



Afi Escuela
de Finanzas

Clustering

Daniel Vélez Serrano
Febrero 2022

Índice

1. Introducción al análisis cluster
2. Metodología
3. Tratamiento de variables
4. Selección de la métrica de desemejanza
5. Selección del tipo de algoritmo
6. Validación de resultados
7. Caso de uso: Segmentación de Estados en función de su nivel de criminalidad
8. Práctica: Segmentación bancaria

1 | Introducción al análisis cluster

Introducción

- El análisis cluster se engloba dentro de la **modelización no supervisada** siendo su objetivo el de agrupar elementos en bloques homogéneos en función de las similitudes entre ellos respecto de una relación de variables input (al ser no supervisado, no existe una variable target). Siendo G el n° de grupos y p el n° de variables, se trata de conseguir:
 - Que los elementos de un mismo grupo sean lo más parecido posible: **maximizar la homogeneidad intragrupo (o minimizar la varianza dentro de cada grupo)**.

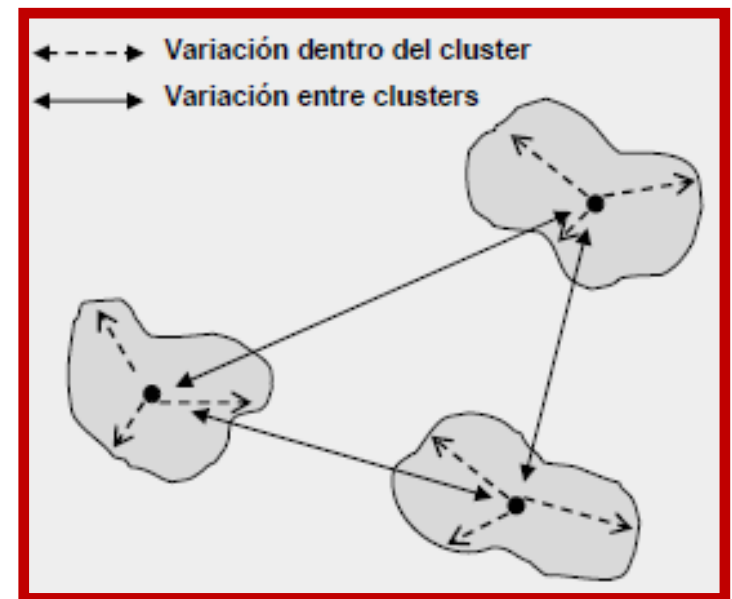
WSS = Within Cluster Sum Of Square =

$$\sum_{i=1}^{n_g} \sum_{j=1}^p (x_{ij} - \bar{x}_j)^2$$

- Que los elementos de dos grupos distintos sean lo más diferente posible: **maximizar la heterogeneidad intergrupo (o maximizar la varianza entre los grupos)**.

Intercluster dissimilarity (distance) =

$$\sum_{g=1}^G \sum_{j=1}^p (\bar{x}_{gj} - \bar{x}_j)^2$$



↑ Índice de Dunn =

$$= \frac{\text{distancia entre centroides (p.e)}}{WSS}$$

Introducción

- El primer paso es determinar el conjunto de elementos sobre el que se quiere obtener una relación de grupos.
- En el ámbito “*marketiniano*” es habitual querer realizar una segmentación de la cartera para aportar conocimiento sobre la tipología de clientes que hay en la compañía. Dentro de este contexto, más allá del análisis descriptivo multivariante, una segmentación puede perseguir otros objetivos:
 - **Articular una acción comercial diferente a cada grupo de clientes.-** interesa que éstos tengan un tamaño suficiente que justifique la inversión que supone realizarla.
 - **Analizar transiciones entre grupos.-** en aquellos casos en los que a éstos es posible asociar un valor representativo de la rentabilidad o el beneficio que supone para la compañía. Para hacer este análisis es necesario garantizar que los segmentos son estables y existen en dos momentos temporales distintos. Por ello se realizan segmentaciones con diferentes muestras en diferentes instantes.
 - **Identificar patrones fraudulentos.-** suelen ser grupos de tamaño reducido que reflejan patrones de comportamiento atípico. De hecho el análisis cluster es un **método de detección de outliers multivariante**.

