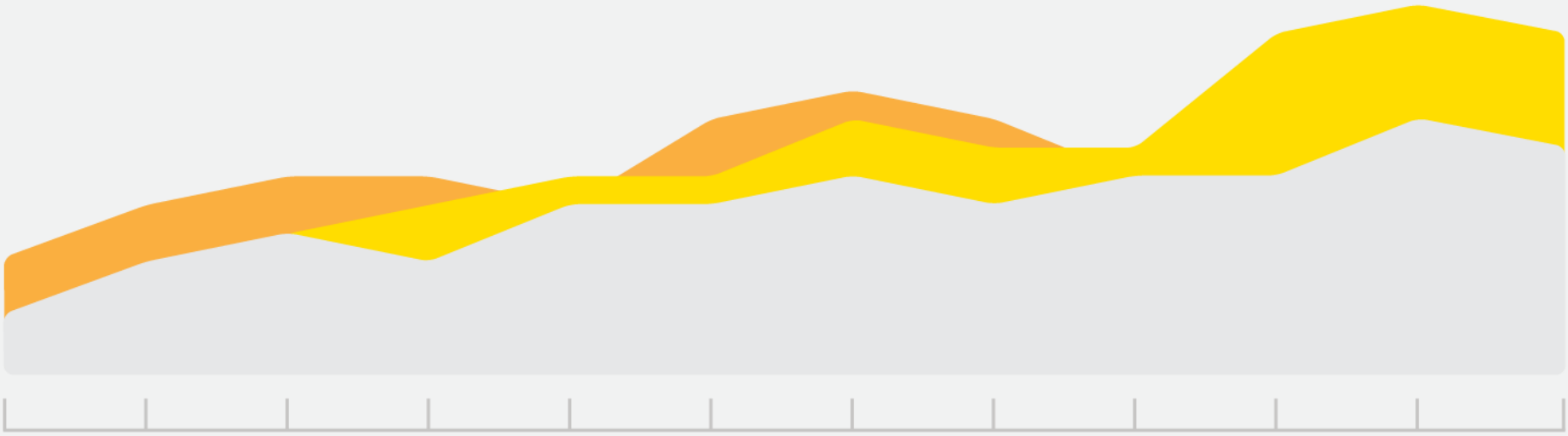
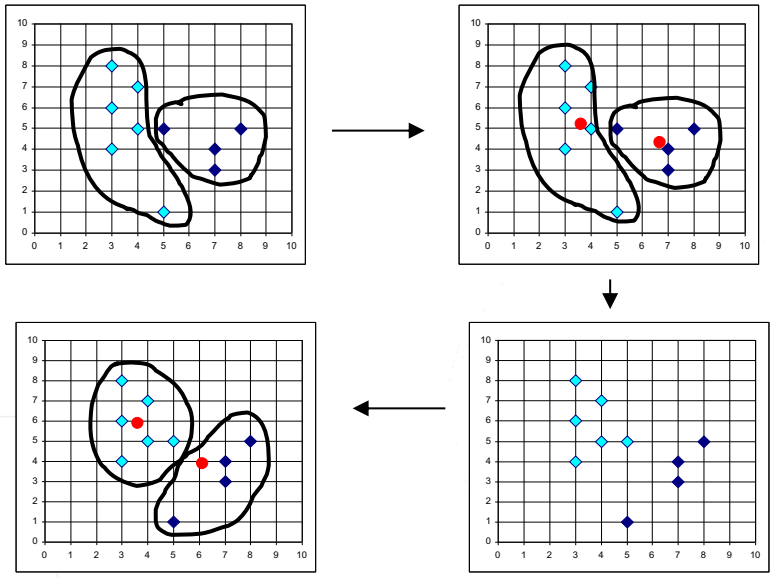
oldemar rodríguez
CONSULTOR en MINERIA DE DATOS

Método k-medias



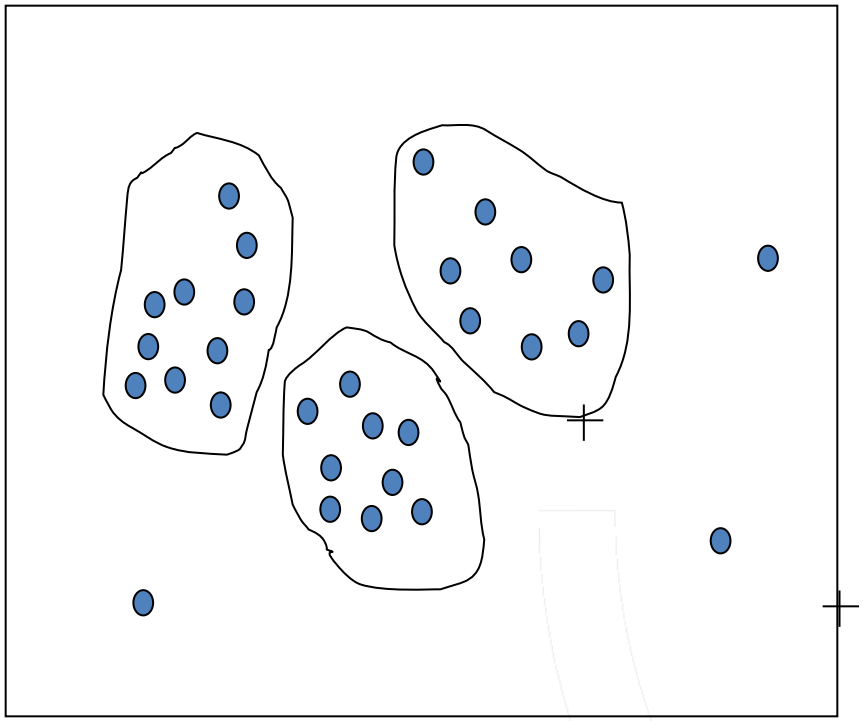
Método K-Means (Nubes Dinámicas)



