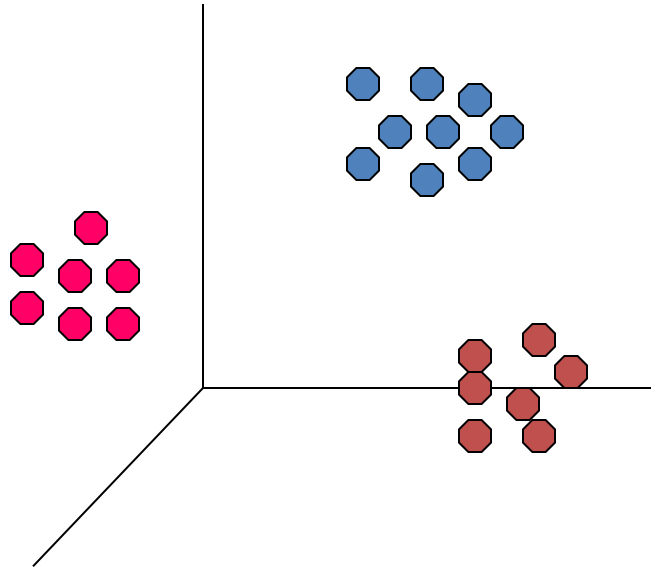
Agrupamiento (Clustering)



Agrupamiento (Clustering)

Distancias Intracluster son minimizadas

Distancias Intercluster son maximizadas



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. Úsalo para clúster.

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*	*Correcto asignado*
Financiero	555	364
Extranjero	341	260
Nacional	273	36
Ciudad	943	746
Deportes	738	573
Entretenimiento	354	278

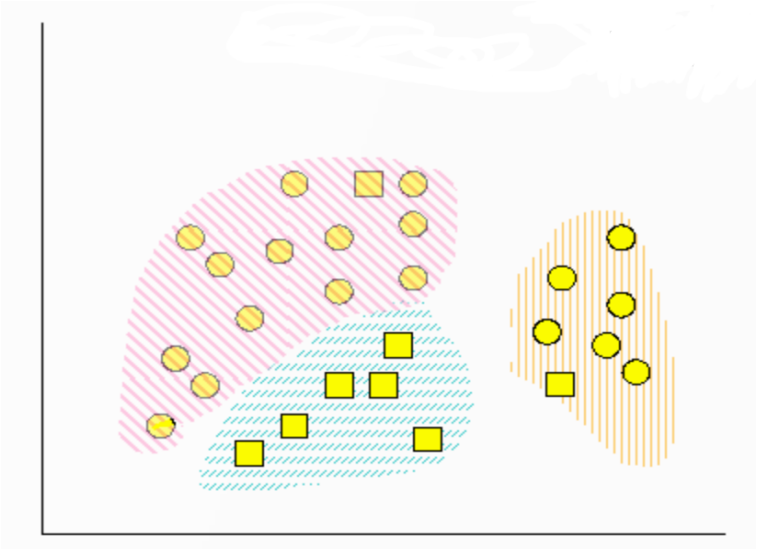
Tipos de clustering

- **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

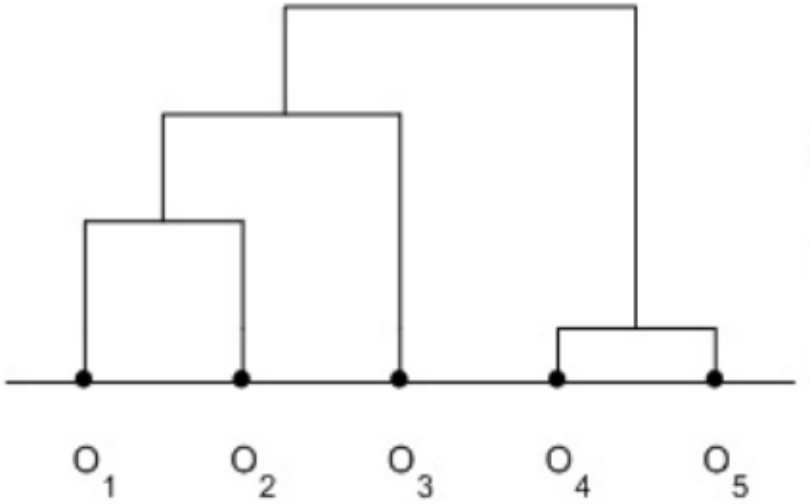
- **Clustering ascendente jerárquico**
  - Crear un dendograma, es decir, crear un conjunto de agrupaciones anidadas hasta construir un árbol jerárquico

Clusterización

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



a) Particional



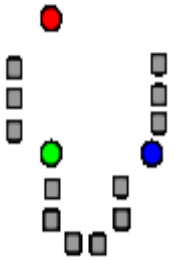
b) Jerárquico

Algoritmo K-medias

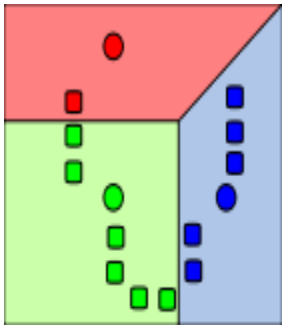
- Método más utilizado de clustering particional
- La idea es situar los prototipos o centros en el espacio, de forma que los datos pertenecientes al mismo prototipo tengan características similares
- Los datos se asignan a cada centro según la menor distancia, normalmente usando la distancia euclídea
- 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.

Algoritmo K-medias

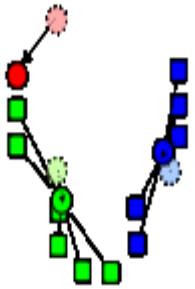
es un método de agrupamiento, que tiene como objetivo la partición de un conjunto (n) en k grupos en el que cada observación pertenece al grupo más cercano a la media.



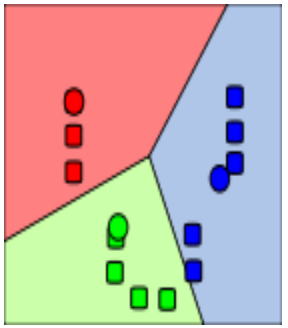
1) *k* centroides iniciales generados aleatoreamente (en este caso *k*=3)



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.

Clusterización: Algoritmo K-Medias

1.Seleccionar centroides aleatorios
2. Asignar cada objeto al grupo cuyo centróide sea el más cercano al objeto.

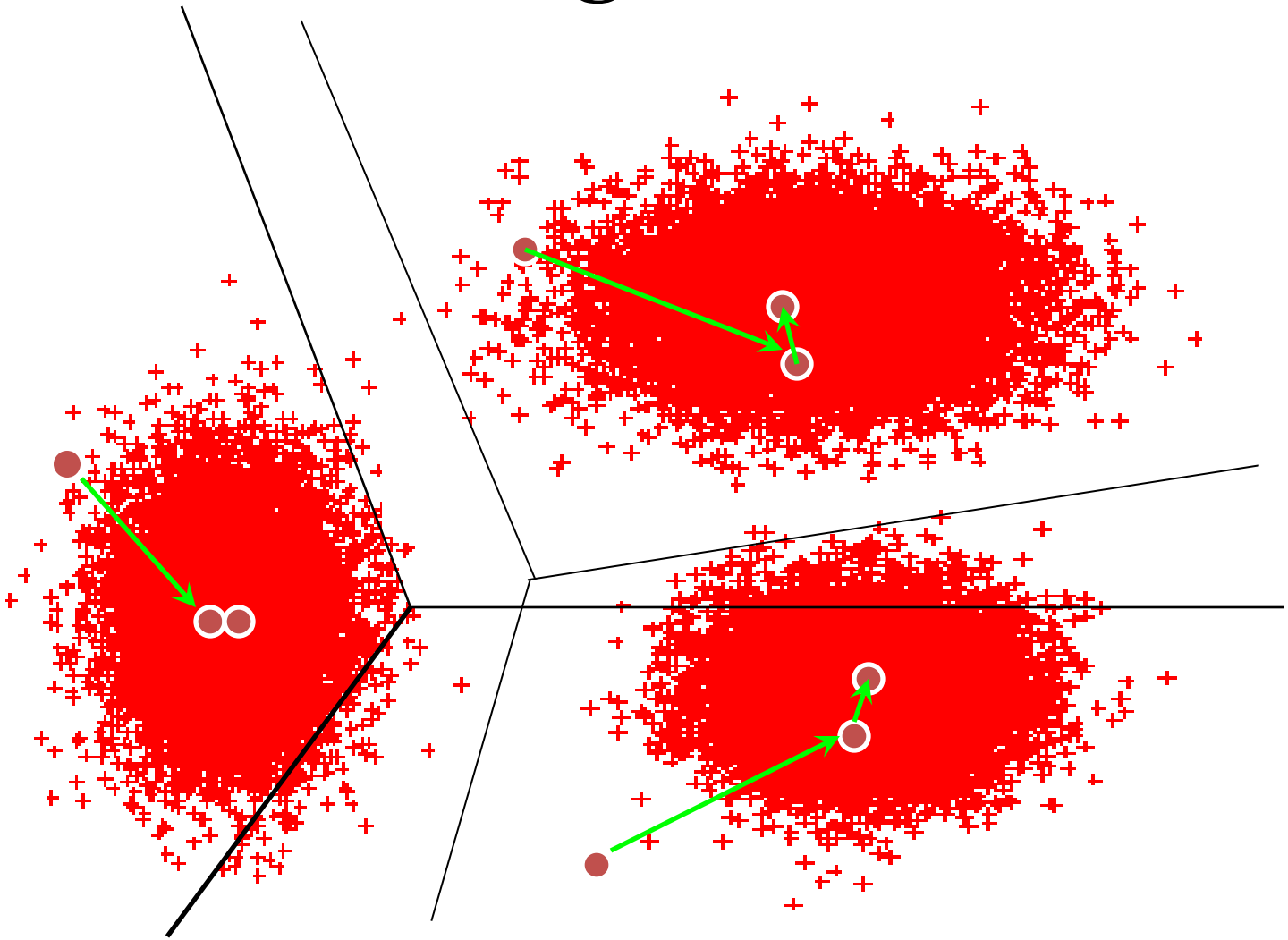
3. Cuando todos los objetos hayan sido asignados, recalcular la posición de los k centróides.

4. Repetir los pasos 2 y 3 hasta que los centróides no varíen

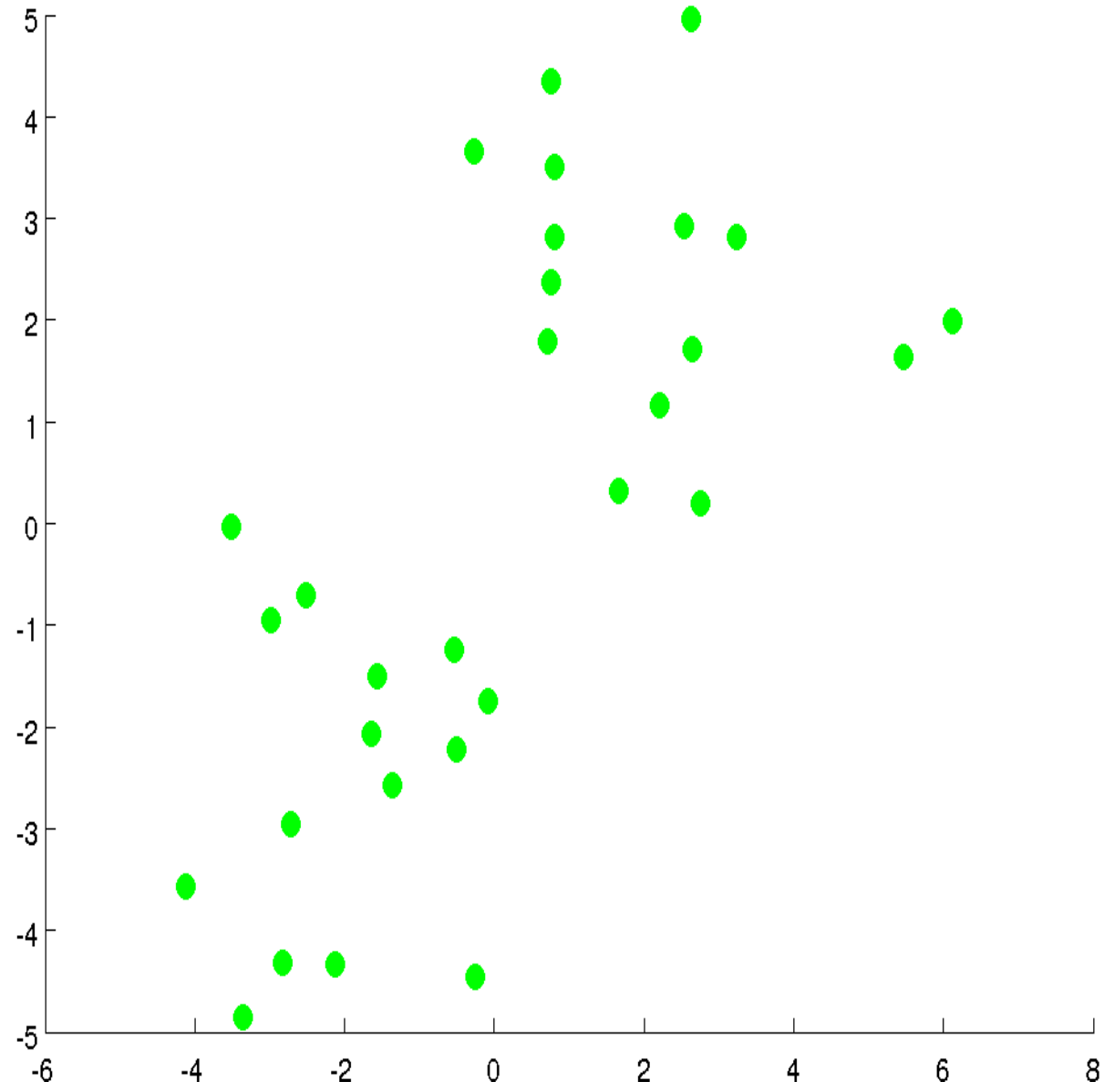
Distancia Euclídea $\delta^2 E\ (X_i, X_j) = ||\ X_i - X_j||2 = (X_i - X_j)T(X_i - X_j)$

