



## Taller IV: Modelos de Segmentación

**Carlos Reveco**

**[creveco@dcc.uchile.cl](mailto:creveco@dcc.uchile.cl)**

**Cinthya Leonor Vergara Silva**

**[cvergarasilv@ing.uchile.cl](mailto:cvergarasilv@ing.uchile.cl)**

**Departamento de Ingeniería Industrial**

**Universidad de Chile**

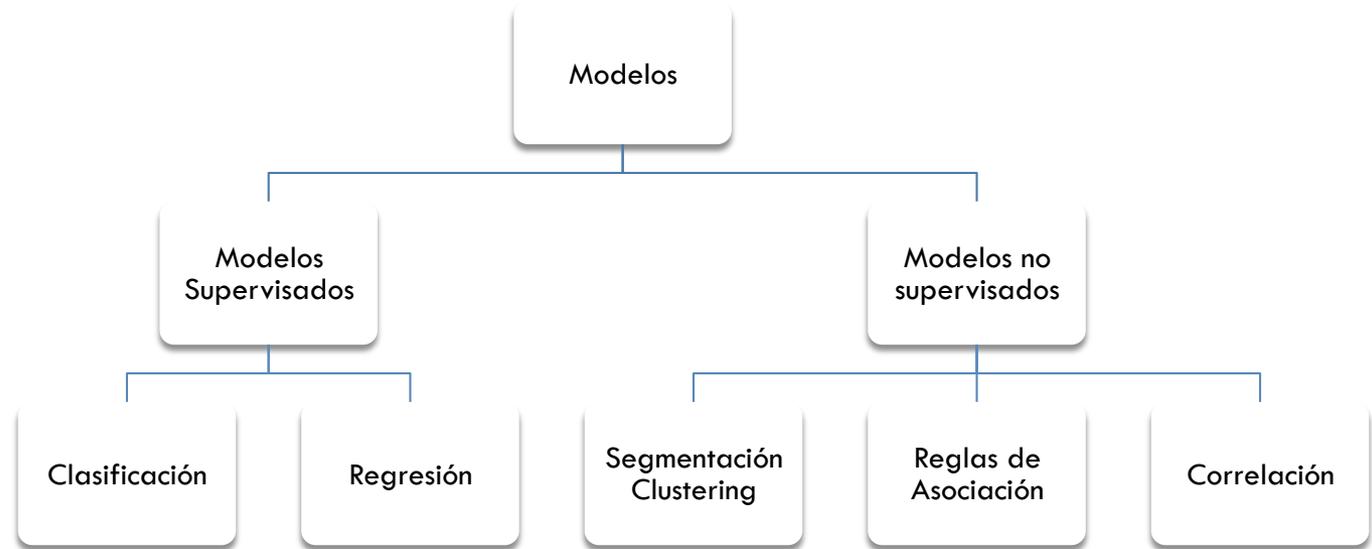
# Tipos de Aprendizaje

2

- Aprendizaje supervisado
  - Se utiliza el conocimiento a-priori del comportamiento de un conjunto de observaciones: Conjunto de Entrenamiento
  - Ejemplos: Análisis Discriminante, Regresión Logística, Árboles de Decisión, Redes Neuronales, Naive Bayes, Support Vector Machines,
  - etc...
- Aprendizaje no-supervisado
  - No se utiliza conocimiento a-priori del comportamiento de un conjunto de observaciones.
  - Ejemplos: k-Medias, Fuzzy C-Means, Kohonen Self Organizing Maps.

# Modelos Supervisados vs No supervisados

## Esquema Tipos de Modelos



# Modelos Supervisados vs No supervisados

## Modelos Supervisados

Tareas predictivas

Hay medidas de error o de capacidad de ajuste

Existe una variable objetivo a modelar

Ej: predecir la demanda por tickets de la prox semana; probabilidad de fuga de un cliente

## Modelos no supervisados

Tareas descriptivas o de resumen de un comportamiento global

Medidas de error más difusas relativas a la capacidad de interpretación de los datos

Existen muchas variables que describen distintos objetos

Ej: Asociación entre productos: Si hace upgrade entonces comprará duty free

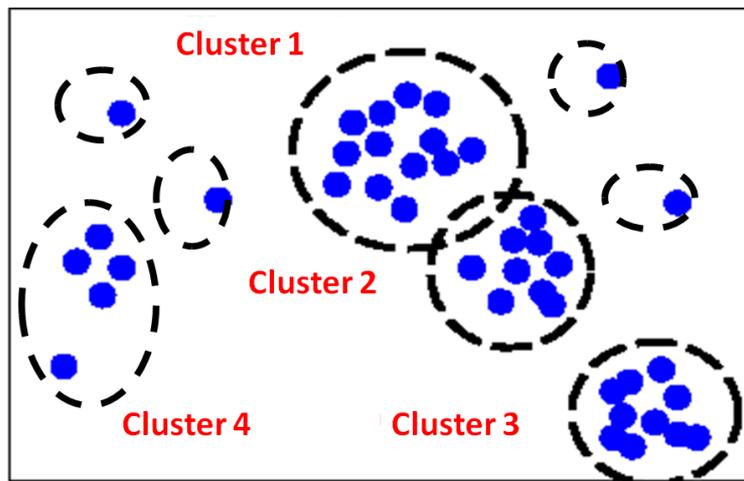
# Conceptos fundamentales:

5

- **Clúster:** Una colección de objetos ...
  - Similares entre aquellos objetos del mismo clúster
  - Distintos a los objetos de otros clúster.
- **Análisis de Clúster**
  - Consiste en agrupar un conjunto de datos en grupos homogéneos en base a los atributos definidos para determinar cuán “similares” son unos objetos de los otros.
- Un buen Clustering produce clúster con:
  - Alto nivel de “similitud” entre los objetos de la misma clase.
  - Bajo nivel de “similitud” entre las distintas clases.
  - La bondad de los clúster dependen directamente de la opinión de los usuarios.

# ¿Para qué hacer Clúster o Segmentación?

- El proceso de segmentación consiste en dividir el mercado en grupos más pequeños con características similares.
- El Objetivo de Segmentar nuestros clientes es entender mejor sus necesidades y hacerles una oferta que les brinde mayor satisfacción.
- La segmentación es un proceso creciente, en el sentido que las empresas trabajan cada día con mayor número de segmentos, ya que es la única forma de aumentar significativamente la satisfacción de los consumidores.



# Aplicaciones

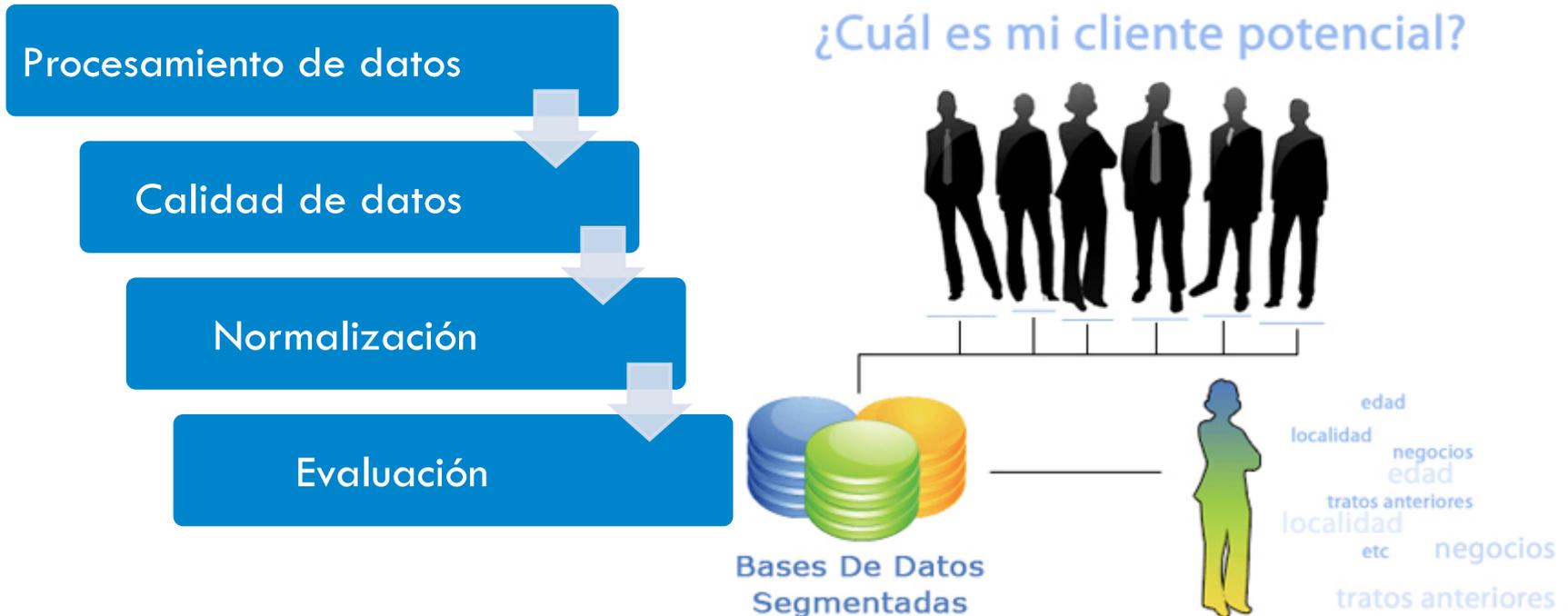
7

- Marketing: Segmentación de Mercado
- Reconocimiento y procesamiento de patrones en imágenes.
- Preprocesamiento de datos para la entrada de un modelo.
- WWW, Clasificación de documentos, Patrones de uso de sitios por usuarios, agrupamiento de sitios de interés.
- (e.g. <http://www.clusty.com>)
- Redes Sociales.