Tareas de la Minería de Datos

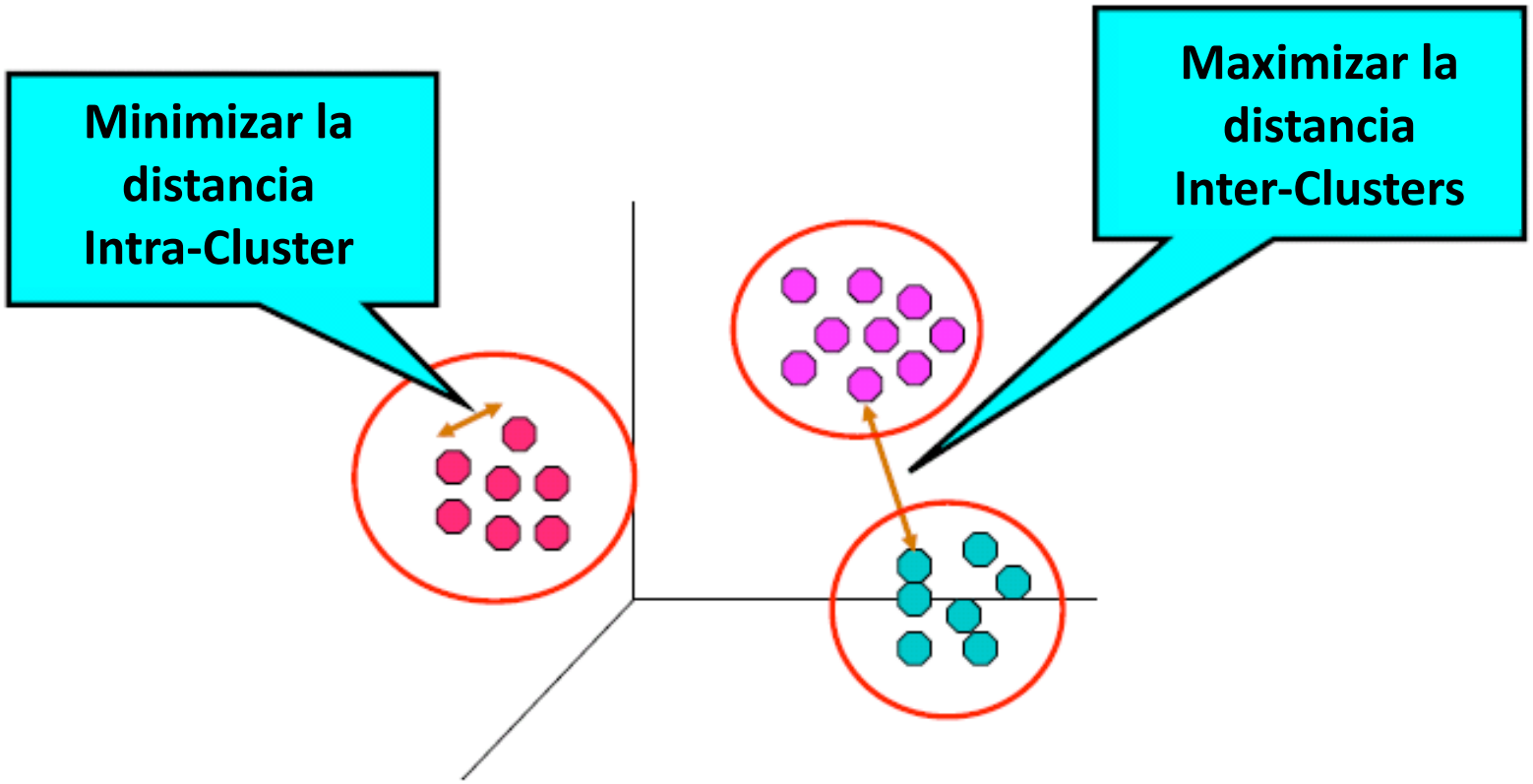
- ***"Clustering":*** (clasificación no supervisada, aprendizaje no supervizado): Es similar a la clasificación (discriminación), excepto que los grupos no son predefinidos. El objetivo es particionar o segmentar un conjunto de datos o individuos en grupos que pueden ser disjuntos o no. Los grupos se forman basados en la similaridad de los datos o individuos en ciertas variables. Como los grupos no son dados a priori el experto debe dar una interpretación de los grupos que se forman.
- Métodos:
  - Clasificación Jerárquica (grupos disjuntos).
  - Nubes Dinámicas – k-means (grupos disjuntos).
  - Clasificación Piramidal (grupos NO disjuntos).

Análisis de Conglomerados



Objetivo:
Obtener clases lo más homogéneas posibles y tal que estén suficientemente separadas.