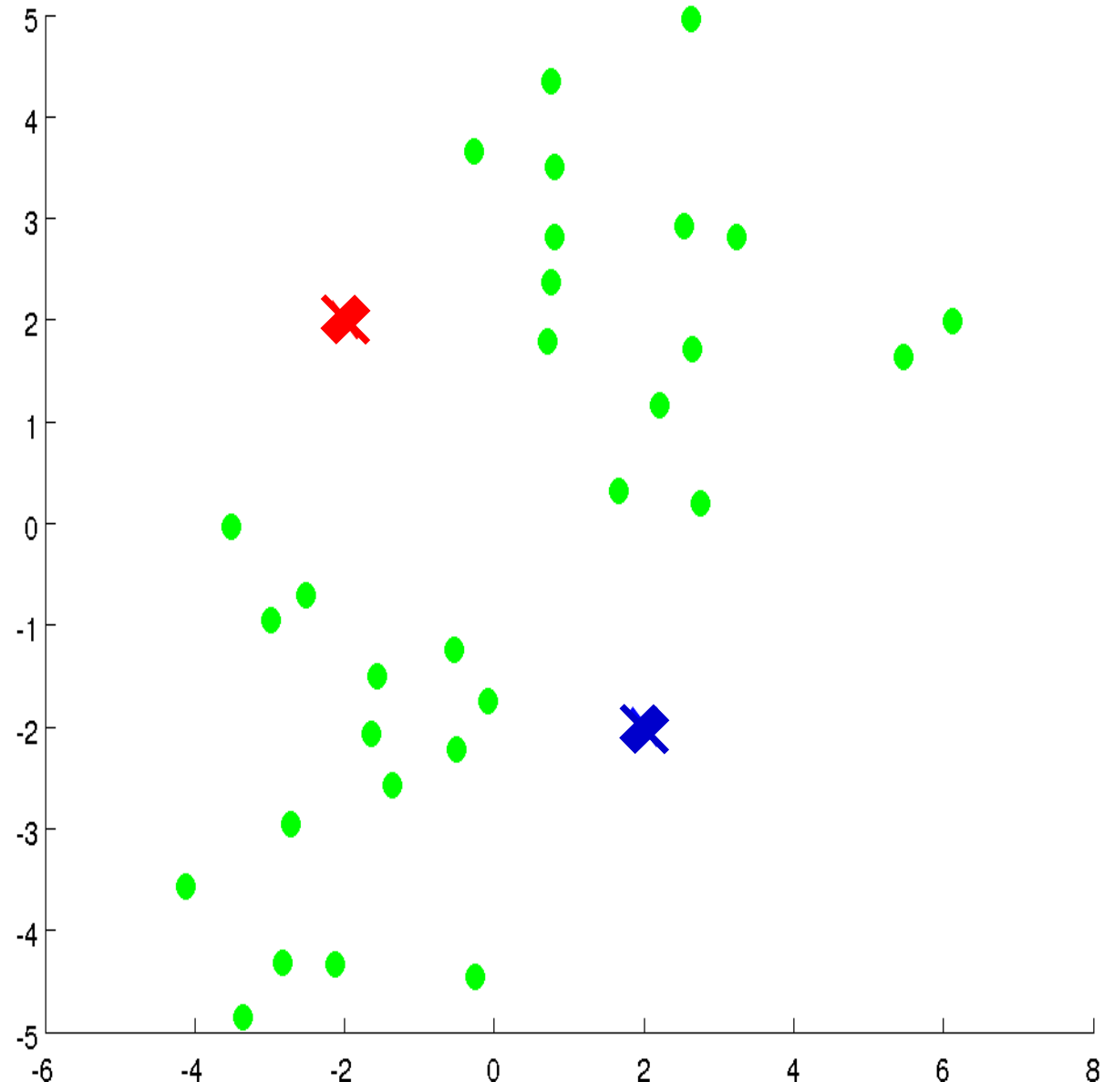
Determinar la media de cada grupo

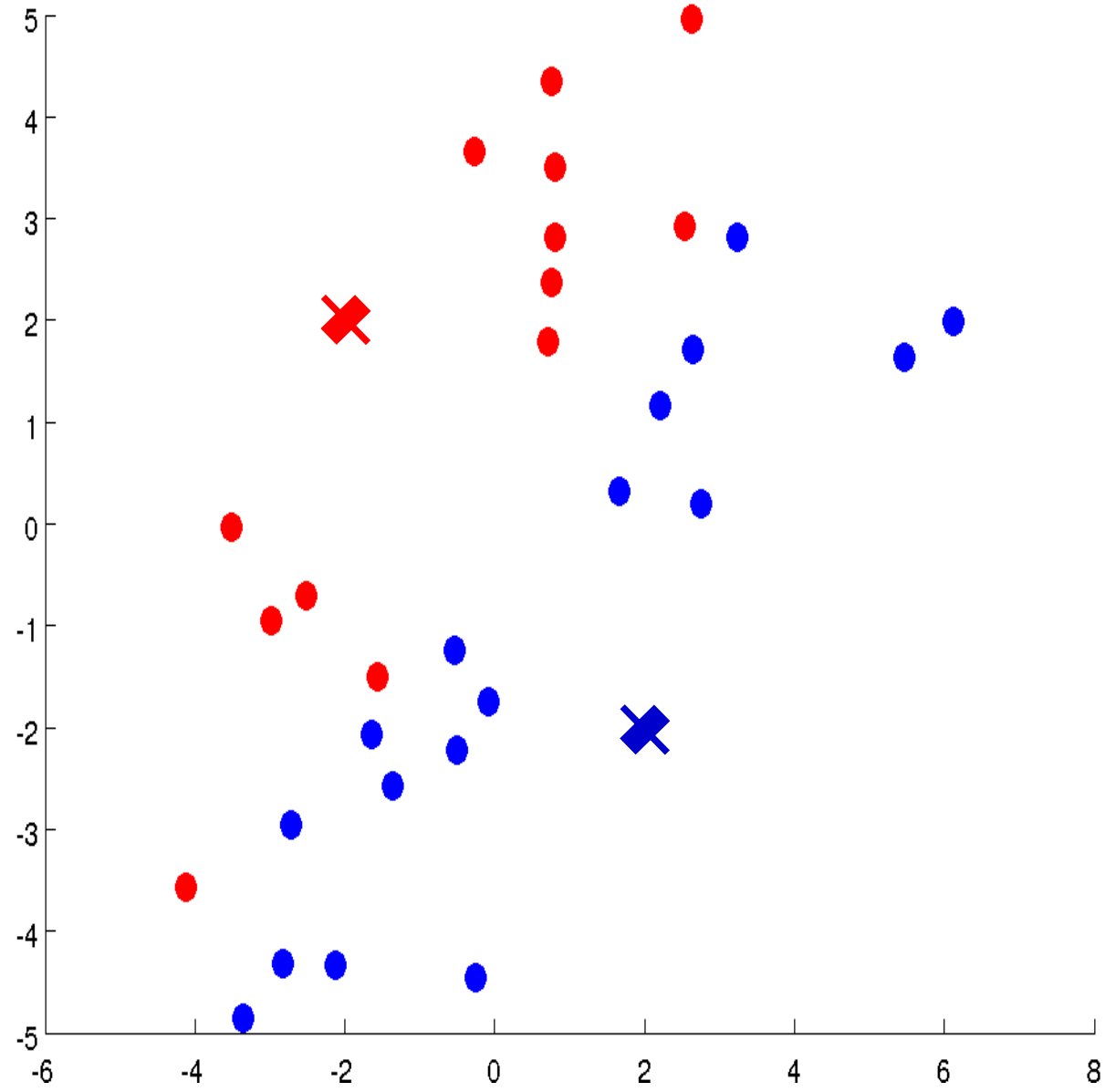
Clusterización: Algoritmo K-Medias

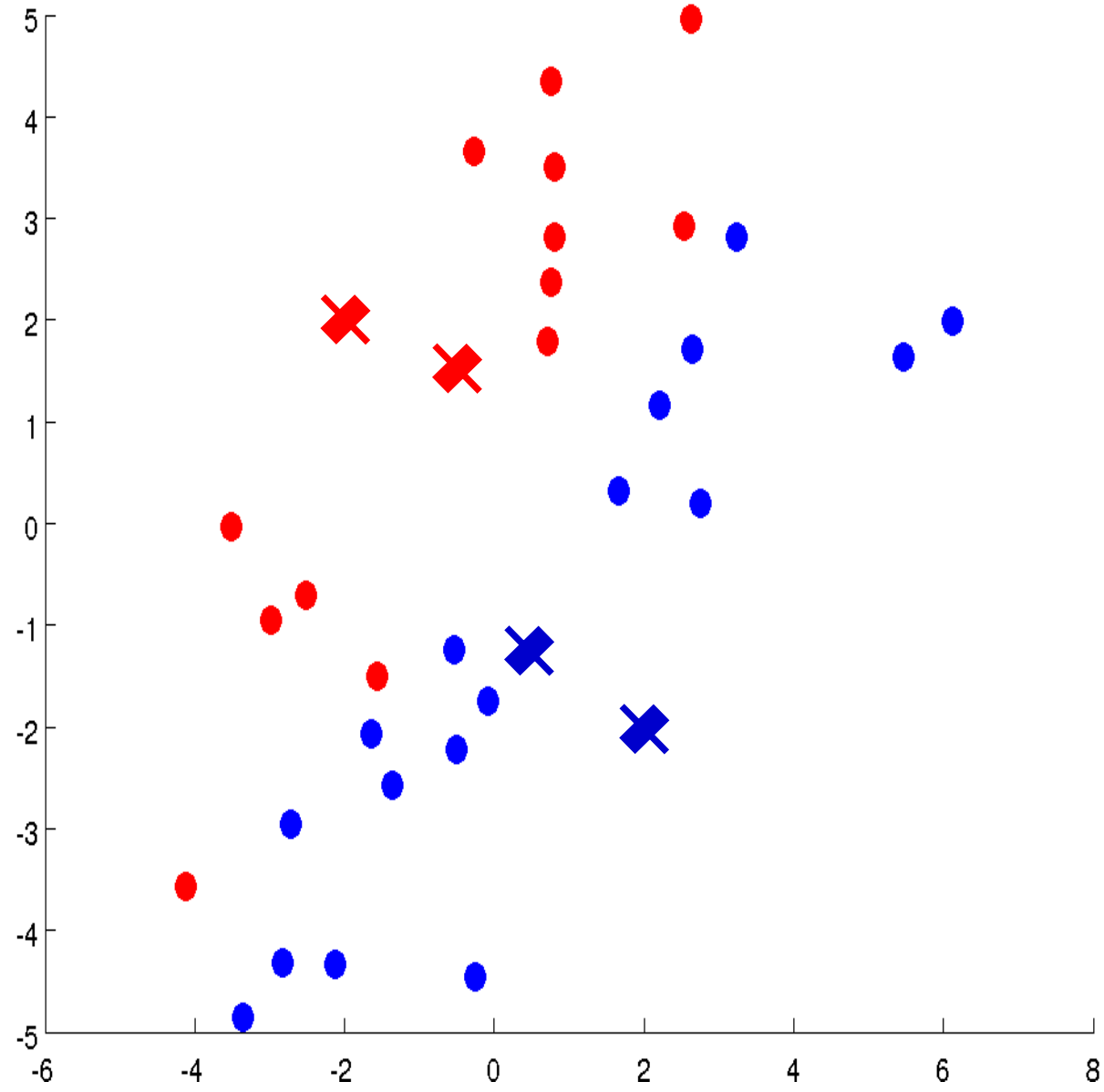


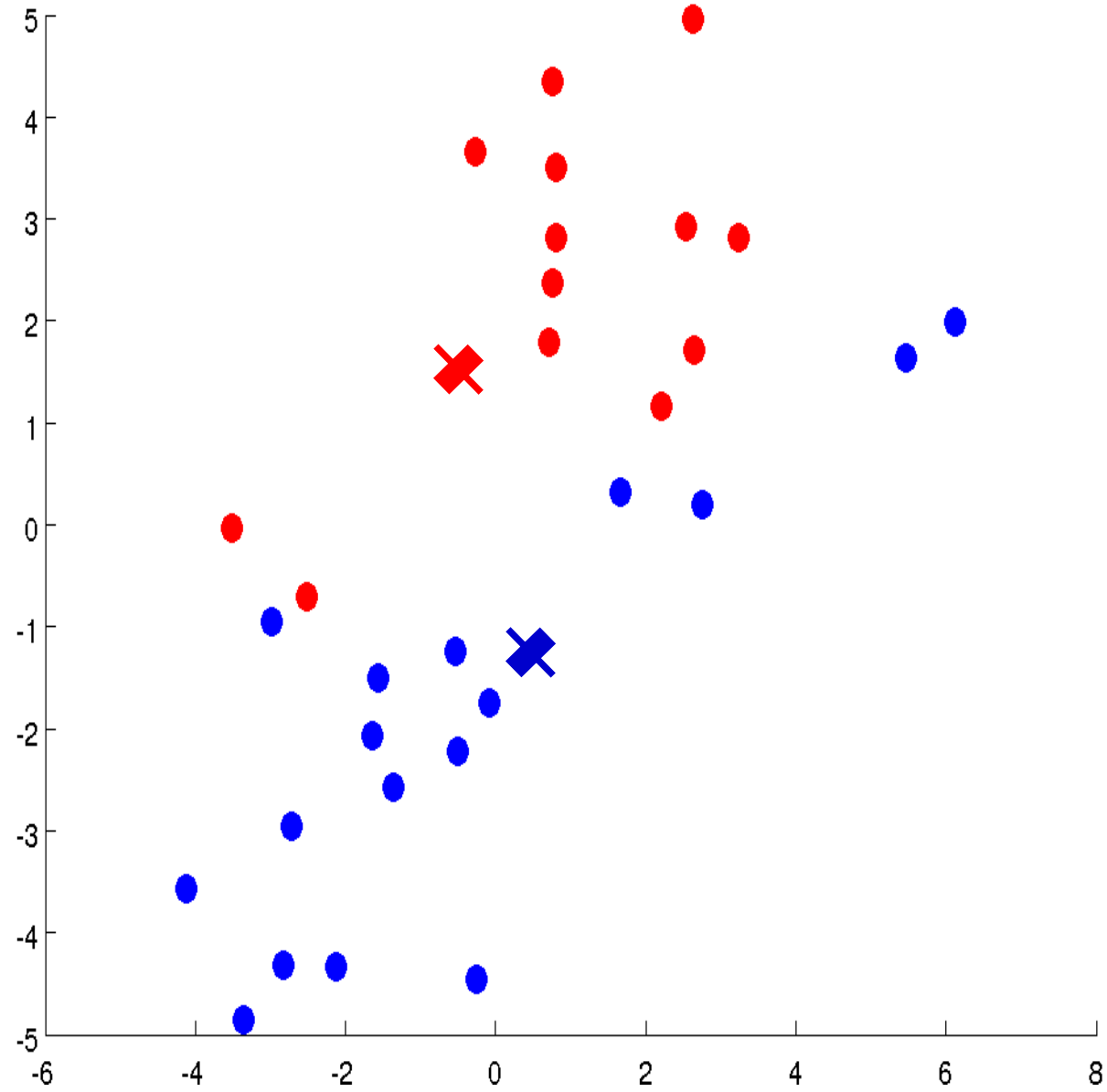
Corrida en frio K-means

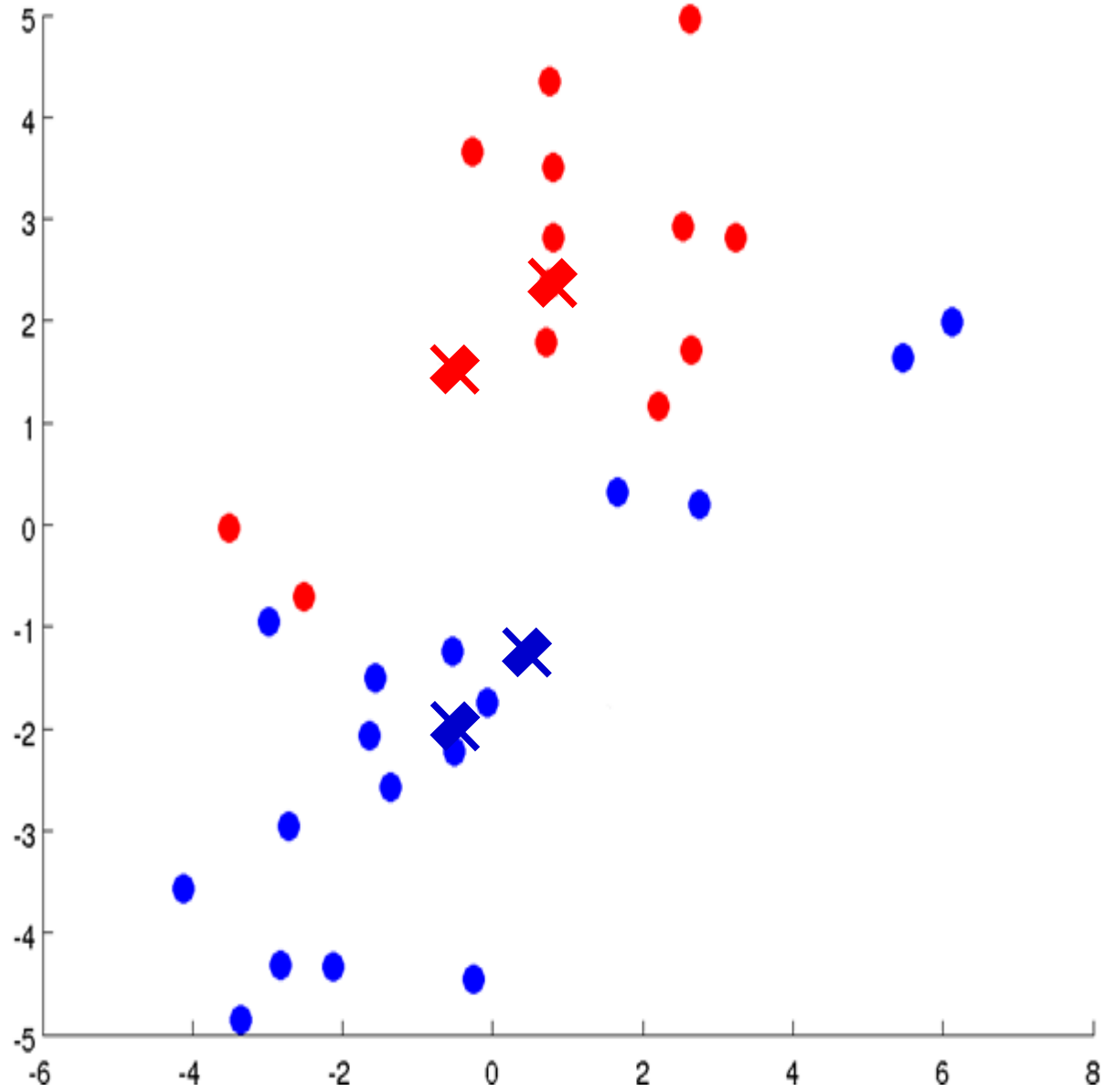

5
4
3
2
1
0
-1
-2
-3
-4
-5
-6
-4
-2
0
2
4
6
8