Introducción

- En función del objetivo marcado se deben **determinar las variables de análisis**, es decir, aquéllas respecto a las cuales se realizará la segmentación.
- Para una misma población de análisis pueden existir juegos diferentes de variables.

Por ejemplo:

Segmentación 1: Perfil bancario

Cuenta a la vista
Depósitos
Fondos de inversión
Acciones/valores
Planes de pensiones
Préstamos hipotecarios
Préstamos personales
Líneas de crédito
Tarjetas de crédito
Tarjetas de débito
Recibos domiciliados
Domiciliación de nómina, pensión, desempleo

Segmentación 2: Perfil humano

Necesidades básicas: alimentación, transporte, etc.
Hijos: material escolar, colegio, etc.
Coche: garaje, taller, parking, etc.
Tecnología: móvil, ordenador, etc.
Ocio: espectáculos, restaurantes, apuestas, etc.
Moda y complementos: tiendas de ropa
Viajes: pago de viajes, compras en el extranjero, etc.
Salud y cuidado personal: gimnasio, sociedades médicas, etc.
Altruista: cruz roja, ONGs, etc.

- Su definición en ocasiones no es inmediata, pudiendo ser necesario disponer de un **conocimiento especializado del negocio**.

2 | Metodología

Metodología

Tratamiento de variables

Imputación de *missings*.- para que puedan calcularse las distancias (o aplicacr una métrica que los tenga en cuenta).

Estandarización.- para que en el cálculo de distancias no tengan más peso aquellas variables con mayor dispersión.

Tratamiento de *outliers*.- pues suelen distorsionar los resultados.

Eliminación de redundancias.- para que no pesen más unas variables que otras.

Seleccionar la métrica de desemejanza a utilizar

Lo habitual es utilizar métricas euclídeas pero **depende** también del **tipo de variables** que se tenga (continuas, ordinales, binarias).

Elegir el tipo de algoritmo de clasificación y segmentar

Elegir el algoritmo de segmentación y aplicarlo teniendo en cuenta la medida de semejanza seleccionada.

Validación de resultados

Asignar un nombre a cada cluster en **función del valor medio** que toman las variables de análisis en dicho grupo, es decir, en función del perfil del **centroide**. Hay que valorar si el **volumen** del grupo es **suficiente** para extraer un resultado concluyente o aplicable.

3 | Tratamiento de variables

Tratamiento de variables: *missings*

- Si para un **registro todas las observaciones tienen valores *missing***, dicho registro es eliminado. En el caso de que existan valores ***missings* pero no para todas las variables** se plantean diversas **soluciones**:
- **Omitir** dichas observaciones.- el **riesgo** que acarrea esta posibilidad es el de eliminar demasiados registros y terminar trabajando con **muy pocos datos o datos sesgados**.
- **Asociarles un valor (imputación de *missings*)**.-
 - Con un valor fijo (media para variables intervalo, moda para variables de clase, etc.)
 - De acuerdo a la **distribución** de la propia variable.
 - Realizando una **predicción** en función de los otros inputs.
 - Generando una **categoría *missing*** con un valor específico.
- También se puede **calcular la distancia** entre las observaciones pero **únicamente respecto de las variables informadas**. En dicho caso, la **métrica** utilizada es **corregida** teniendo en cuenta el número de variables que tienen observaciones *missings*.

$$\sqrt{\frac{n}{m} \sum_{i=1}^n (x_i - s_i)^2}$$

- n = número de variables; m = número de variables con valores no *missings*.
- x_i = valor de la i -ésima variable para una observación.
- s_i = valor de la i -ésima variable para un centroide.

Tratamiento de variables: estandarizar

- El análisis cluster es **muy sensible a las unidades de medida seleccionadas**.
- Las variables que tienen un **mayor rango de variación** tienden a tener **más importancia** a la hora de conformar los clusters. Por ello se **recomienda estandarizar todas las variables**.
Ejemplo: Hacer una segmentación en función de la (renta, edad).