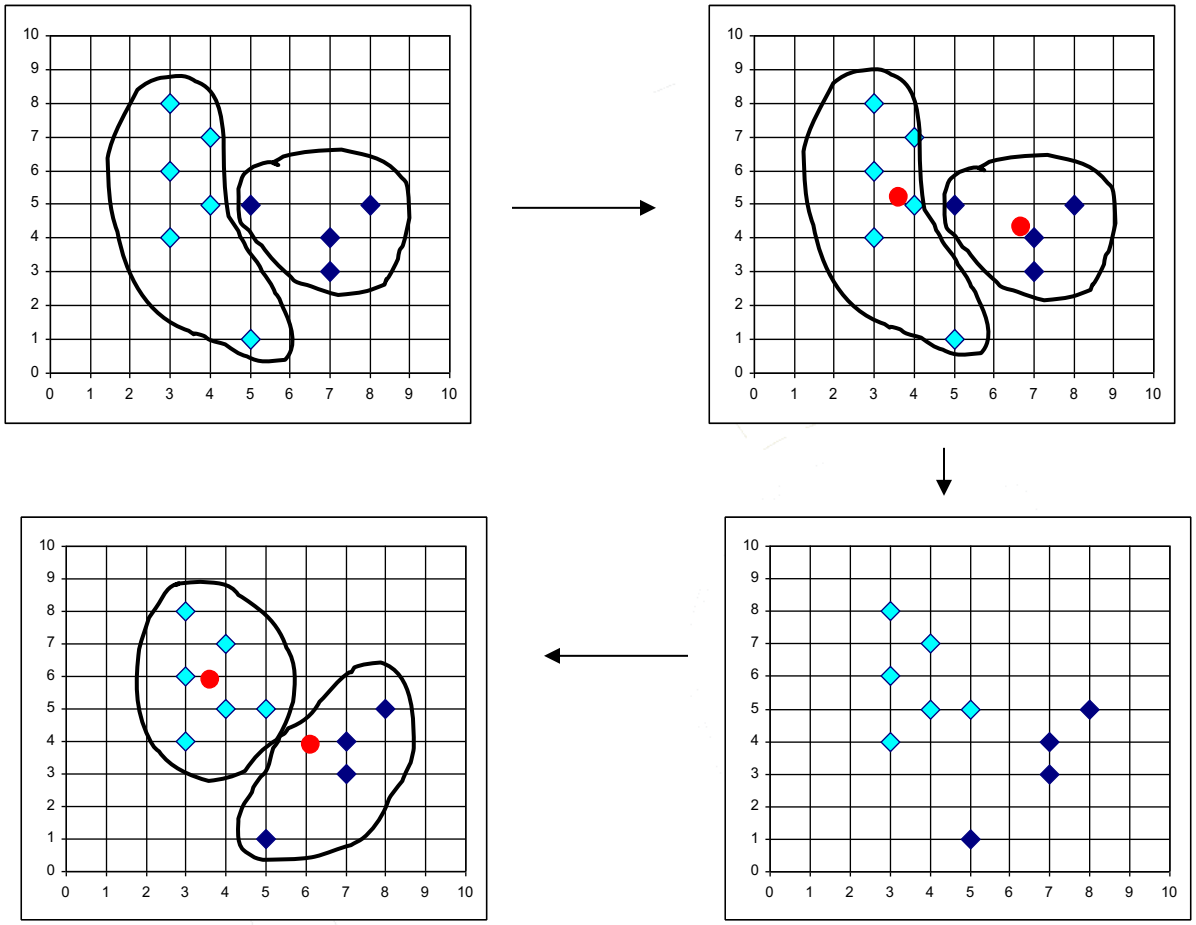
Criterio de la Inercia



Problema combinatorio

- Es necesario hacer notar que, cuando se quiere obtener una partición en *K* clases de un conjunto con *n* individuos, no tiene sentido examinar *todas* las posibles particiones del conjunto de individuos en *K* clases.
- En efecto, se está en presencia de un problema combinatorio muy complejo; sólo para efectos de ilustración, mencionemos que el número de particiones en 2 clases de un conjunto con 60 elementos es aproximadamente 10^{18}, y para 100 elementos en 5 clases anda por 10^{68}.

The *K-Means* Clustering Method (nubes dinámicas)



Criterio de la inercia

Como se ha mencionado, se quiere obtener clases lo más homogéneas posibles y tal que estén suficientemente separadas. Este objetivo se puede concretar numéricamente a partir de la siguiente propiedad:
supóngase que se está en presencia de una partición $P = (C_1, C_2, \ldots, C_K)$ de Ω, donde $\mathbf{g}_1, \mathbf{g}_2, \ldots, \mathbf{g}_K$ son los centros de gravedad de las clases:

$$\mathbf{g}_k = \frac{1}{|C_k|} \sum_{i \in C_k} \mathbf{x}_i,$$

$\mathbf{g}$ es el centro de gravedad total:

$$\mathbf{g} = \frac{1}{n} \sum_{i=1}^{n} \mathbf{x}_i$$

Ejemplo: Estudiantes

Ver NotasEscolaresExcelKMeans.xlsx

Análisis de los Clústeres

	Matemáticas	Ciencias	Español	Historia	EdFísica
Lucía	7	6.5	9.2	8.6	8
Pedro	7.5	9.4	7.3	7	7
Inés	7.6	9.2	8	8	7.5
Luis	5	6.5	6.5	7	9
Andrés	6	6	7.8	8.9	7.3
Ana	7.8	9.6	7.7	8	6.5
Carlos	6.3	6.4	8.2	9	7.2
José	7.9	9.7	7.5	8	6
Sonía	6	6	6.5	5.5	8.7
María	6.8	7.2	8.7	9	7

Centro Gravedad Total de la Nube de Puntos

Matemáticas	Ciencias	Español	Historia	EdFísica
6.79	7.65	7.74	7.9	7.42

Centro Gravedad C1={Pedro,Inés,Ana,José}

Matemáticas	Ciencias	Español	Historia	EdFísica
7.7	9.475	7.625	7.75	6.75

Centro Gravedad C2={Luis,Sonia}

Matemáticas	Ciencias	Español	Historia	EdFísica
5.5	6.25	6.5	6.25	8.85

Centro Gravedad C3={Lucía,Andrés,Carlos,María}

Matemáticas	Ciencias	Español	Historia	EdFísica
6.525	6.525	8.475	8.875	7.375

Definiciones

- *Inercia total* de la nube de puntos:

$$I = \frac{1}{n} \sum_{i=1}^{n} ||\mathbf{x}_i - \mathbf{g}||^2$$

- *Inercia inter-clases*, es decir la inercia de los centros de gravedad respecto al centro de gravedad total:

$$B(P) = \sum_{k=1}^{K} \frac{|C_k|}{n} ||\mathbf{g}_k - \mathbf{g}||^2$$

- *Inercia intra-clases*, es decir la inercia al interior de cada clase:

$$W(P) = \sum_{k=1}^{K} I(C_k) = \frac{1}{n} \sum_{k=1}^{K} \sum_{i \in C_k} \|\mathbf{x}_i - \mathbf{g}_k\|^2$$

Teorema: Igualdad de Fisher

- *Inercia total = Inercia inter-clases*

  +

  Inercia intra-clases

$$I = B(P) + W(P)$$

Ejemplo: Estudiantes

Ver NotasEscolaresExcelKMeans.xlsx

Análisis de los Clústeres

	Matemáticas	Ciencias	Español	Historia	EdFísica
Lucía	7	6.5	9.2	8.6	8
Pedro	7.5	9.4	7.3	7	7
Inés	7.6	9.2	8	8	7.5
Luis	5	6.5	6.5	7	9
Andrés	6	6	7.8	8.9	7.3
Ana	7.8	9.6	7.7	8	6.5
Carlos	6.3	6.4	8.2	9	7.2
José	7.9	9.7	7.5	8	6
Sonía	6	6	6.5	5.5	8.7
María	6.8	7.2	8.7	9	7