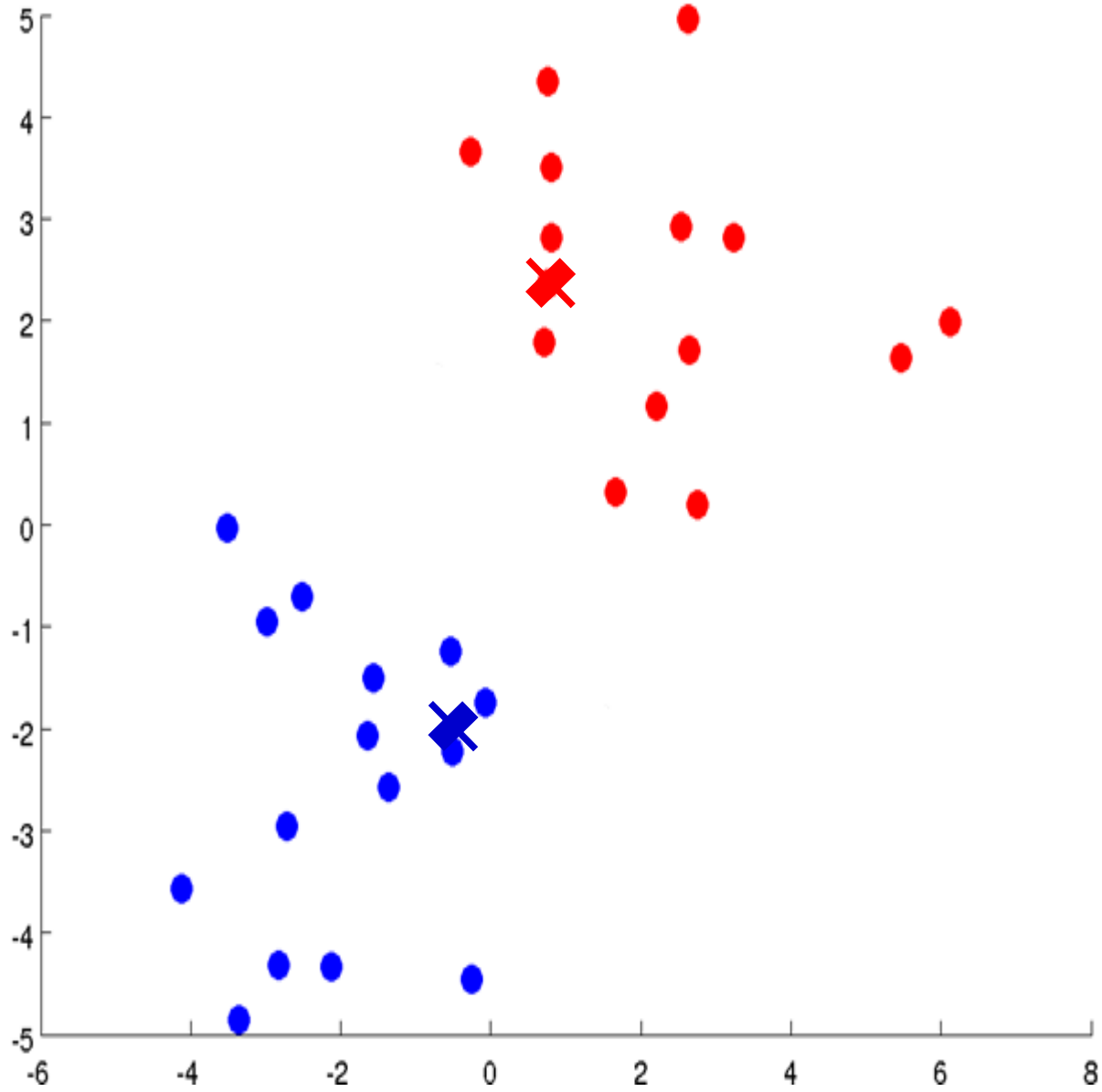

5
4
3
2
1
0
-1
-2
-3
-4
-5
-6
-4
-2
0
2
4
6
8

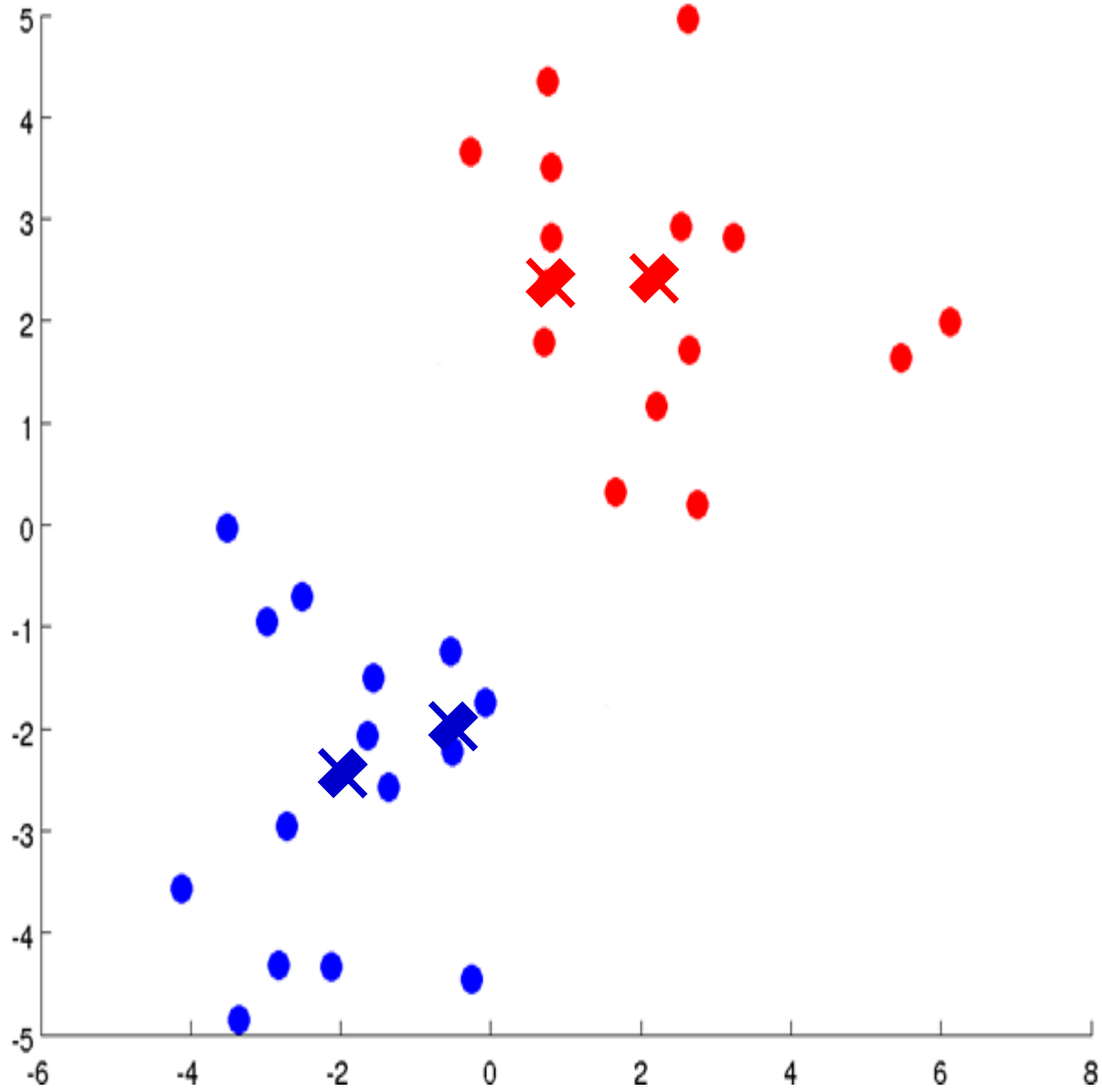

5
4
3
2
1
0
-1
-2
-3
-4
-5
-6
-4
-2
0
2
4
6
8

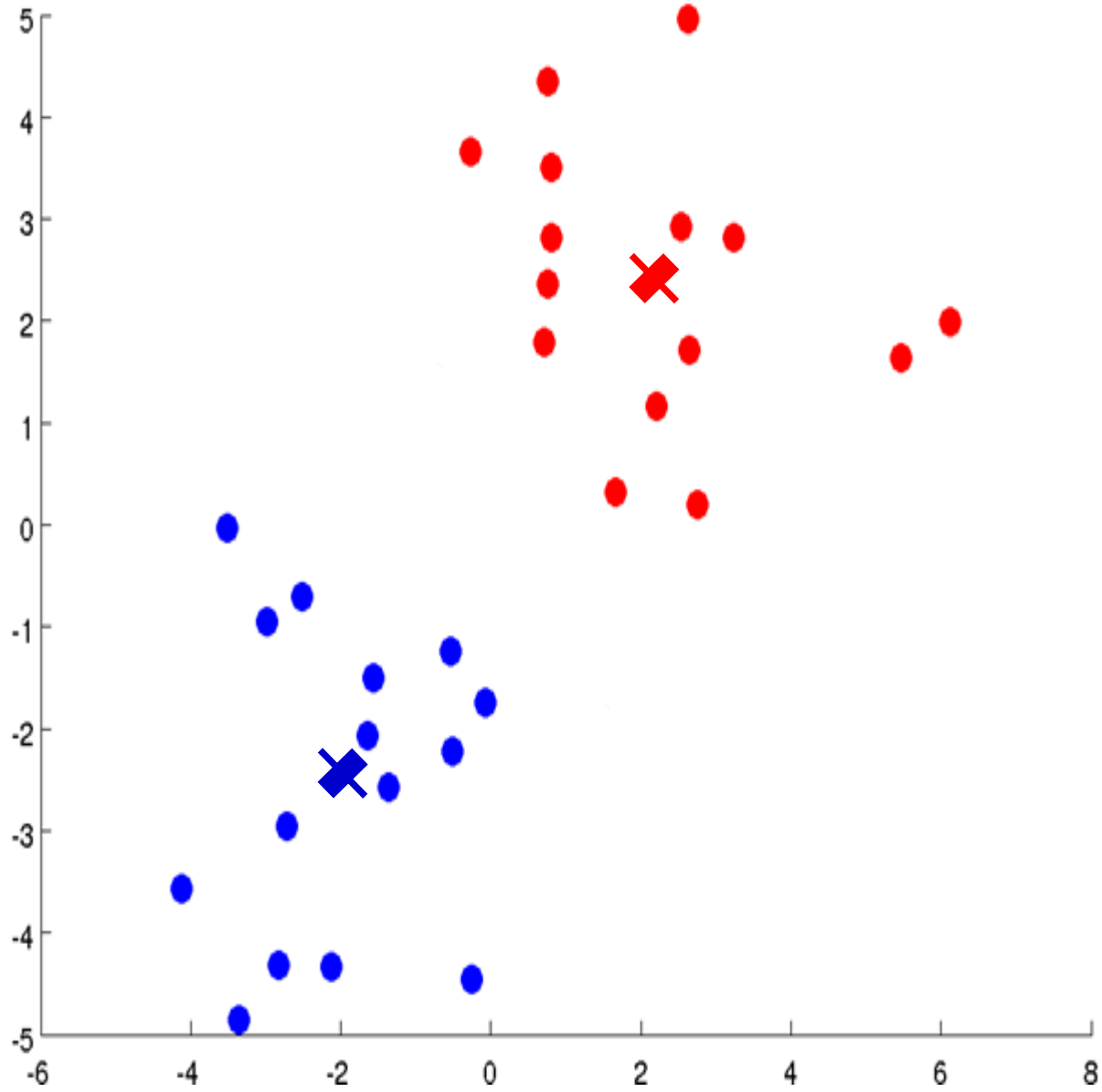

5
4
3
2
1
0
-1
-2
-3
-4
-5
-6
-4
-2
0
2
4
6
8


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

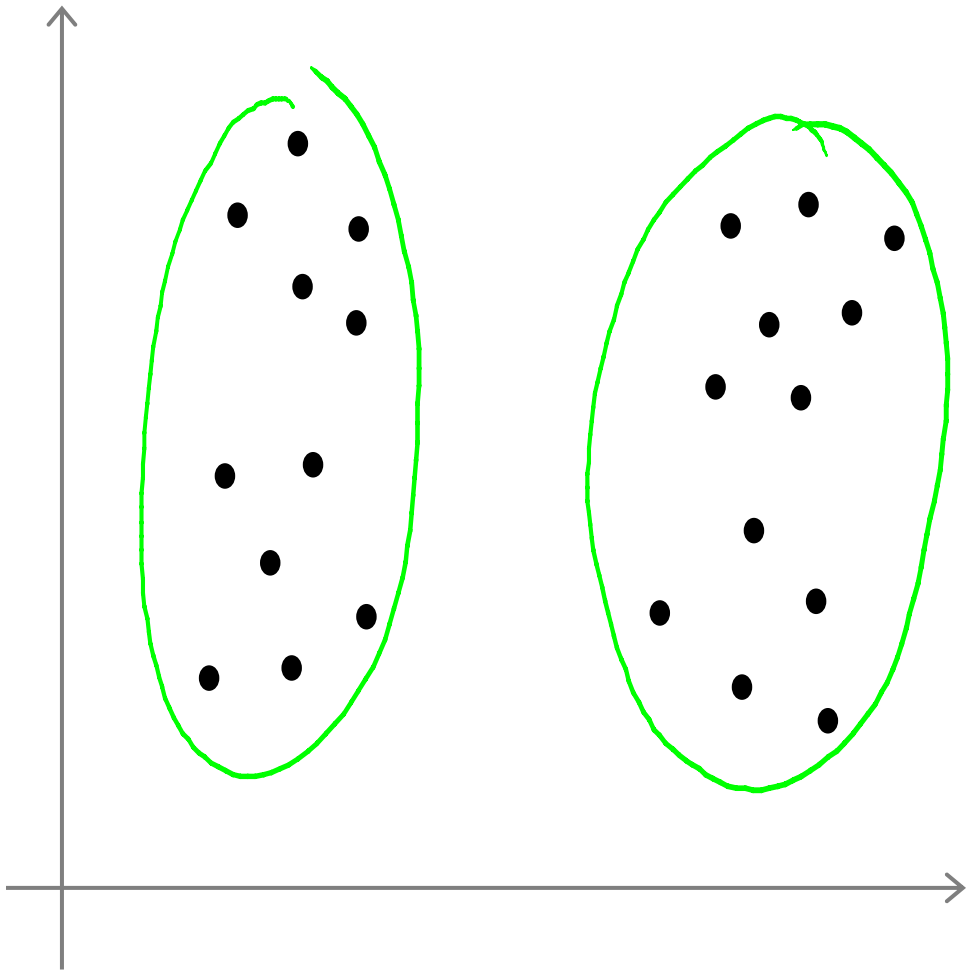

5
4
3
2
1
0