# Metodología de extracción de conocimiento KDD

8

KDD es el proceso no-trivial de identificar patrones previamente desconocidos, válidos, nuevos, potencialmente útiles y comprensibles dentro de los datos.



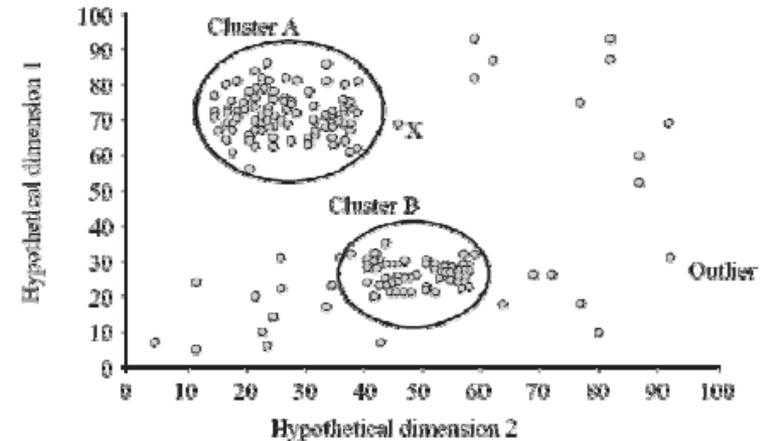
# Decisiones a tomar para realizar una segmentación

Elección de las variables de segmentación y caracterización.

Elección de la técnica de análisis.

Aplicación de la técnica de análisis para identificar segmentos, definiendo el número de segmentos.

Elegir una metodología que valide los segmentos que se han formado.



# Distancia

10

- Definición 1: Una **Distancia** sobre un conjunto  $\Omega$  es una función  $d:\Omega \times \Omega \rightarrow \mathfrak{R}$  tal que:
  - $d(i,j) \geq 0, \quad \forall i,j \in \Omega$
  - $d(i,i) = 0, \quad \forall i \in \Omega$
  - $d(i,j) = d(j,i), \quad \forall i,j \in \Omega$
- Una distancia es **métrica** si se cumple:
  - $d(i,j) \leq d(i,k) + d(k,j), \quad \forall i,j,k \in \Omega$
- Ejemplo: Distancia de Minkowsky, Mahalanobis, Coseno, etc...

# Similitud

11

- Una **similaridad** sobre un conjunto  $\Omega$  es una función  $d: \Omega \times \Omega \rightarrow \mathcal{R}$  tal que:
  - $0 \leq s(i,j) \leq 1 \quad \forall i,j \in \Omega$
  - $1 = s(i,i) \geq s(i,j) \quad \forall i \neq j \in \Omega$
  - $s(i,j) = s(j,i) \quad \forall i, j \in \Omega$
- Proposición: Si  $s(i,j)$  es **similaridad**, entonces las siguientes funciones son distancias:
  - $d(i,j) = \sqrt{1 - s(i,j)}$
  - $d(i,j) = 1 - s(i,j)$
  - $d(i,j) = \sqrt{1 - s^2(i,j)}$
- Una **vecindad** de un punto  $x$  es un conjunto  $V \subseteq \Omega$  tal que  $x \in V$ . A los puntos  $y \in V - \{x\}$  se llaman vecindad de un punto

# Tipos de Algoritmos

12

- **Métodos de Particionamiento:**
  - El número de clases se conoce inicialmente, por lo que el conjunto de datos se divide en nuevos conjuntos de datos similares.
  - A partir de un particionamiento inicial (aleatorio) se reasignan los puntos hasta:
    - Datos al interior de clúster -> pequeña diferencia.
    - Datos entre clúster distintos -> Alta diferencia.
- **Métodos Jerárquicos**
  - El número de Clúster es determinado a partir del método.
  - Top – Down: Se inicia con todos los elementos en un clúster único y se va refinando (Top-Down).
  - Bottom – Up: Se inicia con un par de elementos que forman un clúster y se van agrupando con el resto.

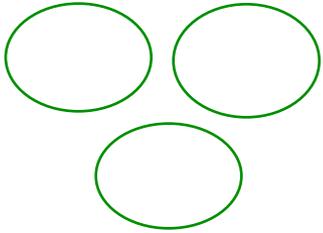
# Tipos de Algoritmos

13

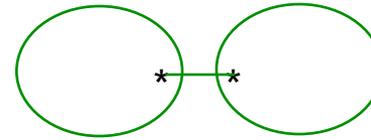
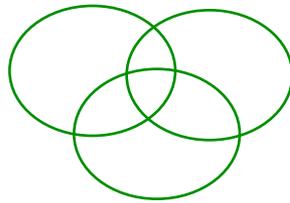
- **Métodos Basados en Densidad:**
  - Elementos vecinos se agrupan en medidas de densidad locales.
  - O cumplir propiedades fijadas a priori a una distancia determinada entre los elementos.
- **Métodos Basados en Modelos**
  - Hipótesis para cada clúster: Distribución o modelo determinado.
  - Objetivo: Determinar el mejor ajuste del modelo.
- **Métodos Heurísticos**
  - Definición de algún criterio externo al modelo para que llegue a una solución aceptable.
  - Ej. K-Medias: pertenencia absoluta, Fuzzy C-Means: grados de pertenencia.

# Métodos de segmentación

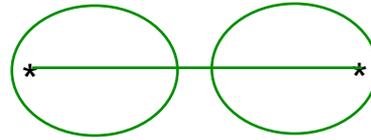
Excluyentes



No excluyentes

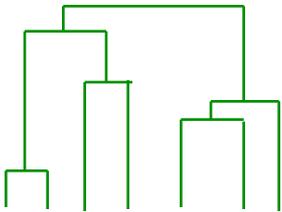


Enlace simple:  
*Mínima distancia*

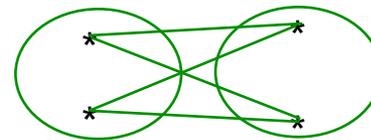
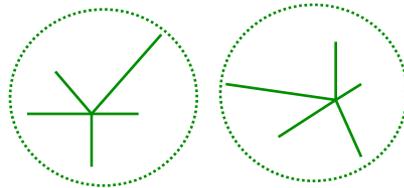


Enlace completo:  
*Máxima distancia*

Jerárquico

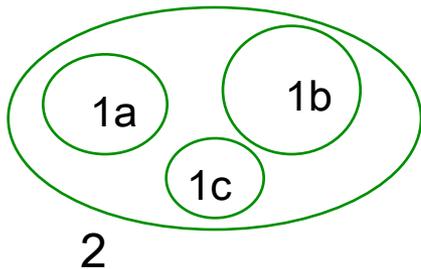


Particionamiento

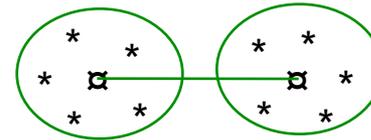
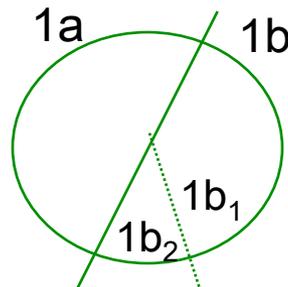


Enlace promedio:  
*Distancia promedio*

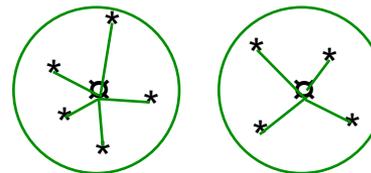
Aglomerativo



Divisivo



Método del Centroide:  
*Distancia entre centros*

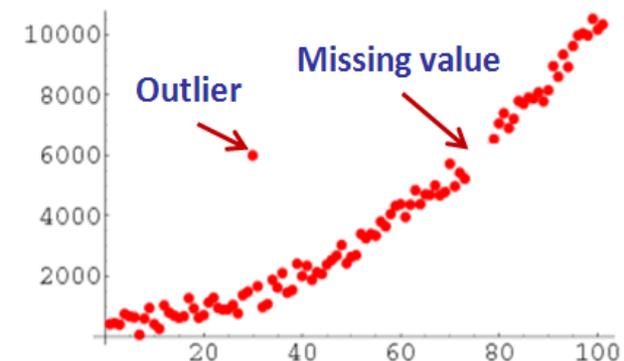


Método Wards:  
*Minimización de within-cluster variance*

# CLUSTERING : Preprocesamiento de Datos

15

- Es relevante escalar los valores de las variables en un mismo rango:
  - ▣ Si no se normaliza o estandariza, algunos atributos pueden tomar mayor relevancia que otros en el modelo, afectando los resultados.
- Es importante utilizar aquellos atributos que son relevantes para disminuir la complejidad computacional:
  - ▣ Muchos atributos pueden no aportar información relevante y afectar los tiempos de cálculo de los segmentos.