Tiene una renta de 20.000€ anuales y 75 años



Tiene una renta de 22.000€ anuales y 25 años

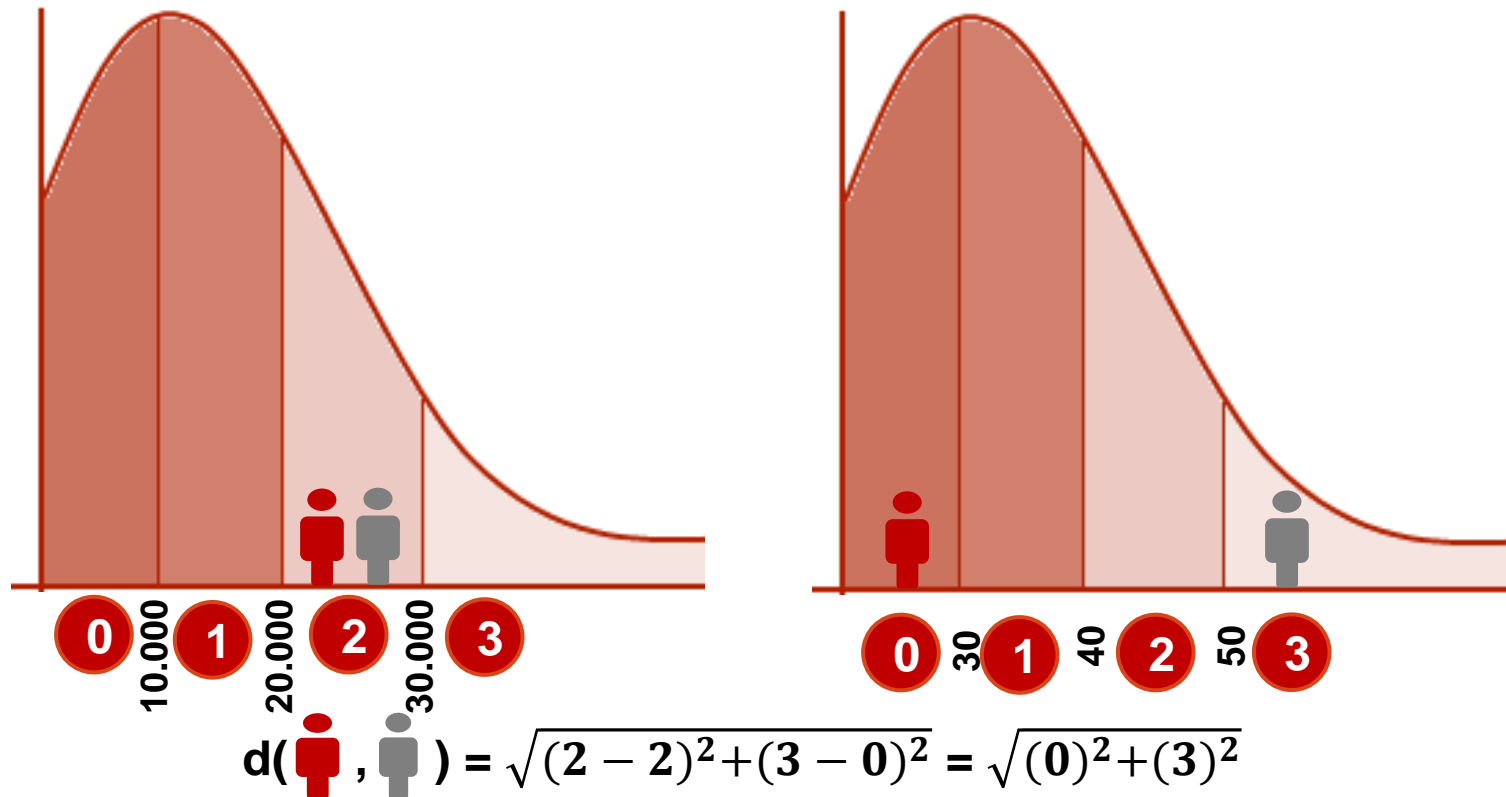
- Al aplicar la fórmula, los valores que marcan la distancia/proximidad entre los dos individuos son los asociados a la variable renta, cuando la distancia real entre ellos parece venir marcada por la edad.

$$d(\text{👤}, \text{👤}) = \sqrt{(20.000 - 22.000)^2 + (75 - 25)^2} = \sqrt{(2.000)^2 + (50)^2}$$

- El método habitual de estandarizar consiste en restar la media y dividir por la desviación típica.
- Otros métodos posibles son dividir por el rango (máximo – mínimo) o discretizar las variables.