-1
-2
-3
-4
-5
-6
-4
-2
0
2
4
6
8


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

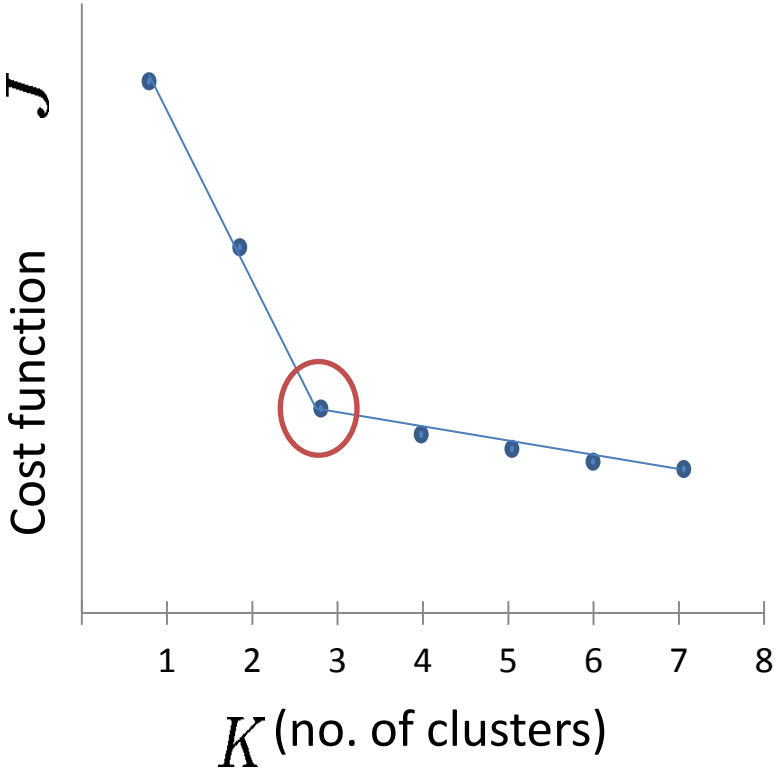

5
4
3
2
1
0
-1
-2
-3
-4
-5
-6
-4
-2
0
2
4
6
8

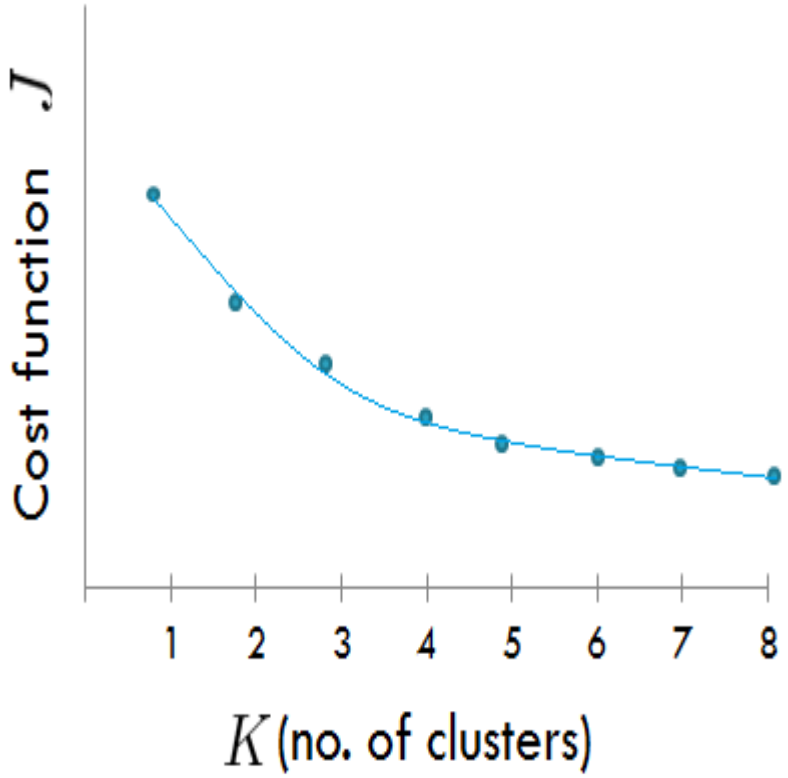
Cual es el correcto valor de K?



Escogiendo el valor de K

Método del codo:



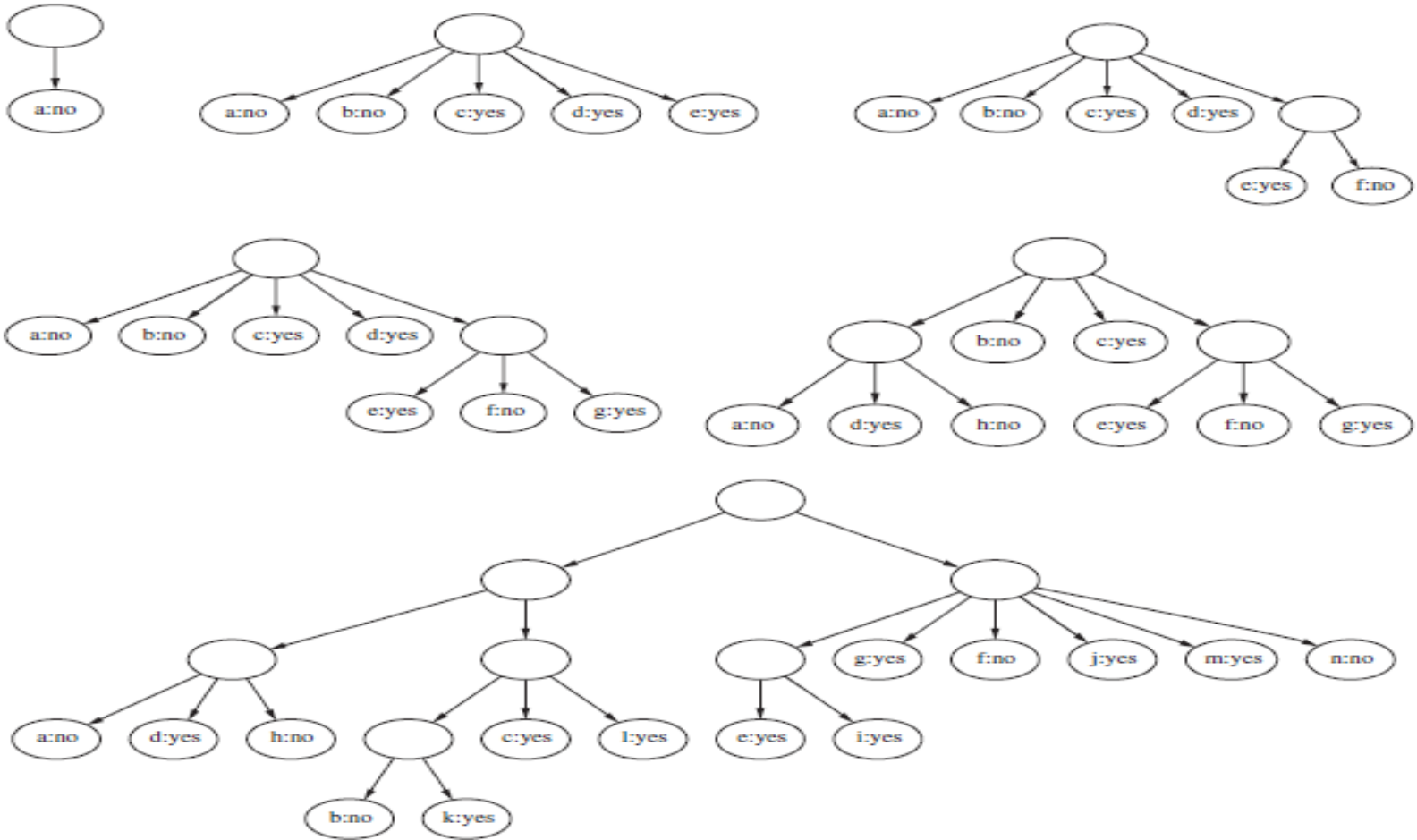


Clustering Incremental