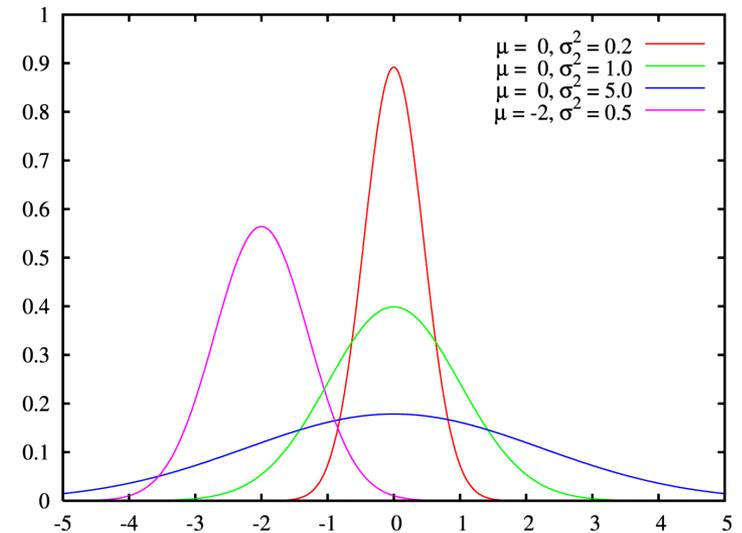
# Normalización

16

- El propósito de la normalización de variables, previo al proceso de agrupamiento, es que las diferentes escalas u órdenes de magnitud en que se pueden encontrar las variables de segmentación no afecten o distorsionen las distancias reales existentes entre los objetos
- Estandarización:  $Z_i = (x_i - \mu) / \sigma$ 
  - $\sim N(0,1)$
- Normalización Rango:  $z_i = (x_i - \min) / (\max - \min)$ 
  - Rango entre 0 y 1

Donde  $X$  son los valores sin estandarizar y  $Z$  los valores estandarizados.

El uso de la normalización permite conocer la media ( $\mu=0$ ) y la desviación estándar ( $\delta=1$ ) de los datos .



# Métodos de Particionamiento

17

- El problema de particionamiento.
  - Problema de optimización, donde dado un  $K$ , se desea encontrar los  $K$  clúster tal que se maximice (o minimice) un criterio de partición.
  - Problema NP-hard.
- Métodos heurísticos:
  - K-Medias: pertenencia absoluta.
  - Fuzzy C-Means: grados de pertenencia.

# Métodos de Clustering Jerárquicos

18

- Clustering Jerárquico Aglomerativo (Bottom Up)
  - Se inicia con cada uno de los datos presentes en la base de datos como un clúster, y se van agrupando iterativamente hasta tenerlos todos incluidos en una jerarquía.
  - Problemas de implementación: Número muy grande de datos iniciales.
- Clustering Jerárquico Divisivo (Top Down)
  - Toda la base de datos se inicia como un clúster. Posteriormente se van definiendo los clúster iterativamente dentro del clúster inicial de manera jerárquica.
  - Ambos tipos de clustering pueden ser utilizados para determinar el número de clústeres presentes en una base de datos.

# Algoritmo Jerárquico Aglomerativo

19

- Cluster\_Aglomerativo (D)
  1. Se inicializan todos los puntos en D como un clúster independiente
  2. Determinar las distancias entre los clúster.
  3. *Mientras* no hay más clúster que agrupar *Hacer*
    1. Combinar 2 clúster que presenten la mínima distancia entre ellos.
    2. Actualizar las distancias entre los clúster.
- Terminar *Mientras*

# Tipos de Enlace

20

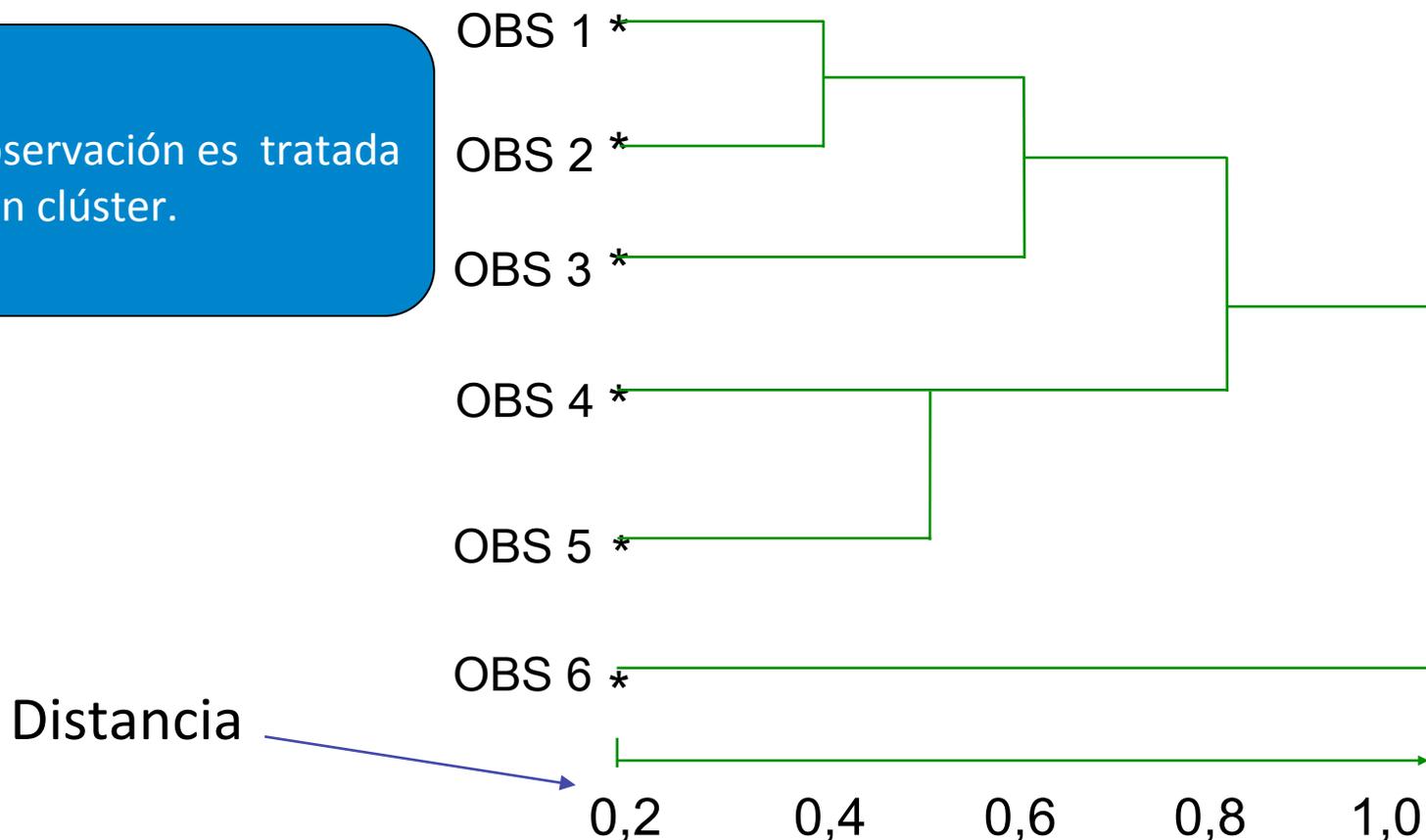
- **Enlace promedio:** Se calcula la distancia media entre los miembros del clúster y las observaciones en consideración a ser incluidas en el clúster.
- **Enlace Singular:** Se calcula la distancia entre todos los miembros del clúster y las observaciones en consideración a incluir en el clúster, seleccionando la menor.
- **Enlace completo:** Se calcula la distancia entre todos los miembros del clúster y las observaciones en consideración a incluir en el clúster. Donde la distancia máxima entre dos clases (objetos) sea la menor.

# Dendograma (1)

Para representar la estructura jerárquica de la formación de los conglomerados se utiliza el dendograma, un gráfico que tiene forma de árbol invertido.