Tratamiento de variables: estandarizar

- Otra posibilidad consiste en **discretizar** el valor de las mismas a partir de los percentiles de su distribución (**encoding tipo rank**). Este método **evita la necesidad de realizar un tratamiento posterior de outliers**.




El **inconveniente** que posee este método de estandarización son los **valores frontera**: dos individuos muy próximos que pertenecen a categorías diferentes.

Tratamiento de variables: estandarizar

- En cuanto a las **variables discretas** se refiere, en primer lugar es necesario **hacerlas cuantitativas**. La manera de recodificarlas difiere en función de que sean ordinales o nominales. Los métodos que **tradicionalmente** se utilizan son la **codificación rango para variables ordinales** y la **codificación GLM para nominales**.

X

d
c
b
a
d
c
b
d
c
d



	RANK (ordinal)	GLM (nominal) OneHot Encoding			
		X(a)	X(b)	X(c)	X(d)
a	0	1	0	0	0
b	0.333333	0	1	0	0
c	0.666667	0	0	1	0
d	1	0	0	0	1

- Se pueden plantear **codificaciones alternativas** en las que el 0 es sustituido por -1:
 - **Si la estandarización de las variables continuas se hace en rango** (se mueven entre 0 y 1), codificar las categóricas en 0's y 1's da demasiado peso a cada categoría: la diferencia entre dos categorías es la misma que la diferencia entre el mayor y menor valor de las continuas.
 - **Si la estandarización de las variables continuas se hace en media/varianza**, se moverán entre -3 y 3, pero con alta probabilidad entre -1 y 1, lo que se alinea con los valores de las discretas.

Tratamiento de variables: outliers

- Los **outliers** pueden dar lugar a **clusters aislados** que no tengan un sentido de negocio propio.
- Para amortiguar el efecto de los mismos, una posibilidad es aplicar alguna transformación a las variables, como por ejemplo el logaritmo.
- Por otra parte, es cierto que **prescindir** de la información que incorporan los **outliers** puede hacer **perder información valiosa** (por ejemplo en modelos de identificación de fraude).
- El análisis de outliers debe **realizarse una vez estandarizados los datos** ya que al realizar la estandarización, los valores extremos pueden perder influencia.
- La fase siguiente consiste en establecer el tipo de algoritmo de *clustering* a utilizar. Existen **métodos que son más o menos sensibles a la presencia de outliers** por lo que, dependiendo del que se seleccione, el tratamiento previo de éstos tendrá mayor o menor importancia.

Tratamiento de variables: redundancias

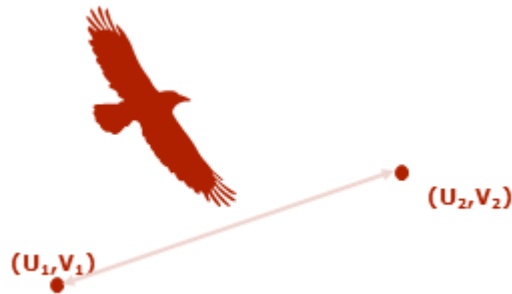
- Con la idea de que **todas las dimensiones de análisis tengan el mismo peso interesa eliminar posibles redundancias.** La eliminación de redundancias se puede hacer:
 - **De manera artesanal.-** mirando correlaciones entre variables (coeficiente de Pearson para continuas, correlaciones tipo rango (coeficientes de Spearman/Kendall) para ordinales). Inconveniente: procedimiento inviable si el número de variables es muy alto.
 - **Mediante un análisis de componentes principales.-** seleccionando un número de variables (incorreladas/ortogonales) que concentren un porcentaje de variabilidad suficientemente alto. El inconveniente de esta técnica es la **pérdida de interpretabilidad.**
 - **Haciendo previamente un cluster de variables.-** se puede hacer:
 - **Por negocio.-** grupos de variables relacionadas con una misma dimensión de negocio (sociodemográfica, ahorro/deuda en banca, siniestralidad en seguros, etc.).
 - **Analíticamente.-** mediante un clustering de variables.
 - De cada **grupo** de variables se selecciona un **representante:**
 - **Por negocio.-** una variable concreta, un valor medio ponderado por negocio, etc.
 - **Analíticamente.-** una media (ponderada o no), o una componente principal. Las componentes principales de los diferentes grupos son oblicuas (no ortogonales entre sí).