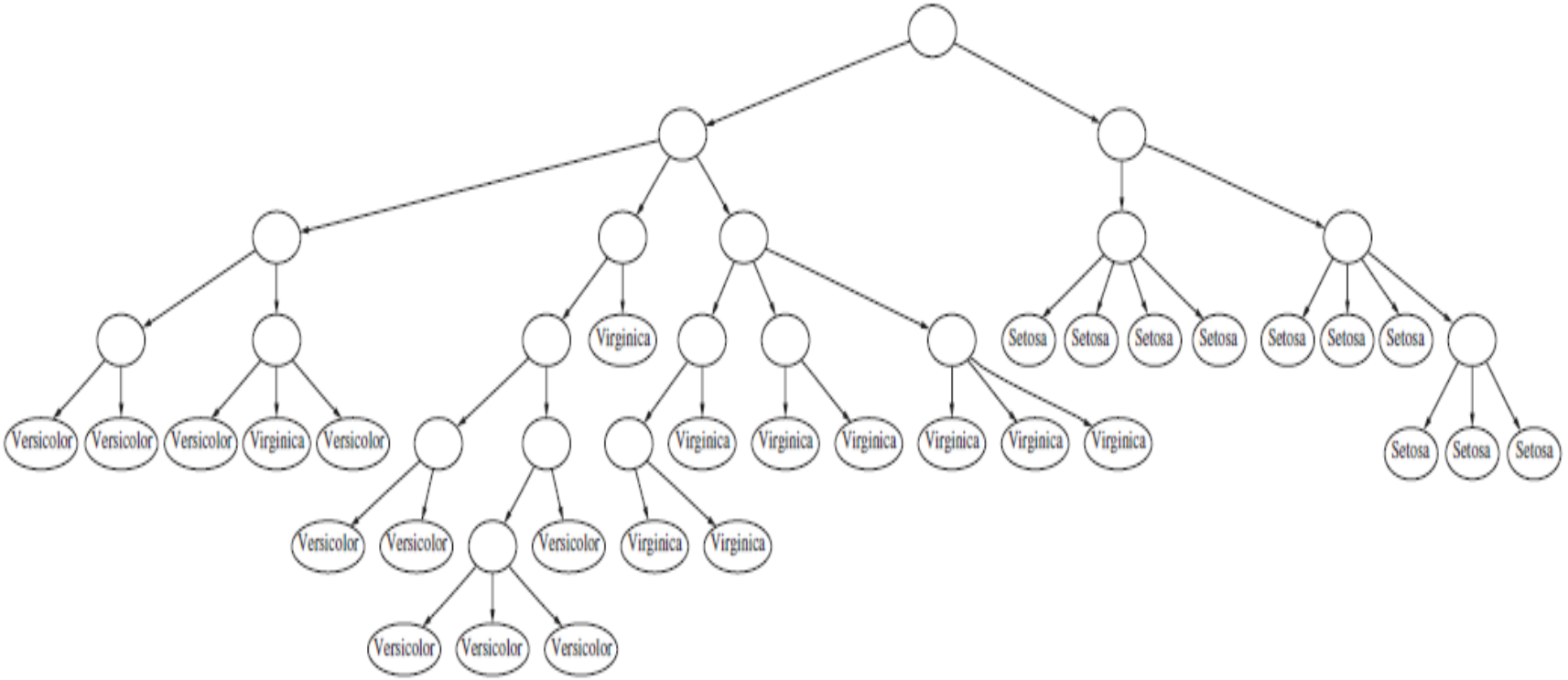
Datos de tiempo

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

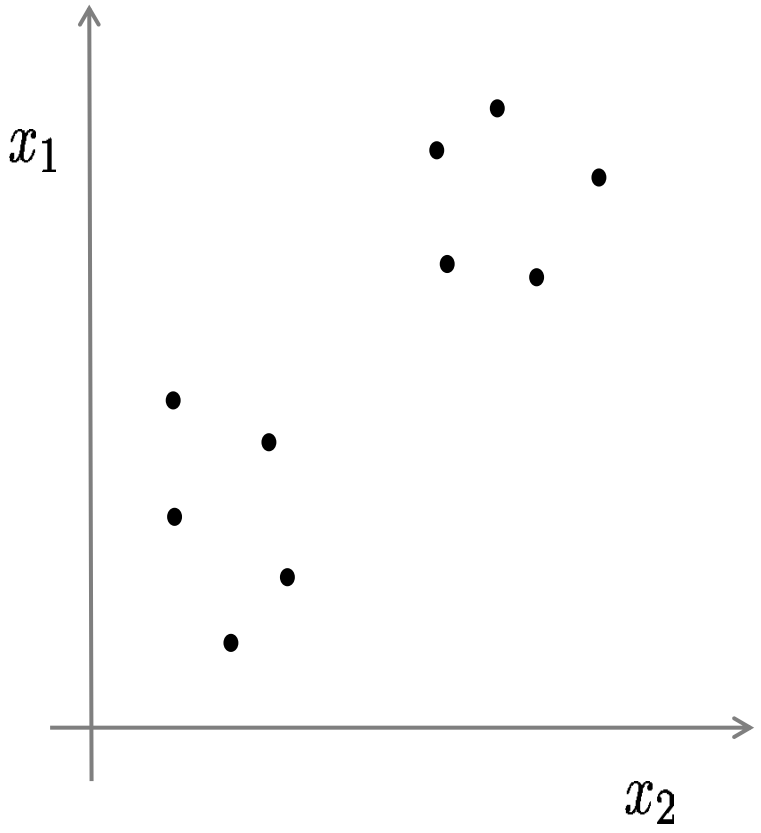
Clustering Incremental



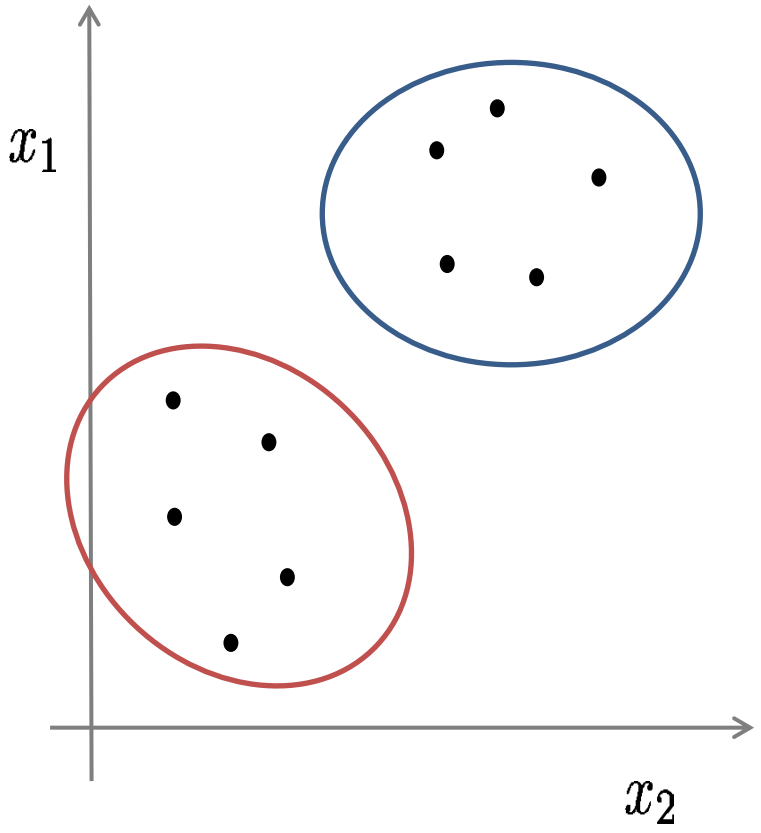
Clustering Incremental



Algoritmo no supervisado



Algoritmo no supervisado



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”