## Paso 0:

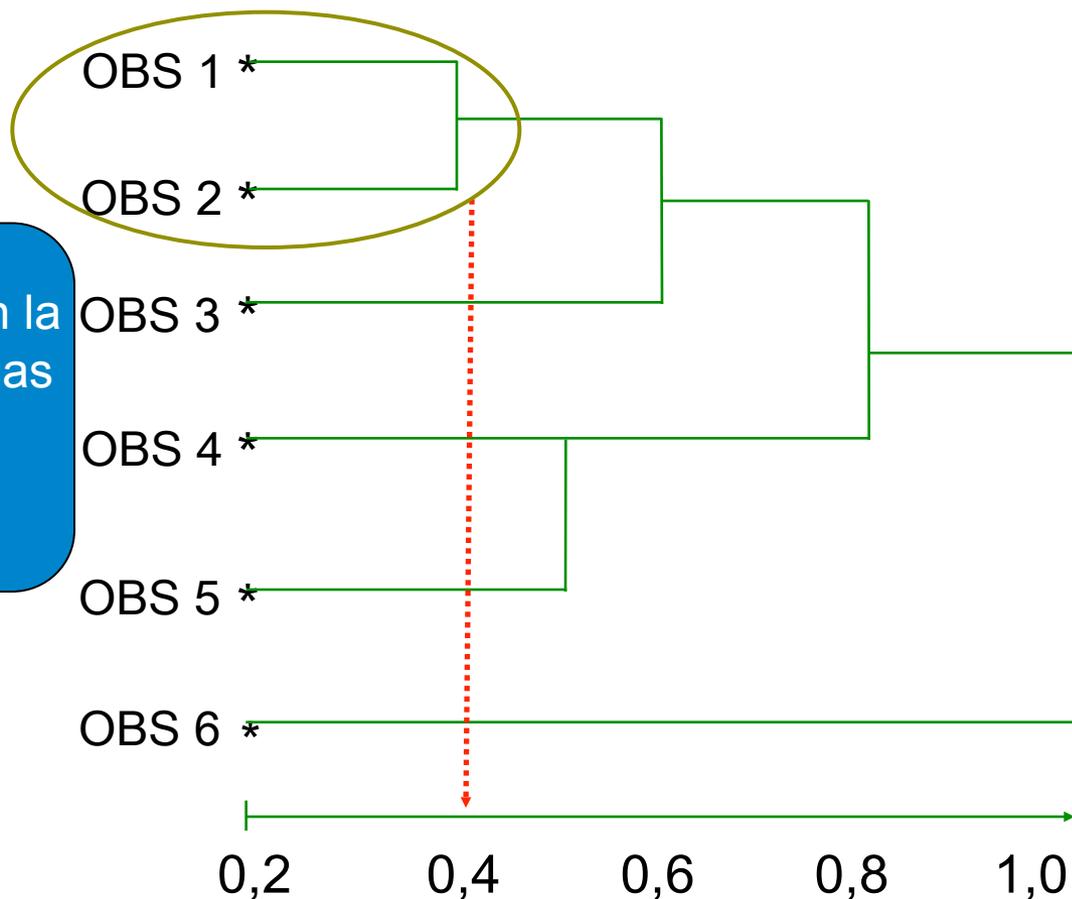
Cada observación es tratada como un clúster.



# Dendrograma (2)

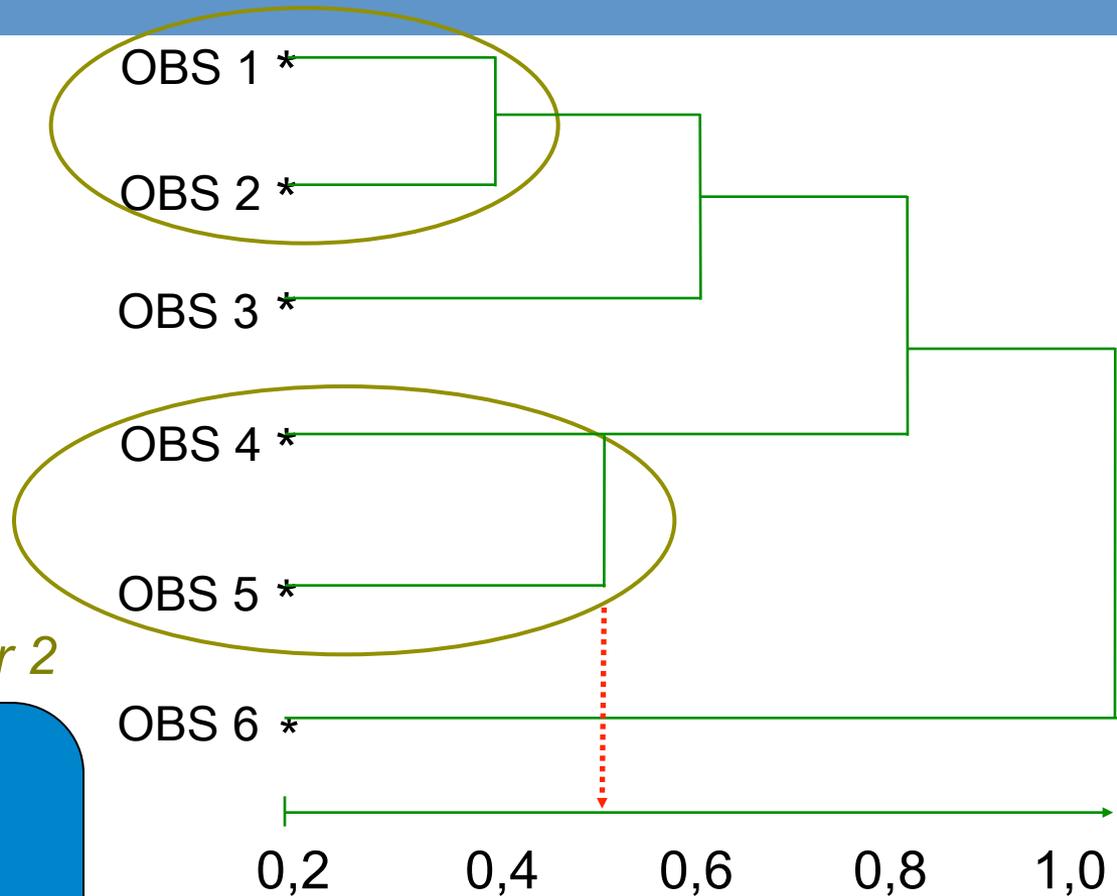
*Cluster 1*

Paso 1:  
Las 2 observaciones con la  
menor distancia entre ellas  
se unen en un clúster



# Dendograma (3)

*Cluster 1*



*Cluster 2*

Paso 2:  
Otras 2 observaciones  
con la menor distancia  
entre ellas (con respecto a  
las restantes) forman el  
cluster 2