Cálculo de I=Inercia Total

	4.3246
	4.7466
	3.1426
	9.3706
	4.3646
	5.6806
	3.2726
	7.5186
	12.283
	2.5106
I=	*5.7214*

Cálculo de B(P)=Inercia Inter-Clases

	4.6434
	9.9291
	2.8287
B(P)=	4.9747

Cálculo de W(P)=Inercia Intra-Clases

1.2181	0.7763	0.8975
1.0131	0.8513	0.8975
0.1881	0.1563	1.795
0.7381	0.7313	
3.1575	2.515	

W(P)= *0.7468*

Centro Gravedad Total de la Nube de Puntos

Matemáticas	Ciencias	Español	Historia	EdFísica
6.79	7.65	7.74	7.9	7.42

I=B(P)+W(P) *5.7214*

Centro Gravedad C1={Pedro,Inés,Ana,José}

Matemáticas	Ciencias	Español	Historia	EdFísica
7.7	9.475	7.625	7.75	6.75

Centro Gravedad C2={Luis,Sonia}

Matemáticas	Ciencias	Español	Historia	EdFísica
5.5	6.25	6.5	6.25	8.85

Centro Gravedad C3={Lucía,Andrés,Carlos,María}

Matemáticas	Ciencias	Español	Historia	EdFísica
6.525	6.525	8.475	8.875	7.375

- ***Objetivo:*** Se quiere que *B(P)* sea máxima y *W(P)* sea mínima

- Como la inercia *I(P)* es fija, dada la nube de puntos, entonces al maximizar *B(P)* se minimiza automáticamente *W(P)*.

- Por lo tanto, los dos objetivos (homogeneidad al interior de las clases y separación entre las clases) se alcanzan al mismo tiempo al querer minimizar *W(P)*.

Teorema: Igualdad de Fisher

Dada una partición $P = (C_1, \ldots, C_K)$ de un conjunto de n individuos Ω en K clases, entonces la propiedad de Fisher establece que

$$I(\Omega) = W(P) + B(P)$$

donde $I(\Omega) = \frac{1}{n}\sum_{\mathbf{x}_i \in \Omega} d^2(\mathbf{x}_i, \mathbf{g})$, $W(P) = \frac{1}{n}\sum_{k=1}^{K}\sum_{\mathbf{x}_i \in C_k} d^2(\mathbf{x}_i, \mathbf{g}_k)$, y $B(P) = \sum_{k=1}^{K} \frac{|C_k|}{n} d^2(\mathbf{g}_k, \mathbf{g})$, siendo $\mathbf{g} = \frac{1}{n}\sum_{\mathbf{x}_i \in \Omega} \mathbf{x}_i$ el centro de gravedad total, $\mathbf{g}_k = \frac{1}{|C_k|}\sum_{\mathbf{x}_i \in C_k} \mathbf{x}_i$ el centro de gravedad de la clase C_k, y d una distancia Euclídea que se puede escribir como $d(\mathbf{x}, \mathbf{y}) = \|\mathbf{x} - \mathbf{y}\|_{\mathbf{M}}$, para alguna métrica $\mathbf{M}$. Se supone que todos los individuos tienen mismo peso igual a $1/n$.

Prueba:

$$
\begin{aligned}
I(\Omega) &= \frac{1}{n}\sum_{i=1}^{n}\|\mathbf{x}_i-\mathbf{g}\|_{\mathbf{M}}^2 \\
&= \frac{1}{n}\sum_{k=1}^{K}\sum_{\mathbf{x}_i\in C_k}\|\mathbf{x}_i-\mathbf{g}_k+\mathbf{g}_k-\mathbf{g}\|_{\mathbf{M}}^2 \\
&= \frac{1}{n}\sum_{k=1}^{K}\sum_{\mathbf{x}_i\in C_k}\|\mathbf{x}_i-\mathbf{g}_k\|_{\mathbf{M}}^2+\frac{1}{n}\sum_{k=1}^{K}\sum_{\mathbf{x}_i\in C_k}\|\mathbf{g}_k-\mathbf{g}\|_{\mathbf{M}}^2 \\
&\quad +\frac{2}{n}\sum_{k=1}^{K}\sum_{\mathbf{x}_i\in C_k}\langle\mathbf{x}_i-\mathbf{g}_k,\mathbf{g}_k-\mathbf{g}\rangle_{\mathbf{M}}
\end{aligned}
$$

Prueba:

$$
\begin{aligned}
&= \frac{1}{n}\sum_{k=1}^{K}\sum_{\mathbf{x}_i \in C_k} \|\mathbf{x}_i - \mathbf{g}_k\|_{\mathbf{M}}^2 + \frac{1}{n}\sum_{k=1}^{K}\sum_{\mathbf{x}_i \in C_k} \|\mathbf{g}_k - \mathbf{g}\|_{\mathbf{M}}^2 \\
&\quad + \frac{2}{n}\sum_{k=1}^{K}\sum_{\mathbf{x}_i \in C_k} \langle \mathbf{x}_i - \mathbf{g}_k, \mathbf{g}_k - \mathbf{g}\rangle_{\mathbf{M}} \\
&= W(P) + \sum_{k=1}^{K}\frac{|C_k|}{n}\|\mathbf{g}_k - \mathbf{g}\|_{\mathbf{M}}^2 \\
&\quad + \frac{2}{n}\sum_{k=1}^{K}\left\langle \sum_{\mathbf{x}_i \in C_k} \mathbf{x}_i - \sum_{\mathbf{x}_i \in C_k} \mathbf{g}_k, \mathbf{g}_k - \mathbf{g} \right\rangle_{\mathbf{M}} \\
&= W(P) + B(P),
\end{aligned}
$$

pues $\sum_{\mathbf{x}_i \in C_k} \mathbf{x}_i - \sum_{\mathbf{x}_i \in C_k} \mathbf{g}_k = |C_k|\mathbf{g}_k - |C_k|\mathbf{g}_k = \mathbf{0}$.