4 | Selección de la métrica de desemejanza

Selección de la métrica de desemejanza

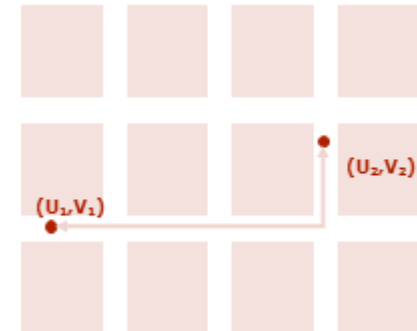
- Independientemente del método es preciso establecer **cómo se medirá la distancia entre los elementos**. El concepto de semejanza es un concepto muy interiorizado en nuestro pensamiento pero complicado de cuantificar. **«¿Qué es más similar a un pato: un pingüino o una perdiz?»**. Depende de la métrica seleccionada:
 - Si pensamos en la capacidad de volar asociaremos al pato con la perdiz
 - Si en la capacidad de nadar, asociaremos al pato con el pingüino.
- Algunas de las **métricas que habitualmente se utilizan** son:

Euclidea



$$d(u, v) = \sqrt{(u_1 - u_2)^2 + (v_1 - v_2)^2}$$

Manhattan (Taxi)



$$d(u, v) = |u_1 - u_2| + |v_1 - v_2|$$

Selección de la métrica de desemejanza

○ Existen **otras métricas basadas en criterios estadísticos**:

- **La distancia de Mahalanobis (1936).**- se recomienda utilizar esta métrica cuando existe correlación entre los datos dado que en su definición implica a la matriz de varianzas – covarianzas, la cual permite tener en cuenta la relación entre ellos.

Su definición se basa en dos consideraciones:

- **Es necesario eliminar la dependencia de las unidades de medida** dado que si no, unas variables pueden tener más peso que otras.- esto llevaría a dividir por la desviación típica

$$d(u, v) = \sqrt{(\vec{u}_1 - \vec{v}_1)^2 + (\vec{u}_2 - \vec{v}_2)^2} \Rightarrow d(u, v) = \sqrt{\left(\frac{\vec{u}_1 - \vec{v}_1}{\sigma_1}\right)^2 + \left(\frac{\vec{u}_2 - \vec{v}_2}{\sigma_2}\right)^2}$$

- Si las variables U y V no son independientes, es necesario incorporar el valor de σ_{12} en el cálculo. Para ello, en lugar de usar $\Sigma = \begin{pmatrix} \sigma_1^2 & 0 \\ 0 & \sigma_2^2 \end{pmatrix}$, se debe emplear $\Sigma = \begin{pmatrix} \sigma_1^2 & \sigma_{12} \\ \sigma_{12} & \sigma_2^2 \end{pmatrix}$.

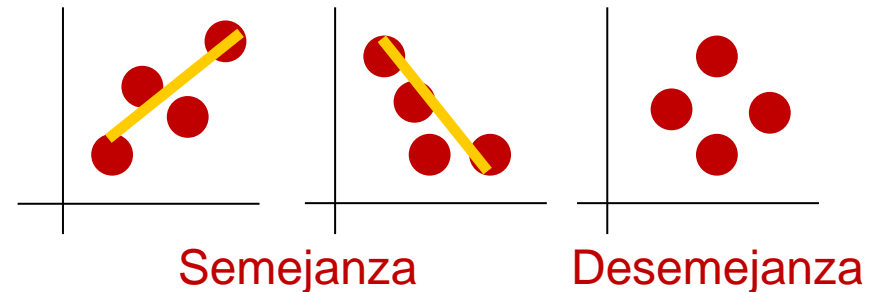
$$d(u, v) = \sqrt{(\vec{u} - \vec{v}) \Sigma^{-1} (\vec{u} - \vec{v})'}$$