# Dendrograma (4)

*Cluster 1*

OBS 1 \*

OBS 2 \*

OBS 3 \*

OBS 4 \*

OBS 5 \*

*Cluster 2*

OBS 6 \*

0,2

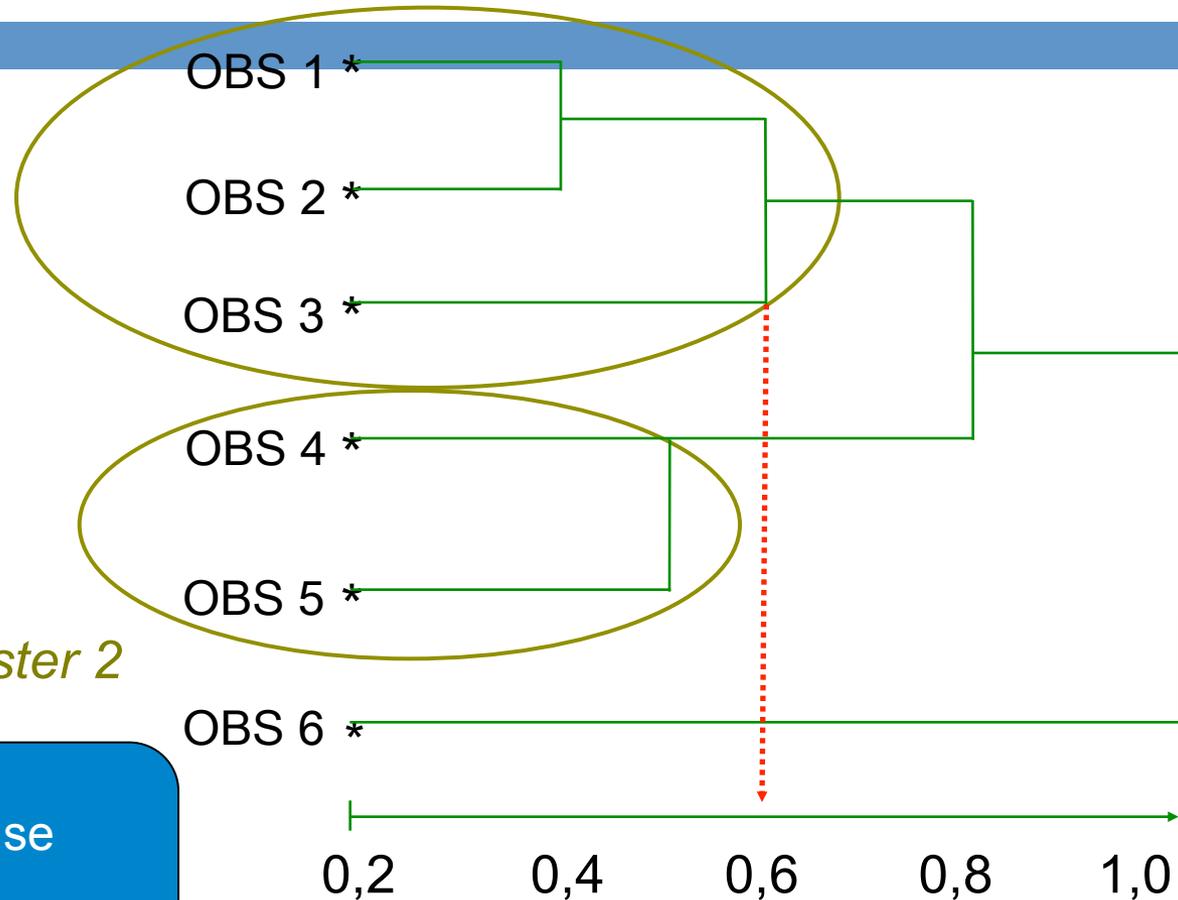
0,4

0,6

0,8

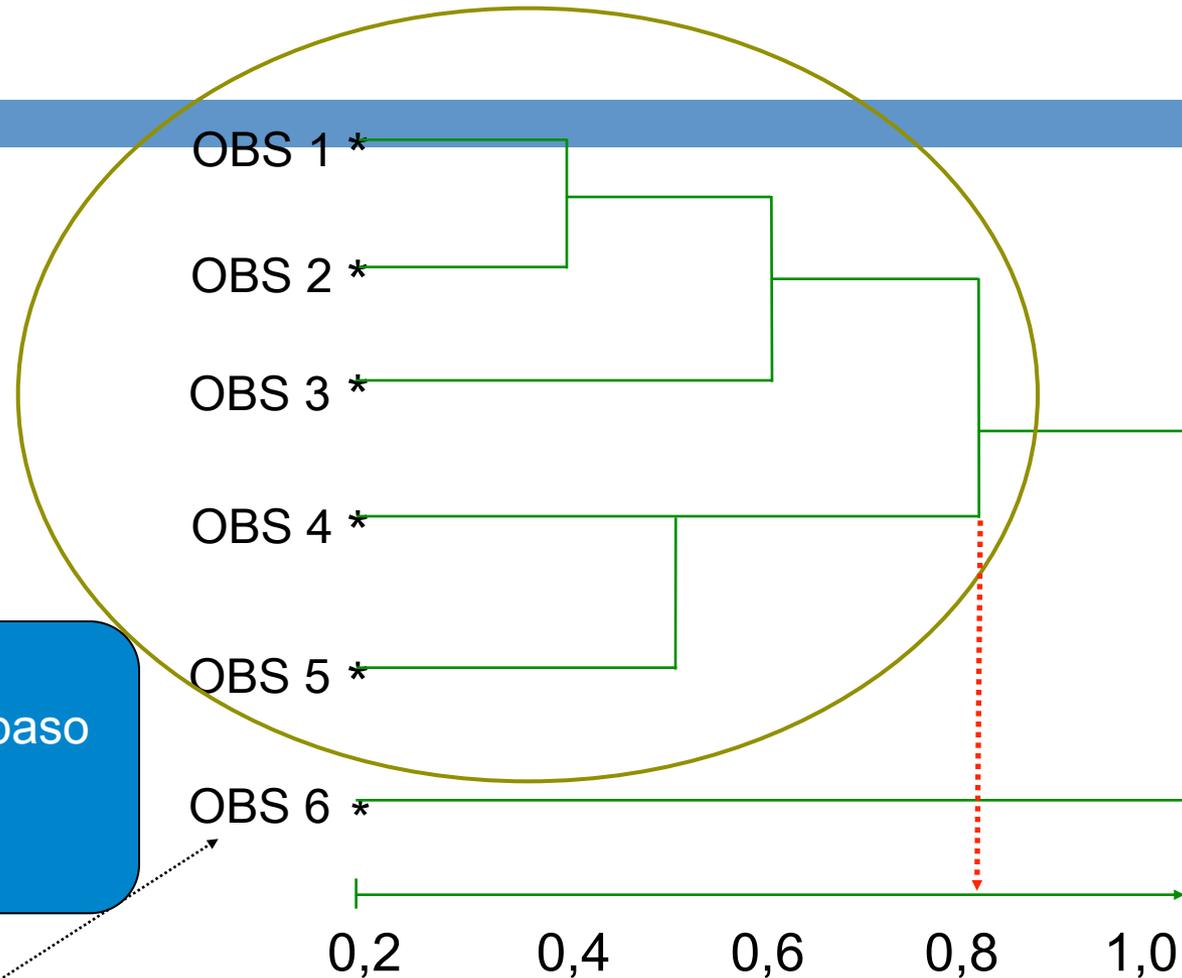
1,0

Paso 3:  
La observación 3 se  
une al cluster 1



# Dendrograma (5)

*“Supercluster”*



Paso 4:  
Clusters 1 y 2, del paso  
3, se unen en un  
Supercluster

Una sola observación queda fuera de  
nuestra agrupación (Outlier)

# K-Means

26

KMedias

- Dado un  $K$ , se implementa fundamentalmente en los siguientes pasos:
  1. Inicializar centroides (aleatorios).
  2. Particionar los objetos en  $K$  subconjuntos no vacíos según distancia.
  3. Calcular los centroides de los clúster de las respectivas particiones. El centroide representa el centro de cada clúster.
  4. Asignar a cada objeto al clúster cuyo centroide sea el más cercano.
  5. Volver al paso 2, y detenerse cuando ya no ocurran actualizaciones.

# Algoritmo K-Means (McQueen. 1967)

El objetivo es minimizar la disimilitud de los elementos dentro de cada cluster y maximizar la disimilaridad de los elementos que caen en diferentes clusters.

**Input:** Un conjunto de datos  $S$  y  $k$  número de clusters a formar;

**Output:**  $L$  una lista de los clusters en que caen las observaciones de  $S$ .

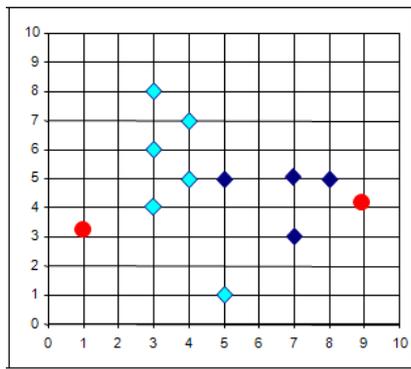
1. Seleccionar los centroides iniciales de los  $K$  clusters:  $c_1, c_2, \dots, c_K$ .
2. Asignar cada observación  $x_i$  de  $S$  al cluster  $C(i)$  cuyo centroide  $c(i)$  está mas cerca de  $x_i$ . Es decir,  $C(i) = \operatorname{argmin}_{1 \leq k \leq K} \|x_i - c_k\|$
3. Para cada uno de los clusters se recalcula su centroide basado en los elementos que están contenidos en el cluster y minimizando la suma de cuadrados dentro del cluster. Es decir,

$$WSS = \sum_{k=1}^K \sum_{C(i)=k} \|x_i - c_k\|^2$$

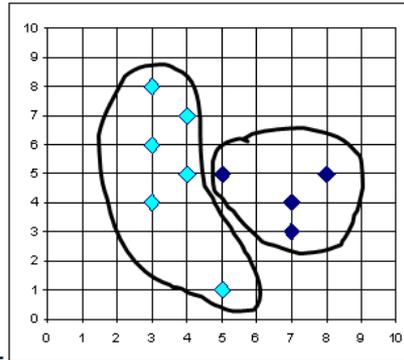
Ir al paso 2 hasta que se consiga convergencia.

# Método K- means

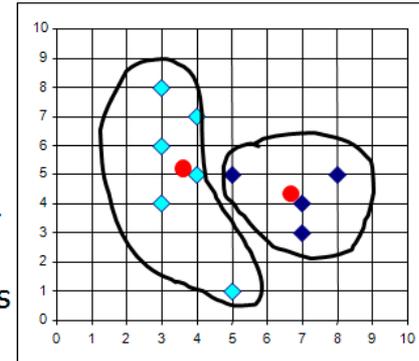
28



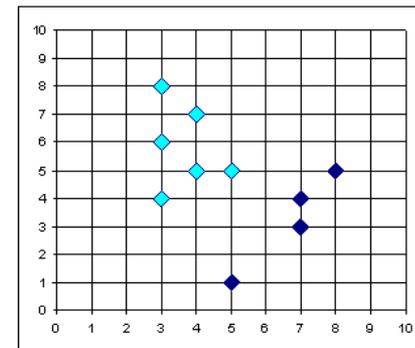
Asignar cada instancia al cluster mas cercano.