Objetivo del Método K-means

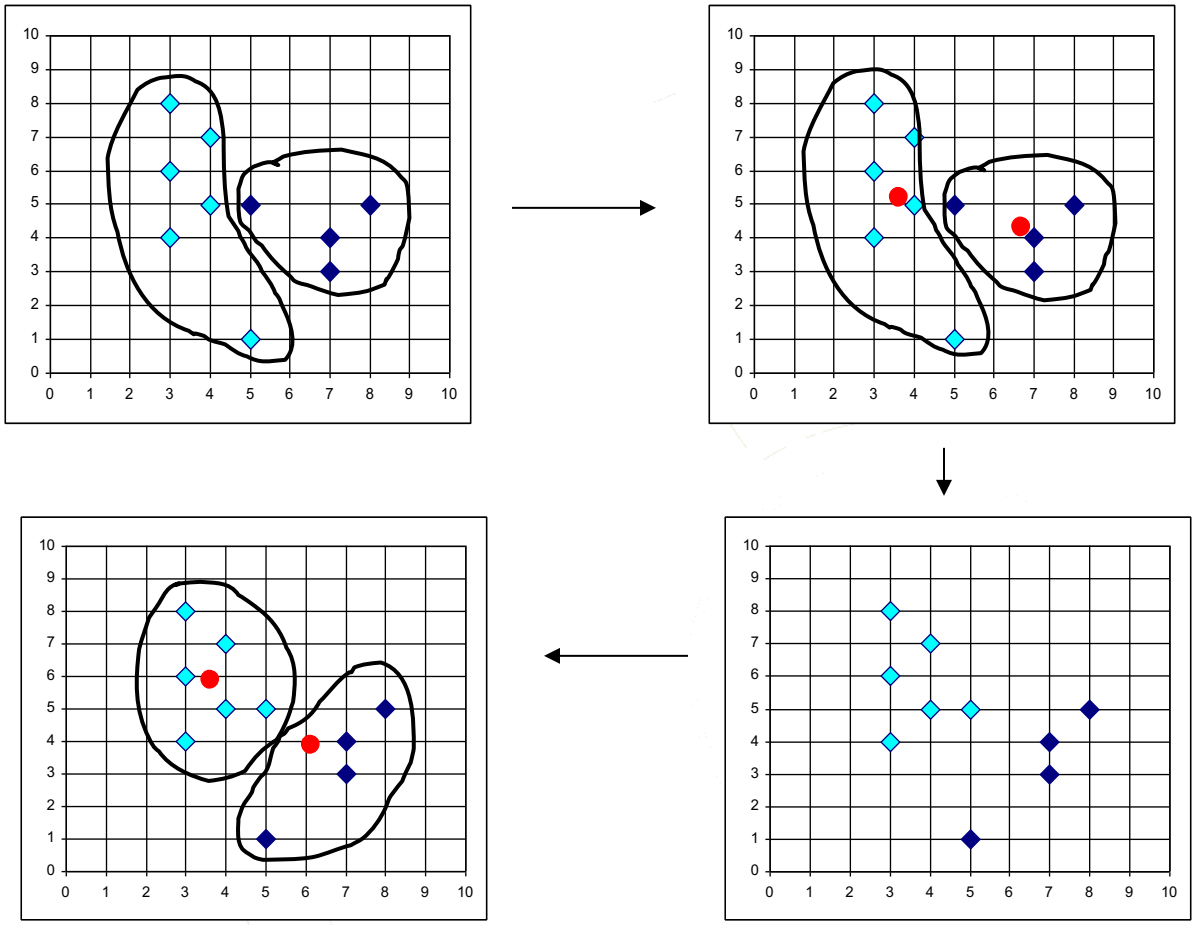
- Así, el objetivo en el método de K-means es encontrar una partición *P* de **W** y representantes de las clases, tales que *W*(*P*) sea mínima.

Método de k-medias

- Existe un poco de confusión en la literatura acerca del método de las k-medias, ya que hay dos métodos distintos que son llamados con el mismo nombre.
- Originalmente, Forgy propuso en 1965 un primer método de reasignación-recentraje que consiste básicamente en la iteración sucesiva, hasta obtener convergencia, de las dos operaciones siguientes:

1. Representar una clase por su centro de gravedad, esto es, por su vector de promedios.
2. Asignar los objetos a la clase del centro de gravedad más cercano.

The *K-Means* Clustering Method (nubes dinámicas)



- McQueen propone un método muy similar, donde también se representan las clases por su centro de gravedad, y se examina cada individuo para asignarlo a la clase más cercana.
- La diferencia con el método de Forgy es que inmediatamente después de asignar un individuo a una clase, el centro de ésta es recalculado, mientras que Forgy primero hacía todas las asignaciones y luego recalculaba los centros.
- Variantes del método de Forgy son propuestas en Francia como Método de Nubes Dinámicas por E. Diday a partir en 1967.
- Es McQueen quien propone el nombre “k-means”, que se usa hasta la fecha, aún si estos métodos también reciben nombres como nubes dinámicas, centros móviles, o reasignación-recentraje.

K-Means Clustering Algorithm

Algorithm 1 Basic K-means Algorithm.

1: Select K points as the initial centroids.
2: **repeat**
3: Form K clusters by assigning all points to the closest centroid.
4: Recompute the centroid of each cluster.
5: **until** The centroids don't change

Método de Forgy

Denotaremos Ω el conjunto de n individuos que queremos clasificar, todos dotados de pesos iguales $1/n$, y supondremos que están descritos por p variables cuantitativas $x^1, x^2, \ldots, x^p$.

En el caso en que se está en presencia de variables cuantitativas, tiene sentido el cálculo de promedios y de distancias Euclídeas. Por lo tanto, también tiene sentido que cada clase esté representada por su centro de gravedad, esto es, por un individuo ficticio cuyas coordenadas son los valores promedio de las variables para los individuos pertenecientes a la clase. Este es el caso más simple y el usado más corrientemente. Generalmente, se usará la distancia Euclídea clásica en este contexto.