Selección de la métrica de desemejanza

○ Existen **otras métricas basadas en criterios estadísticos**:

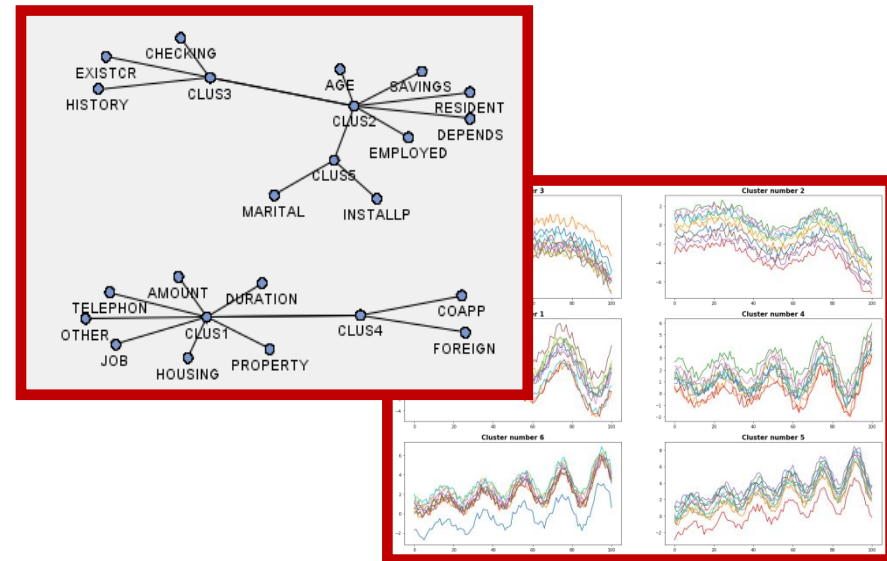
- **Correlación.-** cuando se hace un clustering de variables (en lugar de elementos), una manera habitual de medir distancia entre ellas es a través de correlaciones.

$$d(u, v) = \frac{\sum(u_i - \bar{u})(v_i - \bar{v})}{\sqrt{\sum(u_i - \bar{u})^2 \sum(v_i - \bar{v})^2}}$$



Este tipo de distancia puede ser útil:

- Para agrupar variables parecidas y seleccionar un representante de cada grupo con vistas a **eliminar redundancias entre variables**.
- En el contexto de **clustering de series temporales**.



Selección de la métrica de desemejanza

- Existen también **métricas** especialmente orientadas a realizar **segmentaciones con variables binarias**:

- **Índice de Rassel-Rao:** $d(O_i, O_j) = 1 - \frac{n_{IJ}}{n}$
- **Emparejamiento simple:** $d(O_i, O_j) = 1 - \frac{n_{IJ} + n_{ij}}{n}$
- **Coeficiente o índice de Jaccard:** $d(O_i, O_j) = 1 - \frac{n_{IJ}}{n_{IJ} + n_{Ij} + n_{iJ}}$

siendo:

n_{IJ} : número de variables para las que en O_i y O_j hay 1's a la vez (1,1).

n_{ij} : número de variables para las que en O_i y O_j hay 0's a la vez (0,0).

n_{ij} : número de variables para las que en O_i y O_j hay (1,0).

n_{iJ} : número de variables para las que en O_i y O_j hay (0,1).