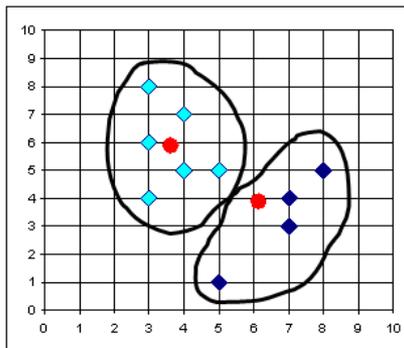
Actualizar los centroides



reasignar



Actualizar los centroides



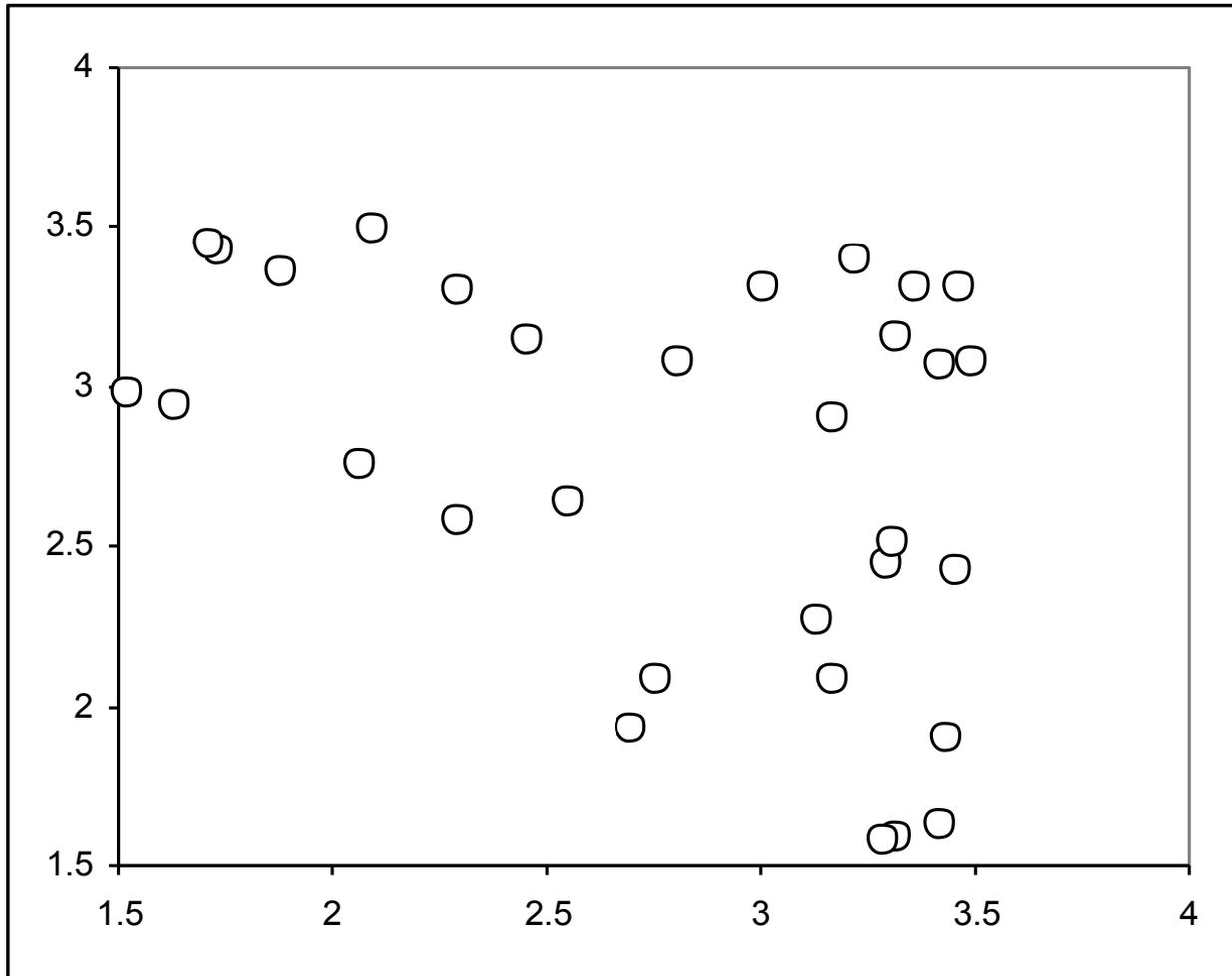
reasignar

K=2

Escoger arbitrariamente K instances como los centroides

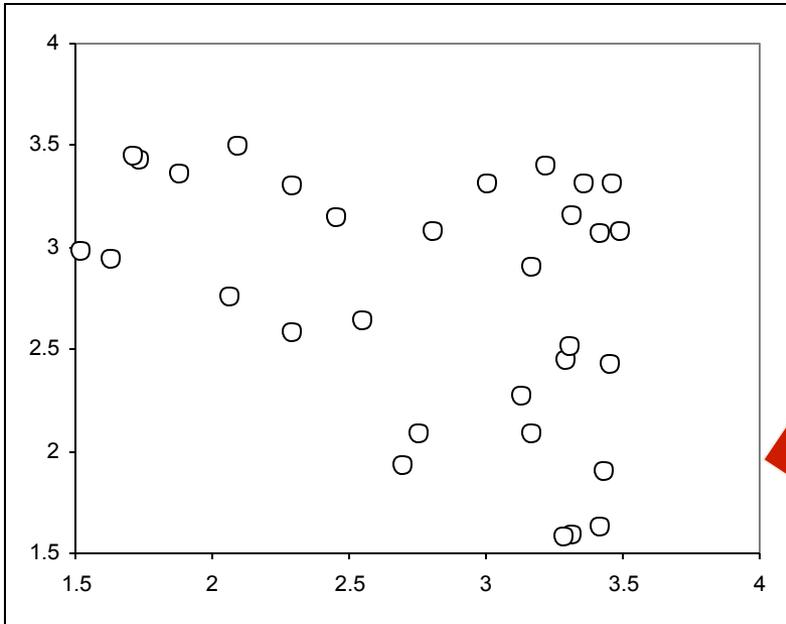
# Ejemplo de análisis de clúster (1)

Datos:

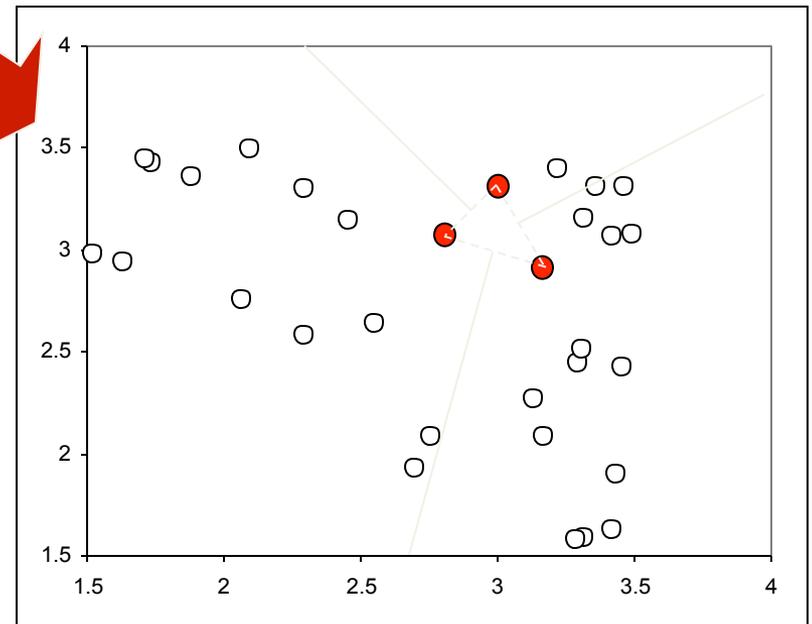


# Ejemplo de análisis de clúster (2)

## Iteración 1:

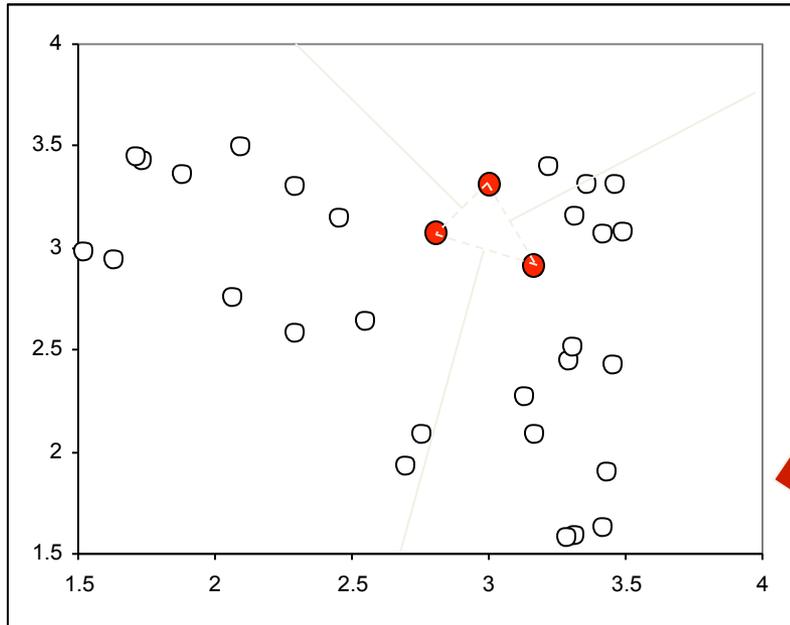


- Elegir los centros de los tres clusters aleatoriamente
- Localizar cada punto en su centro de cluster más cercano