Algoritmo: K-means

0. *Inicialización:* Escoger al azar K objetos de Ω, que servirán como núcleos iniciales[8]. Esto es, escoger al azar $\mathbf{g}_1, \ldots, \mathbf{g}_k$ en Ω; sean

$$C_1 := \emptyset, \ldots, C_K := \emptyset.$$

1. *Asignación:* Asignar cada objeto a la clase del centro de gravedad más cercano. Es decir, para todo $i \in \Omega$ hacer:
si $d(\mathbf{x}_i, \mathbf{g}_{k^*}) \leq \{d(\mathbf{x}_i, \mathbf{g}_k)$ para todo $k = 1, \ldots, K\}$ entonces asignar $\mathbf{x}_i$ a la clase C_{k^*}; si el mínimo se alcanza para dos clases diferentes entonces asignarlo a la clase de índice menor.

2. *Representación:* Calcular los centros de gravedad de la partición. Así, para todo $k \in \{1, \ldots, K\}$ hacer: $\mathbf{g}_k := \frac{1}{|C_k|} \sum_{\mathbf{x}_i \in C_k} \mathbf{x}_i$.
Calcular el criterio $W := \sum_{k=1}^{K} \sum_{i \in C_k} ||\mathbf{x}_i - \mathbf{g}_k||^2$.

Algoritmo: K-means

3. *Control de parada:* Si la variación en el criterio W entre la iteración anterior y la presente es menor que un umbral dado, o si se sobrepasa el número máximo de iteraciones entonces detenerse, de lo contrario ir al paso 4.

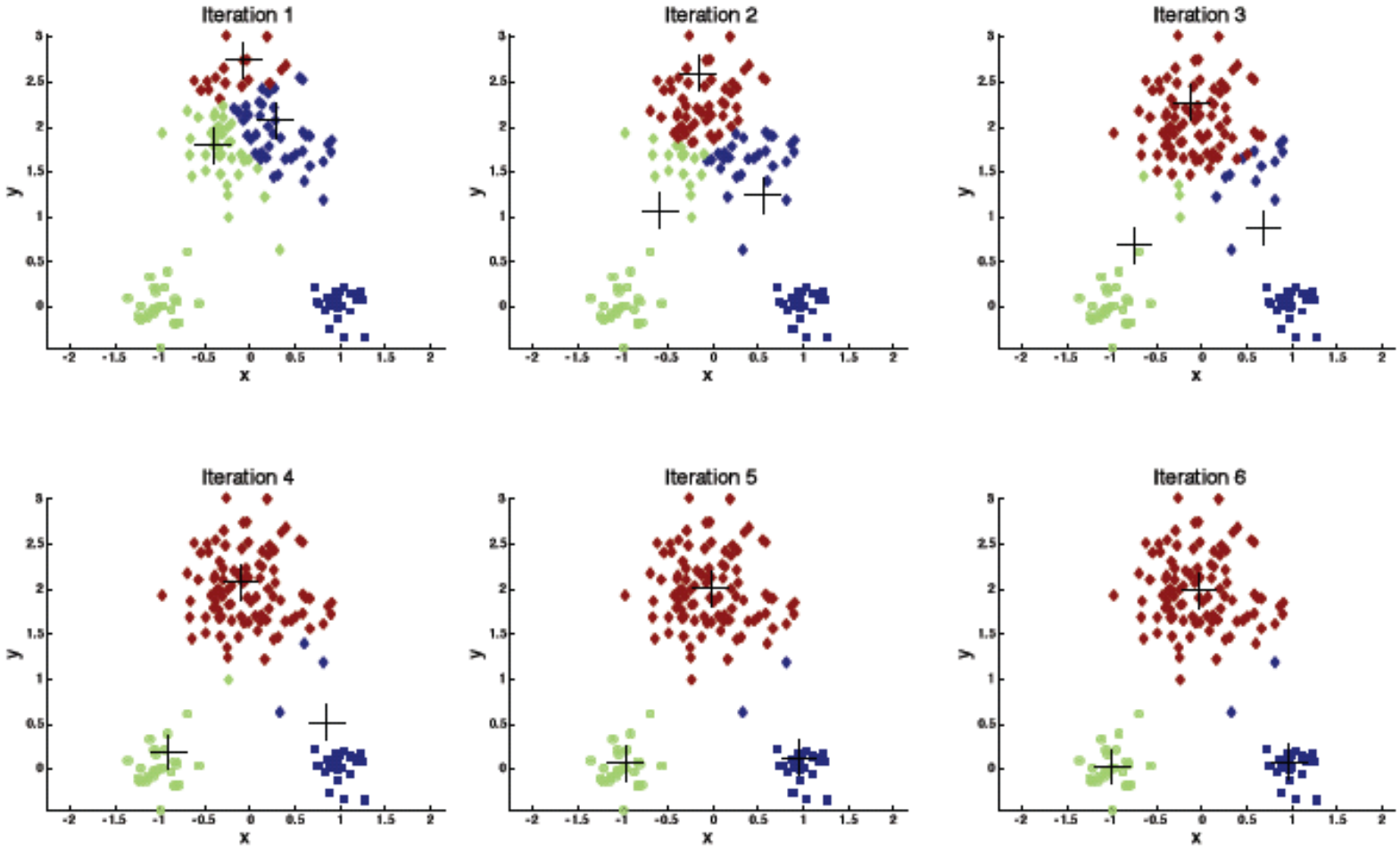
4. *Preparación:* Poner $C_1 := \emptyset, \ldots, C_K := \emptyset$; ir al paso 1.

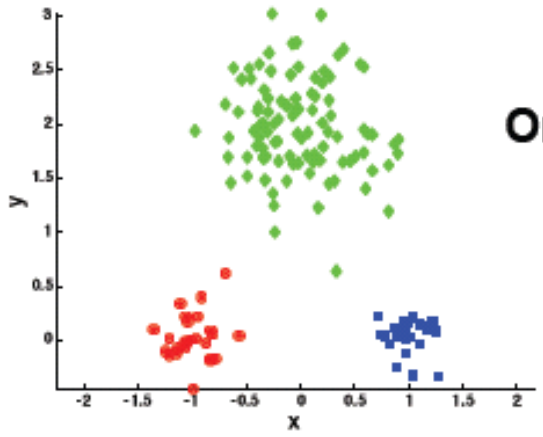
El resultado de la aplicación del método de k-medias, dependerá de la escogencia inicial de los núcleos. Por ello, se recomienda correr varias veces el método y escoger la mejor solución obtenida en esas corridas.