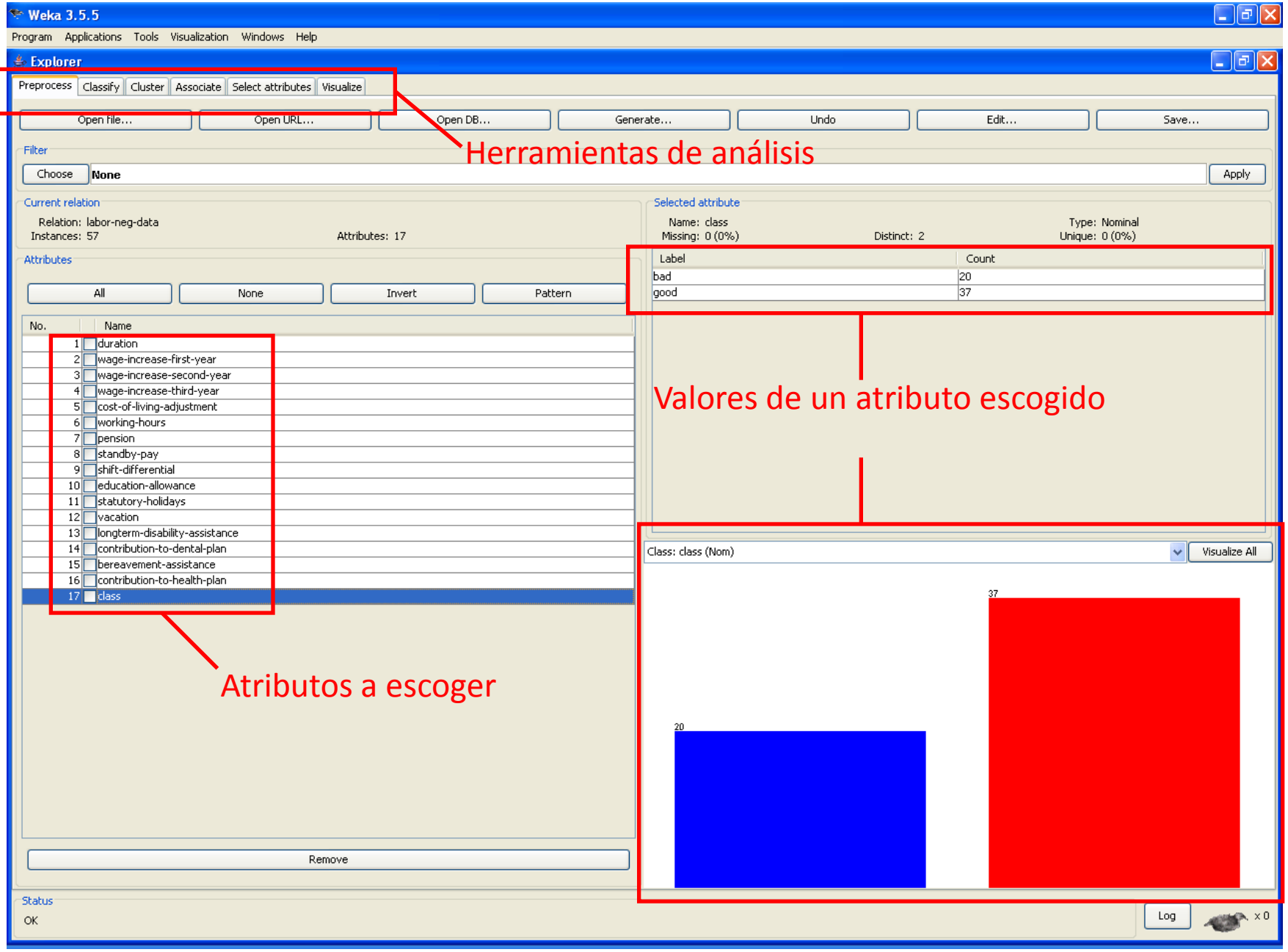

```
@relation heart-disease-simplified

@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

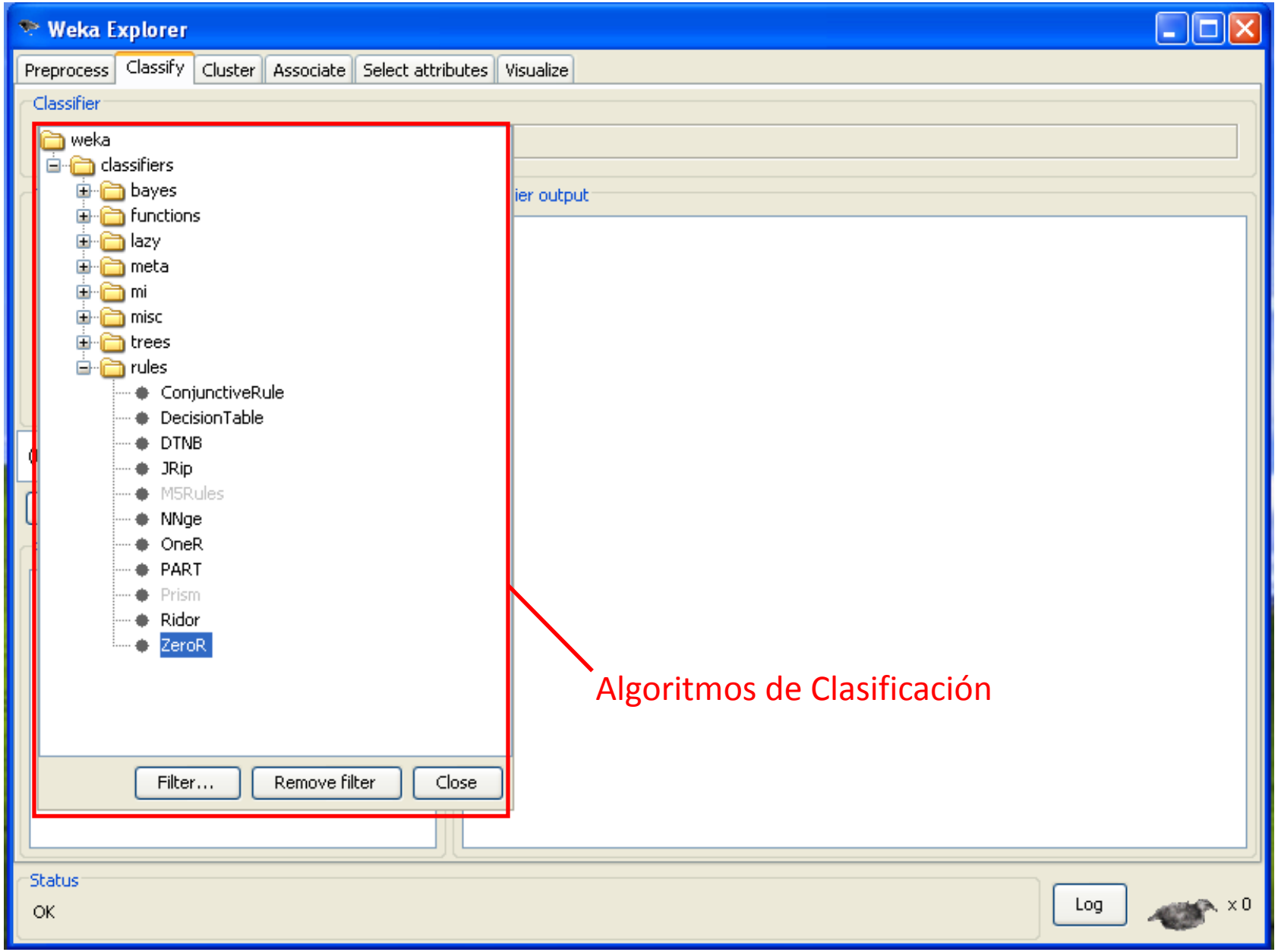
- Con interfaz gráfica de usuario

- 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
Algoritmos de Clasificación
Status
OK
Log
x 0

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

- Instances contains Attribute and Instance
  - como recuperar un valor de una instancia?

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

  - Como recuperar un atributo?

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

Clases para carga de datos

- como 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.SMO();
e.buildClassifier(trainingInstances);
```