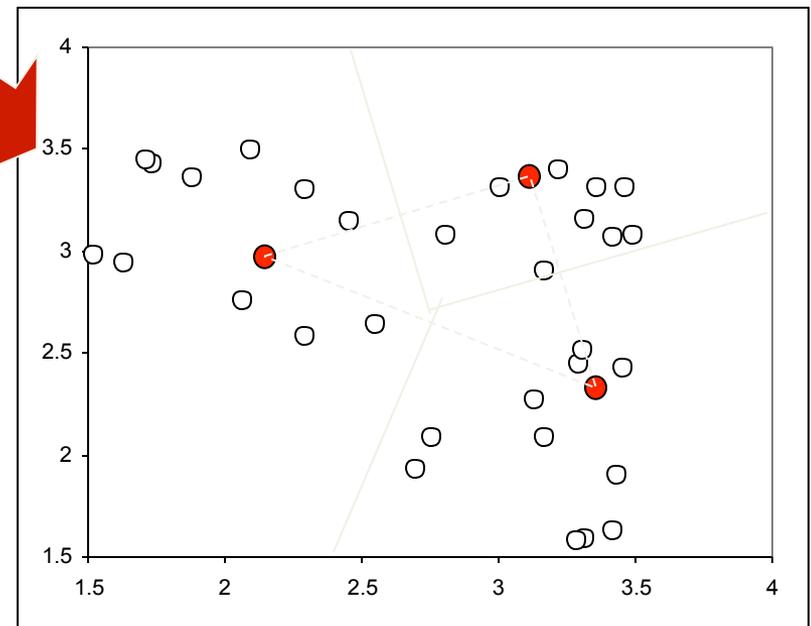


# Ejemplo de análisis de clúster (3)

## Iteración 2:

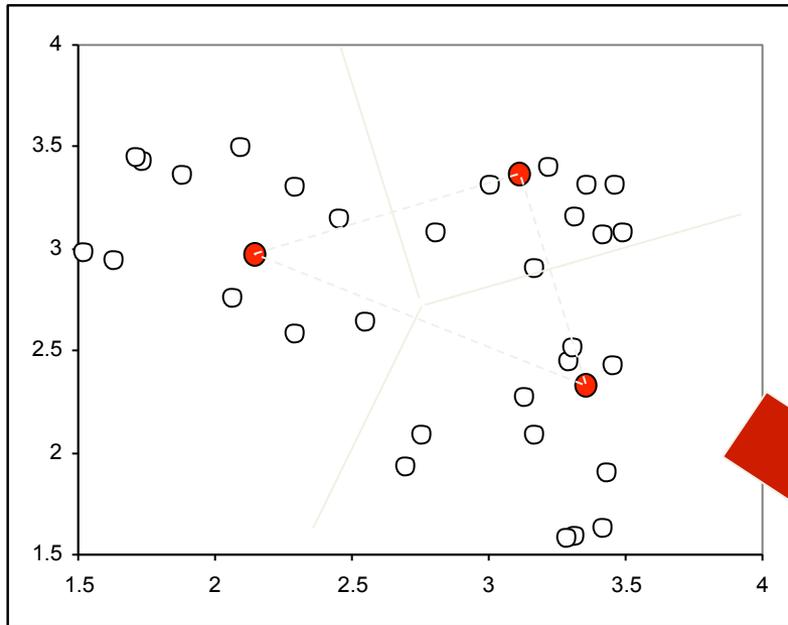


- Calcular nuevamente los centros de los clusters desde los centroides escogidos en la iteración 1.
- Localizar cada punto en el centroide que está más cerca a él.

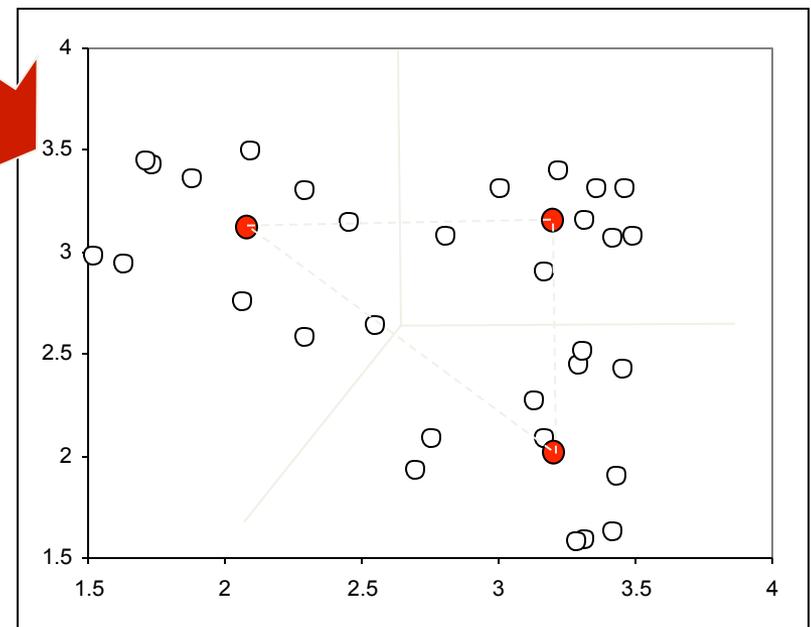


# Ejemplo de análisis de clúster (4)

## Iteración 3:



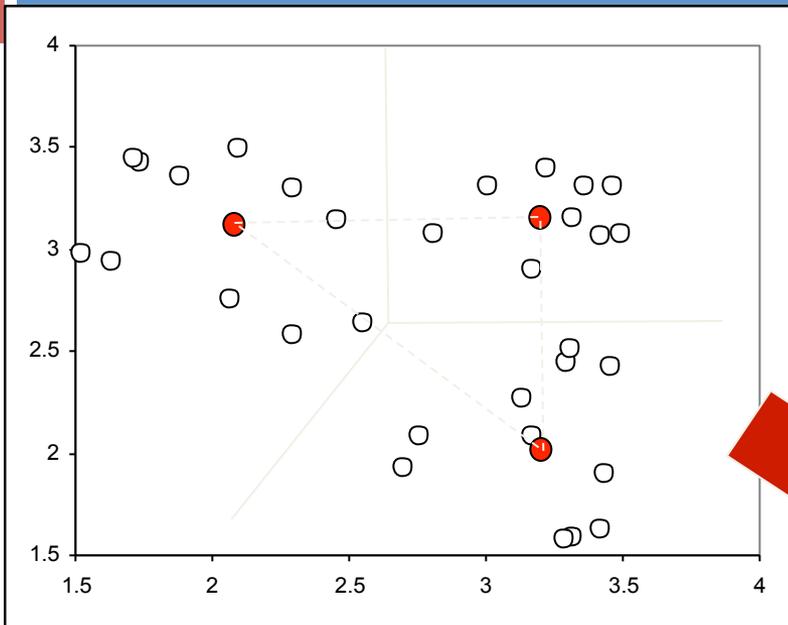
- Recalcular los centros como los centroides encontrados en la iteración 2.
- Localizar cada punto en el centroide que está más cerca a él.



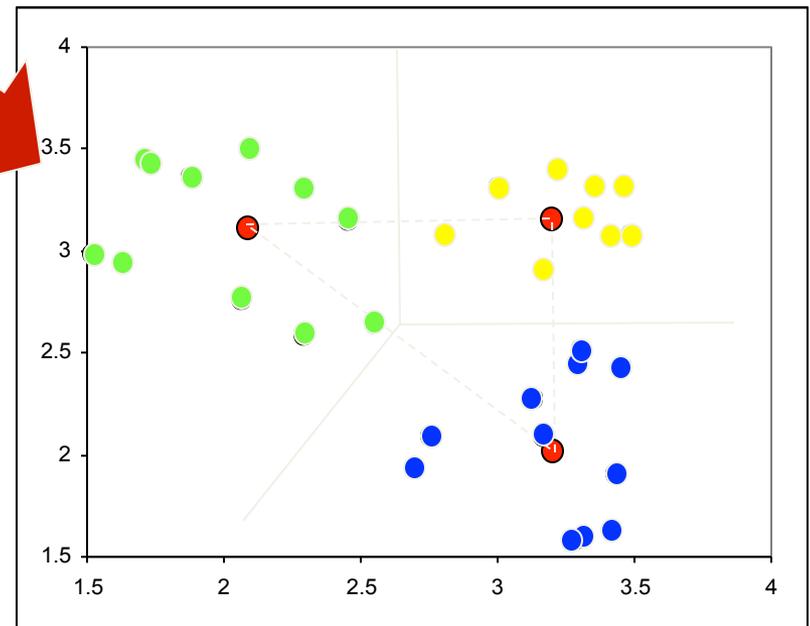
# Ejemplo de análisis de clúster (5)

## Iteración 4 y etapa final:

33



- Recalcular los centros como los centroides de los clusters desde la iteración 3.
- Nada cambió!!
- Ok, está listo.