Ejemplo de las notas escolares

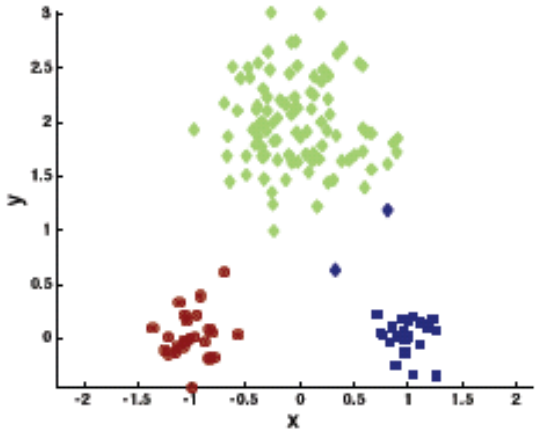
Formas Fuertes

Partición P	Número de veces obtenida	$W(P)$	$B(P)$
$C_1 = \{$Lucía,Andrés,Carlos,María$\}$ $C_2 = \{$Luis,Sonia$\}$ $C_3 = \{$Pedro,Inés,Ana,José$\}$	17 (68%)	0.75	4.97
$C_1 = \{$Lucía,Andrés,Carlos,María, Luis,Sonia$\}$ $C_2 = \{$Pedro,Inés$\}$ $C_3 = \{$Ana,José$\}$	3 (12%)	2.48	3.24
$C_1 = \{$Lucía,Andrés,Carlos,María, Luis,Sonia$\}$ $C_2 = \{$Inés,Ana,José$\}$ $C_3 = \{$Pedro$\}$	2 (8%)	2.52	3.20
$C_1 = \{$Lucía,Andrés,Carlos,María, Luis,Sonia$\}$ $C_2 = \{$Inés,Ana$\}$ $C_3 = \{$Pedro,José$\}$	1 (4%)	2.55	3.17
$C_1 = \{$Lucía,Andrés,Carlos,Luis, Sonia$\}$ $C_2 = \{$Pedro,Inés$\}$ $C_3 = \{$Ana,José,María$\}$	1 (4%)	2.72	3.00
$C_1 = \{$Lucía,Andrés,Carlos,María, Pedro,Inés,Ana,José$\}$ $C_2 = \{$Luis$\}$ $C_3 = \{$Sonia$\}$	1 (4%)	3.06	2.66

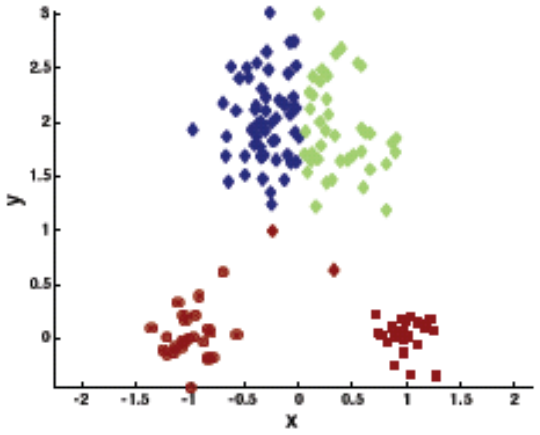




Original Points

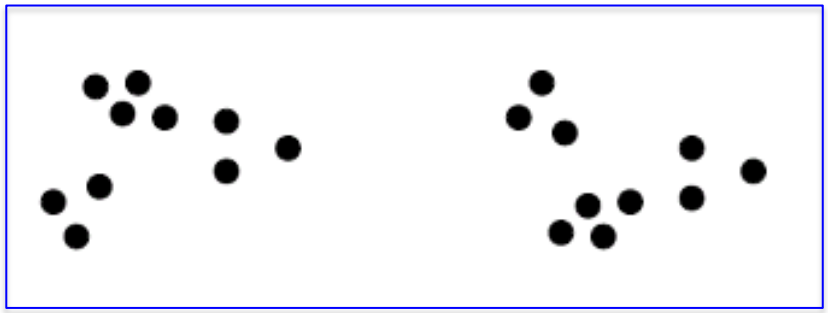


Optimal Clustering

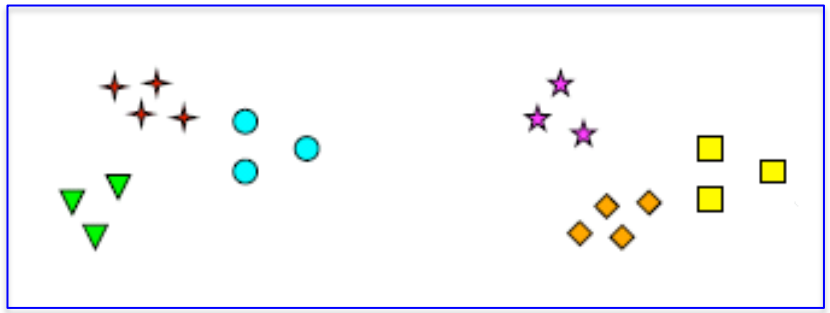


Sub-optimal Clustering

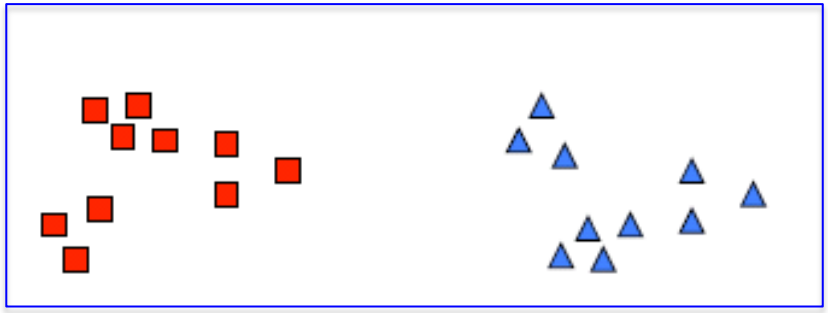
¿Cuántos clústeres?