Clases para la evaluación

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

- Como 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
```

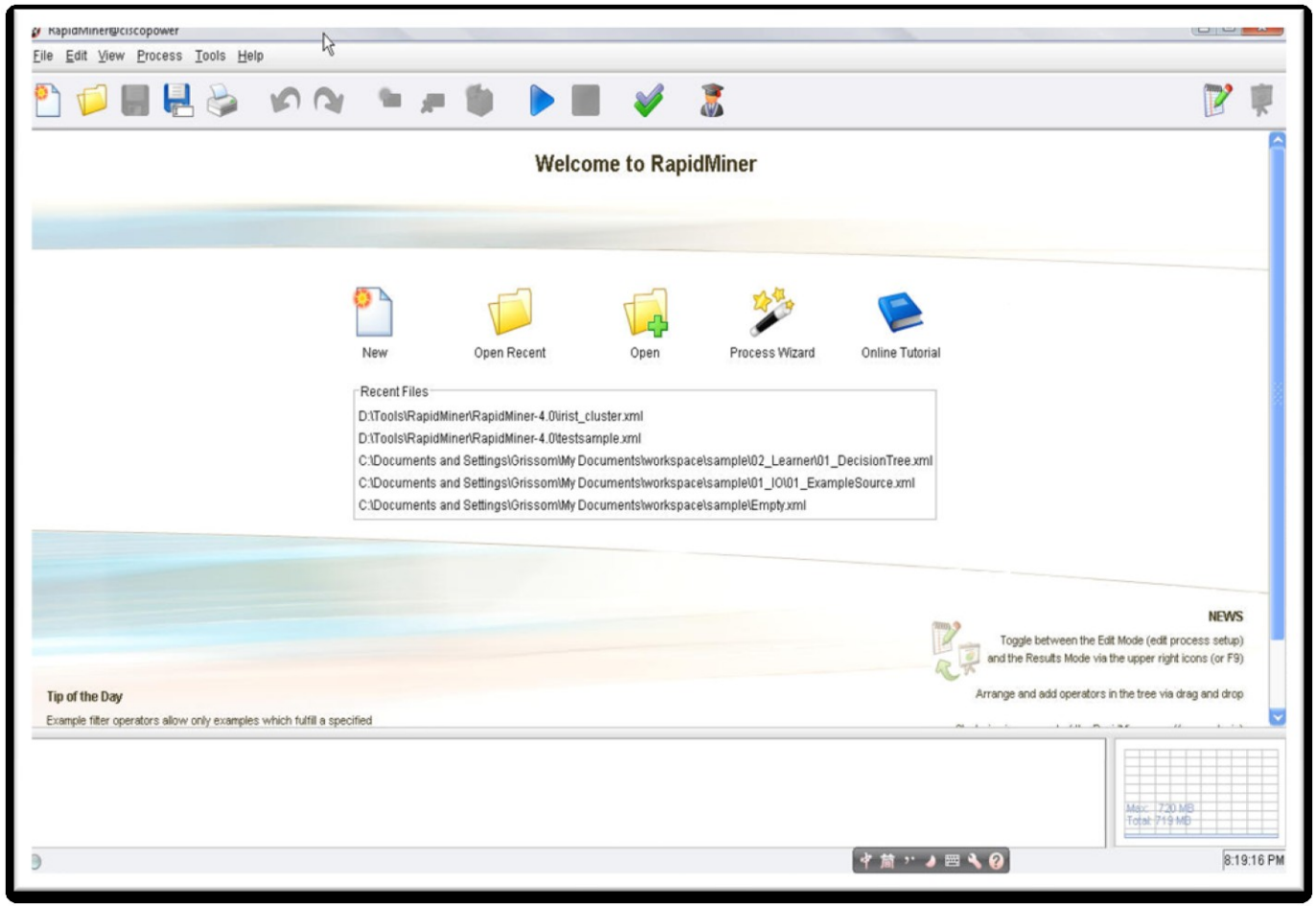
Tres pasos para el uso

- Asignar el archivo de datos primero
- Seleccionar funcionalidad
- Ejecutar Función utilizando rápido Minero

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

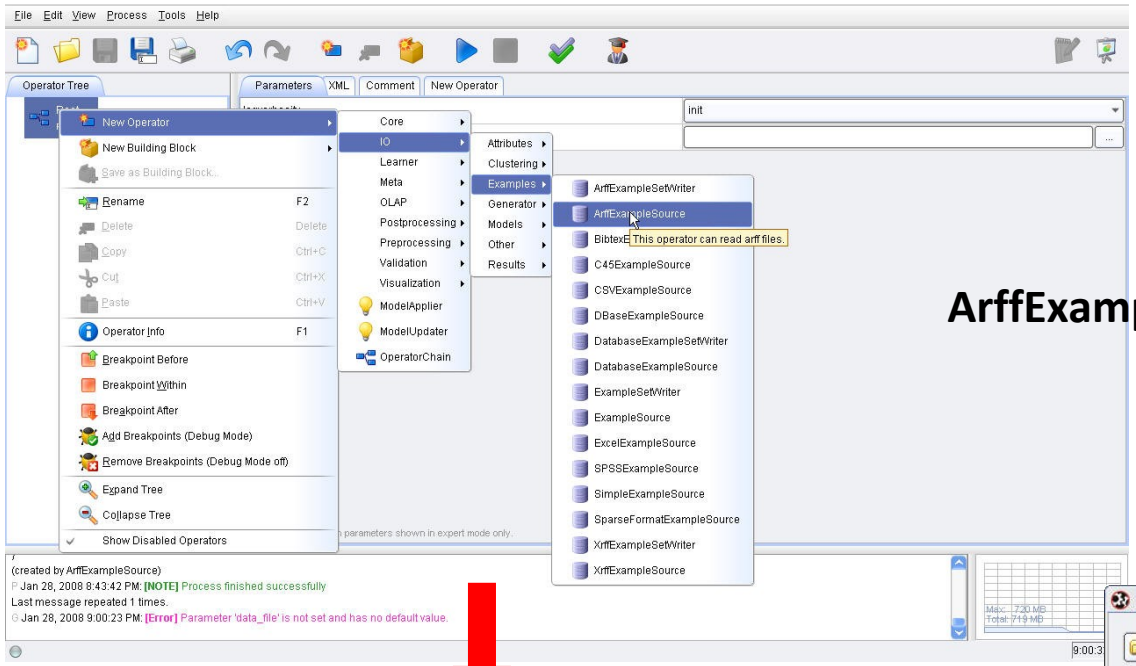
http://rapid-i.com/content/view/130/82/

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

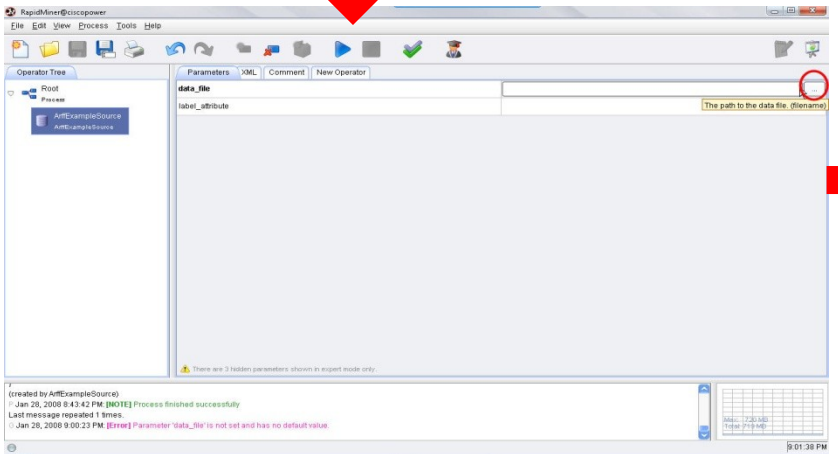


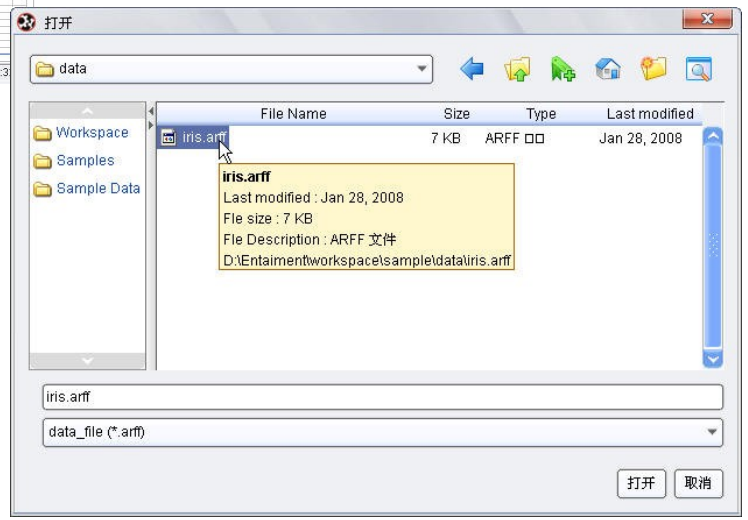
Establecer una fuente de datos para el proyecto

- Arff, excel , etc



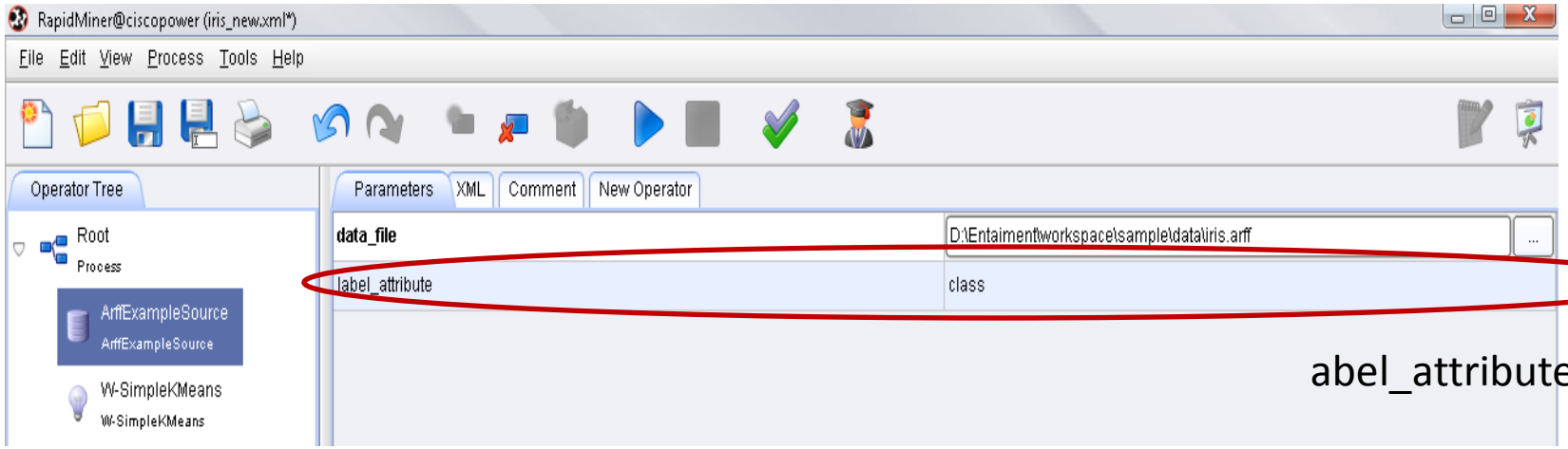
ArffExampleSource" menu





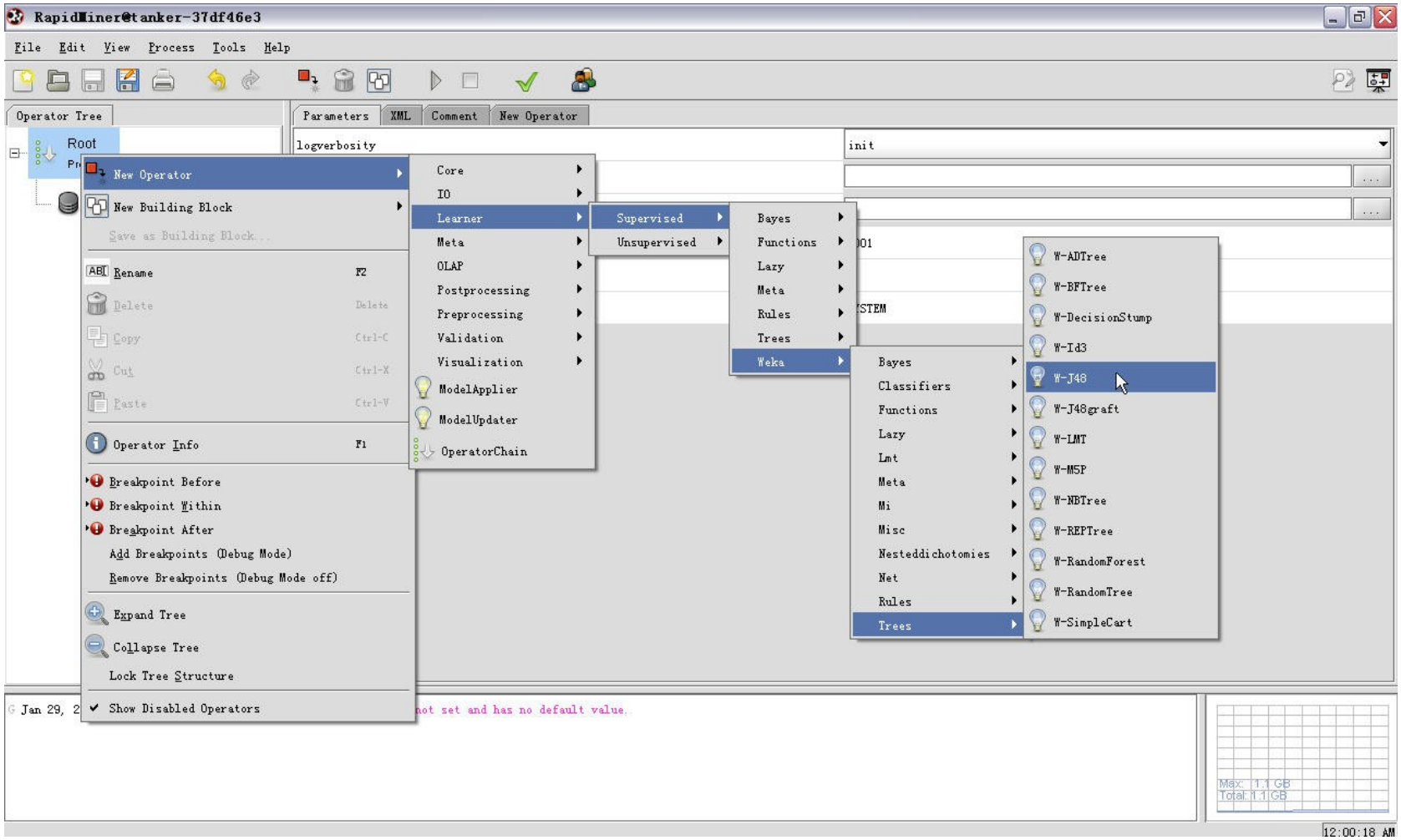
Escoger etiqueta atributo