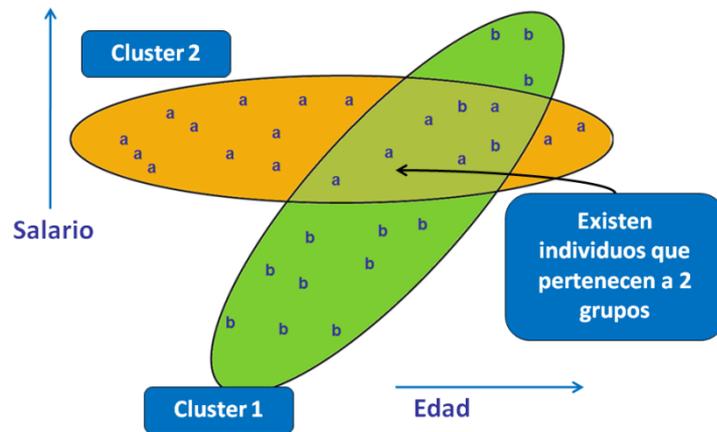
# Características del Algoritmo K-Means

- No se satisface el criterio de optimización globalmente, sólo produce un óptimo local.
- El algoritmo de k-means es computacionalmente rápido.
- Puede trabajar bien con datos faltantes (missing values).
- Es sensible a “outliers”.
- Los objetos pertenecen solo a un segmento.

# Agrupamiento con Lógica Difusa

35

El algoritmo Fuzzy Cmeans (FCM) es un método de segmentación que permite que un elemento pertenezca a más de un grupo mediante un grado de pertenencia.



Éste permite encontrar un conjunto de prototipos representativos de cada cluster y los grados de pertenencia de cada elemento a cada cluster.

$x_i$  : objeto  $i$ ;  $c_j$  : centro de clase  $j$ ;  
 $d^2(x_i, c_j)$  : distancia entre  $x_i$  y  $c_j$   
 $m$  : parámetro difuso ( $1 < m < \infty$ )

Algoritmo: Fuzzy c-means (FCM)

- $n$  objetos,  $c$  clases

\*  $u_{i,j}$  = grado de pertenencia de objeto  $i$  a clase  $j$ , ( $i=1, \dots, n$ ;  $j=1, \dots, c$ )

$$U = (u_{i,j})_{i,j}$$

$$u_{i,j} \in [0,1]; \sum u_{i,j} = 1; i = 1, \dots, n$$

# Algoritmo: Fuzzy C-means (FCM)

36

## Paso 1:

Determinar una matriz U con  $u_{i,j} \in [0,1]$ ;

## Paso 2:

Determinar los centros de las clases:

## Paso 3:

Actualizar los grados de pertenencia:

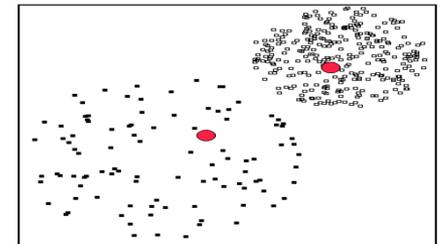
## Paso 4:

Criterio para detener:  $\|U^{k+1} - U^k\| < \varepsilon$

$$\sum_{j=1}^c u_{i,j} = 1$$

$$c_j = \frac{\sum_{i=1}^n u_{i,j} * x_i}{\sum_{i=1}^n u_{i,j}^m}$$

$$u_{i,j} = \frac{1}{\sum_{k=1}^c \left( \frac{d(x_i, c_j)}{d(x_i, c_k)} \right)^{\frac{2}{m-1}}}$$

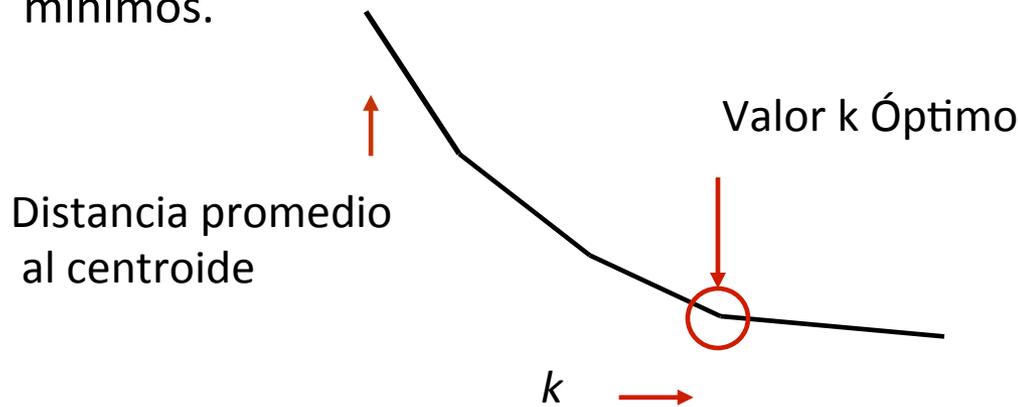


Fuzzy k-means

# Elección del número óptimo de segmentos “k”

## Regla del codo

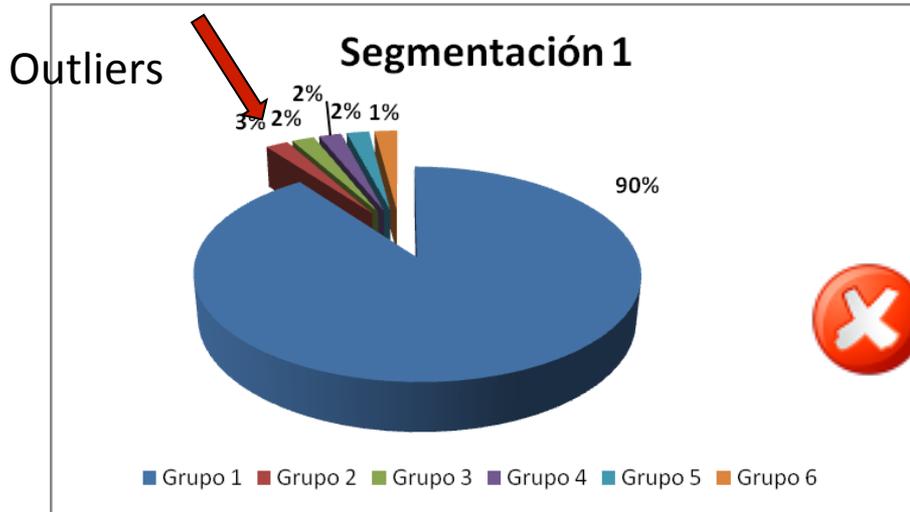
- Para escoger la cantidad adecuada de grupos, se deben observar los cambios en la distancia promedio al centroide a medida que el número de agrupaciones aumenta.
- Se observa que la distancia promedio disminuye rápidamente hasta encontrar en número  $k$  óptimo de particiones, luego, los cambios son mínimos.



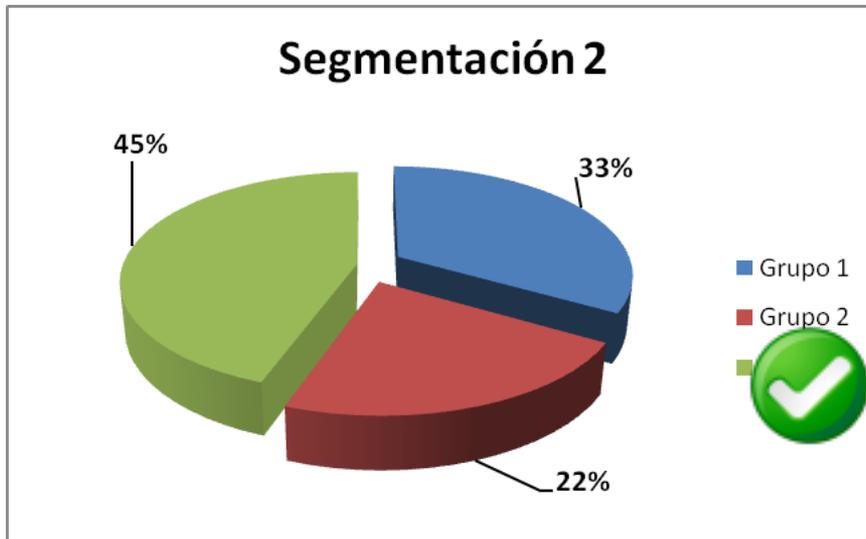
# Evaluación



38



En la figura de la segmentación 1 se puede observar un ejemplo de una mala segmentación, donde los grupos 2, 3, 4, 5 y 6, representan tan solo el 10%, lo anterior se debe a que los outliers no fueron detectados en el procesamiento de datos, por lo cual no se eliminaron



En la figura de la segmentación 2 los outliers fueron eliminados, obteniéndose 3 grupos.

# Bibliografía

39

- Jim Bezdek, Introduction to Clustering and Classification, EVIC. Santiago, Chile. 2005.
- Brian S. Everitt, Cluster Analysis, 1981
- Maurice Lorr, Cluster Analysis for Social Scientists, 1983
- Charles Romesburg, Cluster Analysis for Researchers, 1984
- Aldenderfer and Blashfield, Cluster Analysis, 1984
- Kmeans: <http://nlp.stanford.edu/IR-book/html/htmledition/k-means-1.html>
- SOM: <http://psb.stanford.edu/psb01/DAT/Hawaii10.pdf>
- Fuzzy Cmeans: <http://www.stanford.edu/~ctj/teeth/seg0.html>