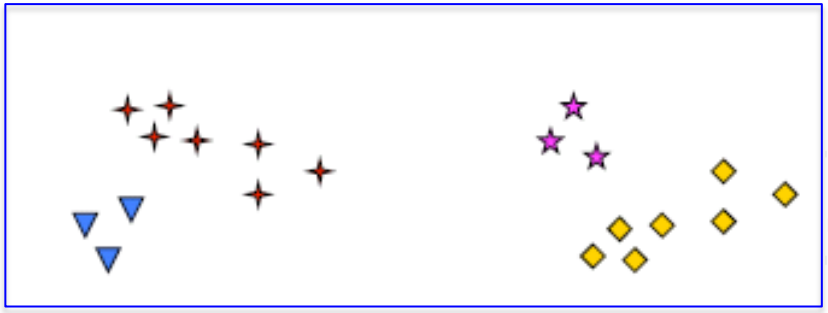
Datos originales



6 clústeres

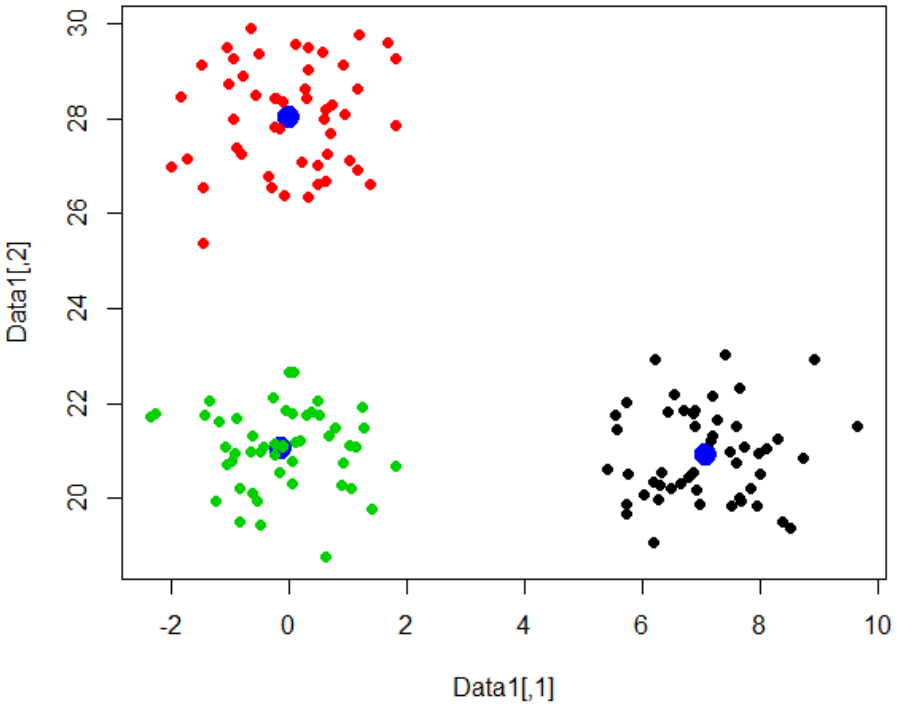


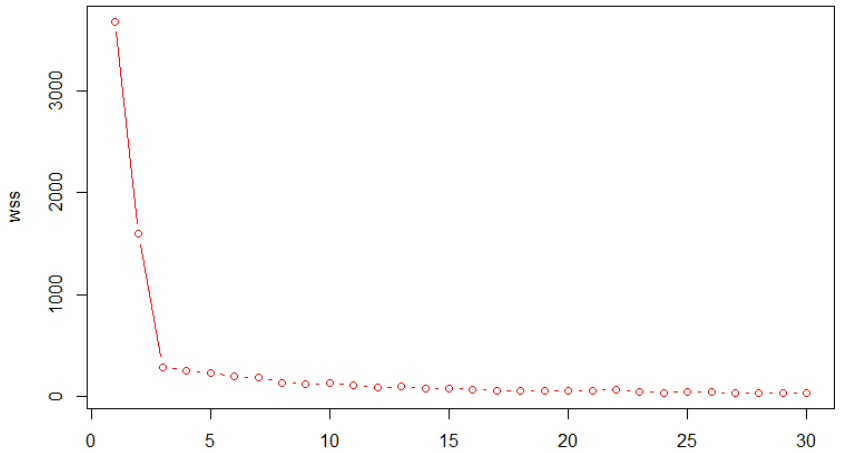
2 clústeres



4 clústeres

¿Cuántos clústeres?





k x (inercia intra-clases)

El “codo” indica que k=3 es la cantidad adecuada de clústeres

K-means en R

Description

Perform k-means clustering on a data matrix.

Usage

```
kmeans(x, centers, iter.max = 10, nstart = 1,
       algorithm = c("Hartigan-Wong", "Lloyd", "Forgy",
                     "MacQueen"))
## S3 method for class 'kmeans'
fitted(object, method = c("centers", "classes"), ...)
```

Arguments

`x`	numeric matrix of data, or an object that can be coerced to such a matrix (such as a numeric vector or a data frame with all numeric columns).
`centers`	either the number of clusters, say *k*, or a set of initial (distinct) cluster centres. If a number, a random set of (distinct) rows in `x` is chosen as the initial centres.
`iter.max`	the maximum number of iterations allowed.
`nstart`	if `centers` is a number, how many random sets should be chosen?
`algorithm`	character: may be abbreviated.
`object`	an R object of class `"kmeans"`, typically the result `ob` of `ob <- kmeans(..)`.
`method`	character: may be abbreviated. `"centers"` causes `fitted` to return cluster centers (one for each input point) and `"classes"` causes `fitted` to return a vector of class assignments.
`...`	not used.

K-means en R

Value

`kmeans` returns an object of class `"kmeans"` which has a `print` and a `fitted` method. It is a list with components:

`cluster`	A vector of integers (from `1:k`) indicating the cluster to which each point is allocated.
`centers`	A matrix of cluster centres.
`totss`	The total sum of squares.
`withinss`	Vector of within-cluster sum of squares, one component per cluster.
`tot.withinss`	Total within-cluster sum of squares, i.e., `sum(withinss)`.
`betweenss`	The between-cluster sum of squares, i.e. `totss-tot.withinss`.
`size`	The number of points in each cluster.

Gracias....

oldemar rodríguez

CONSULTOR en M1NƐR14 DƐ D4T0S