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

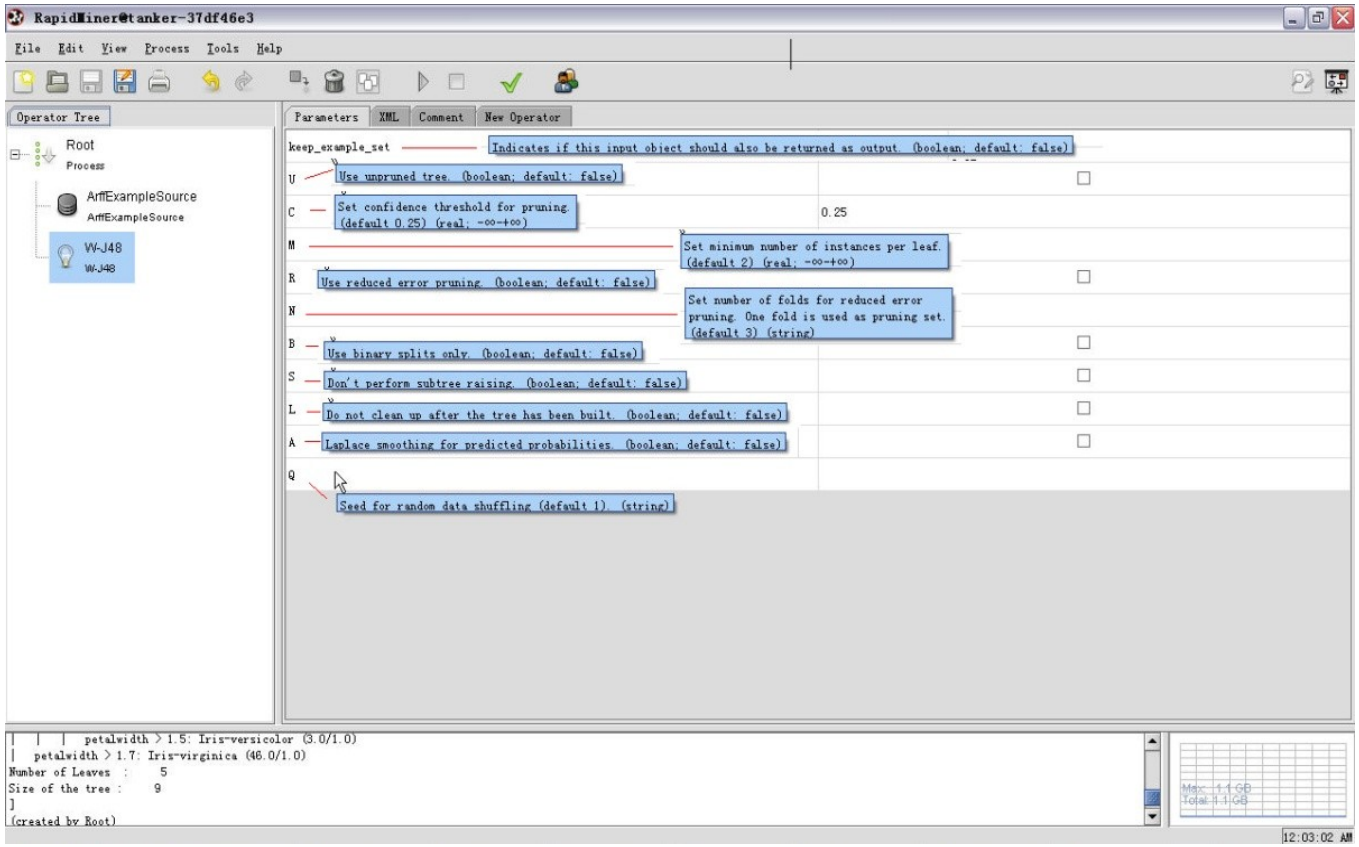


Seleccionar funcionalidad

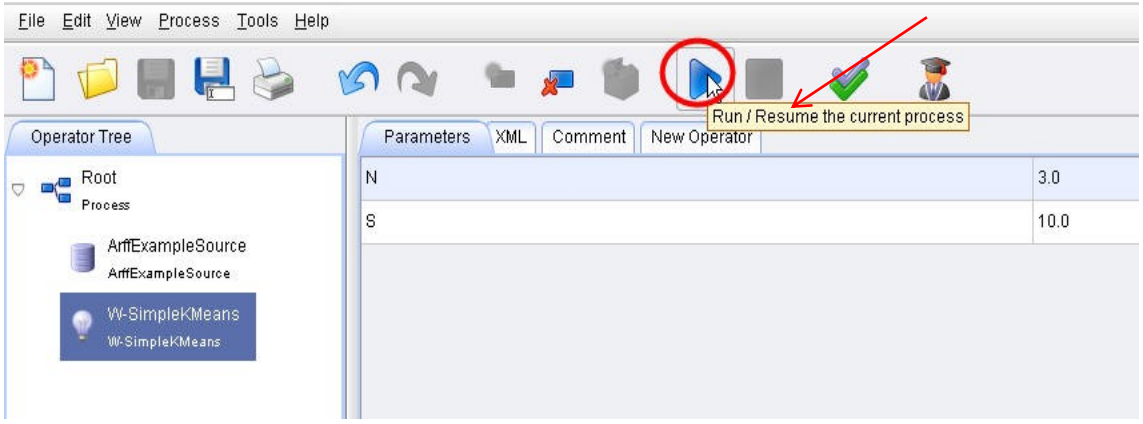
clustering, classification, decision tree, etc.



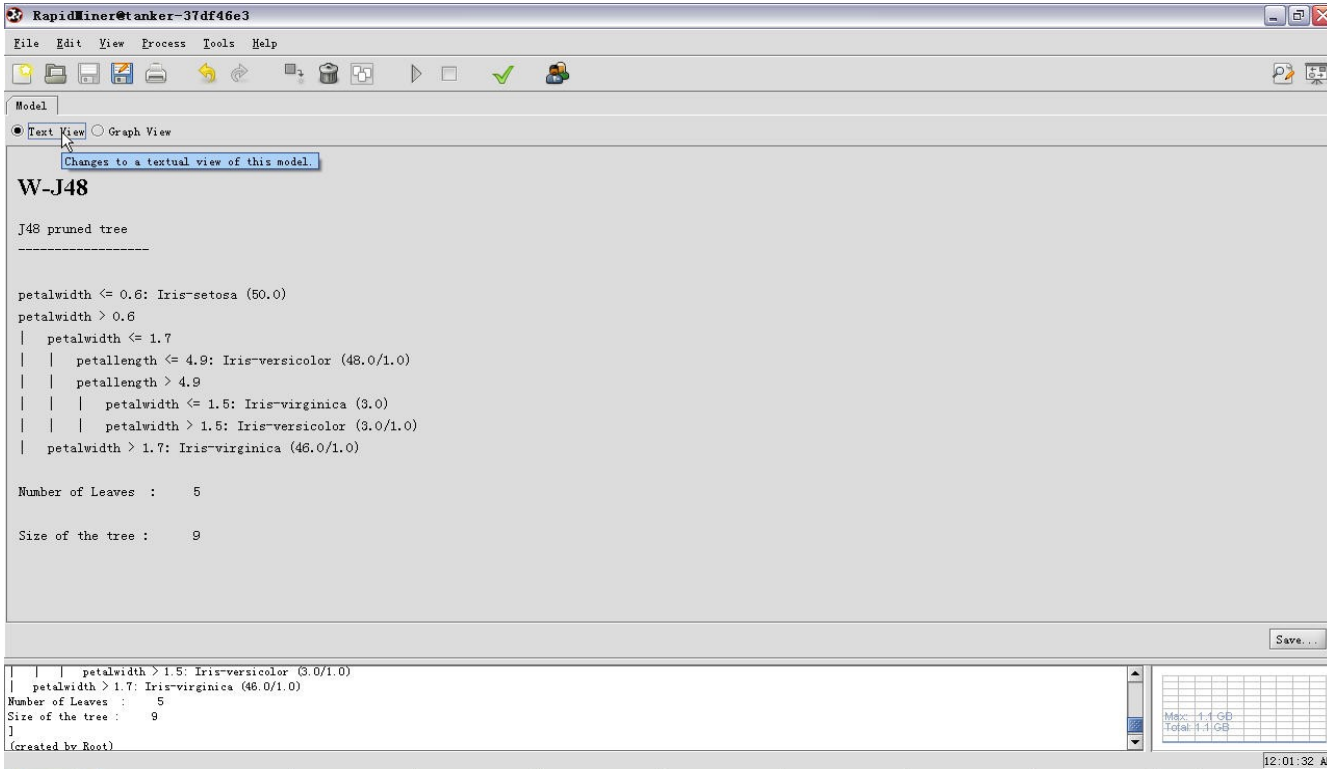
Escoger parámetros



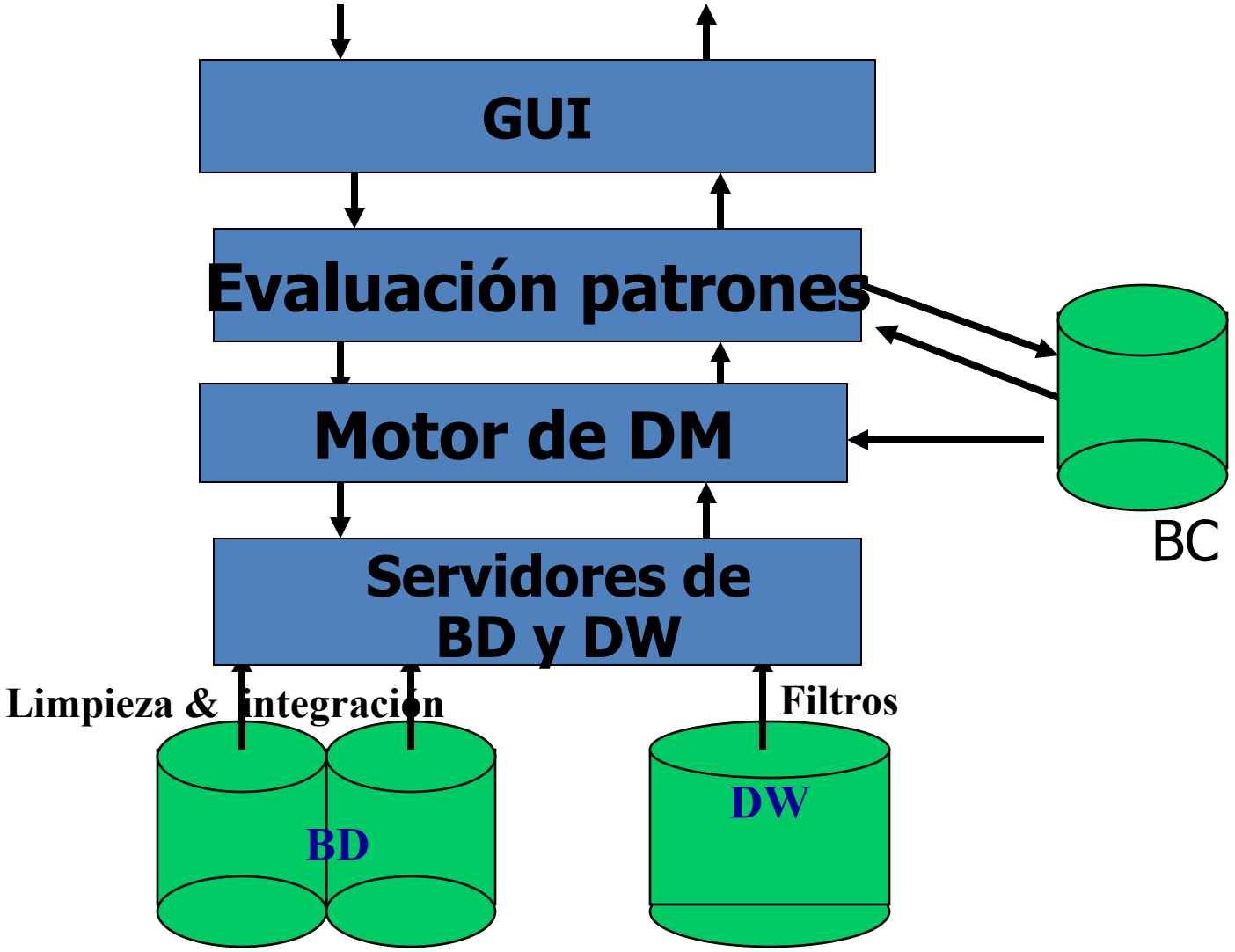
Ejecutar



resultados



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
...	...	...	...	...	...	...	...

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:

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

Minería de Datos e Inteligencia de Negocio

Gestión de personal de una empresa: ¿Qué clases 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:

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

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