N : número variables = $n_{IJ} + n_{ij} + n_{ij} + n_{iJ}$

5 | Selección del tipo de algoritmo

Algoritmos de clasificación

- Existen **diferentes tipos de algoritmos**. Algunos de los más conocidos son:

Tipo de Algoritmo	Nombre
PARTICIONAL basados en procesos iterativos que buscan una agrupación óptima (normalmente local) respecto de una función objetivo	K-MEANS
	PAM (K-MEDIODS)
	CLARA
	CLARANS
JERÁRQUICO basados en la construcción de una jerarquía entre los grupos	AGLOMERATIVOS
	DIVISIVOS
	HDBSCAN
BASADO EN DENSIDAD basados en la ubicación espacial y en la distancia a un número de vecinos especificado	DBSCAN
	GMM

5.1 | Métodos particionales

Métodos particionales

- Los **procedimientos particionales** surgen como **alternativa a los jerárquicos** por los problemas computacionales que éstos presentan.
- **Asignan los elementos a clusters una vez que el número de éstos a considerar ha sido especificado.** Así, la solución de k clusters no es solo una combinación de 2 grupos que parte de una solución de $k+1$ clusters sino que se basa en la búsqueda de la mejor solución (la óptima) de k clusters.

- Generalmente **proporciona mínimos locales** (no globales) respecto a SCDG:

$$W(K) = \sum_{g=1}^G W_g = \sum_{g=1}^G \sum_{i=1}^{n_g} \sum_{j=1}^p (x_{ijg} - \bar{x}_{jg})^2$$

- Esta cantidad es la suma de cuadrados dentro de los grupos (SCDG) y mide la distancia de las observaciones de cada grupo a su **centroide: punto respecto al cual la suma de las distancias de todas las observaciones se minimiza.**
- Los algoritmos responden a la siguiente **estructura**:
 - Parten de **k centroides iniciales.**
 - Calculan la distancia de los n individuos a los k centroides y realizan la **asignación individuo-cluster** (al más próximo): nueva partición y nuevo SCDG.
 - Establece un **criterio de convergencia** repitiendo el paso anterior hasta que se alcance.
- Su complejidad es $O(n \cdot p \cdot k \cdot i)$ donde n es el n° de observaciones, p el n° de variables, k el n° de grupos e i el n° de iteraciones necesarias.

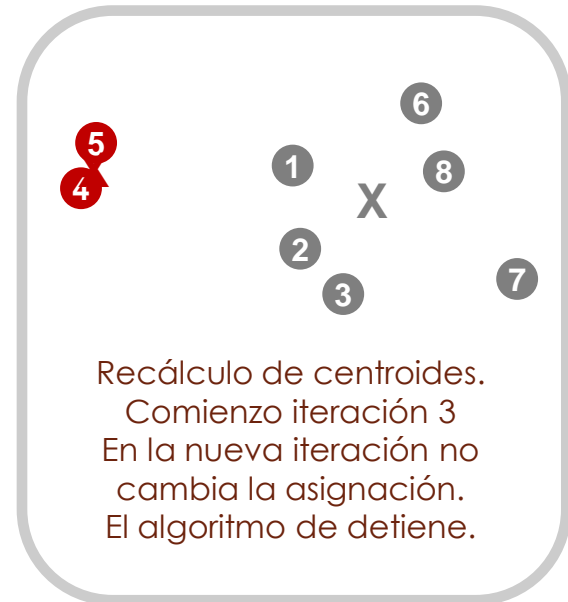
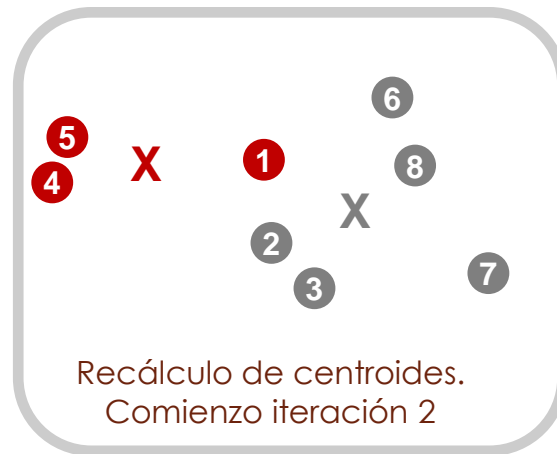
Métodos particionales: Método de Forgy

La **técnica más conocida de cluster de optimización es el algoritmo de las k-medias** que presenta muchas variantes:

Método de Forgy (1965).

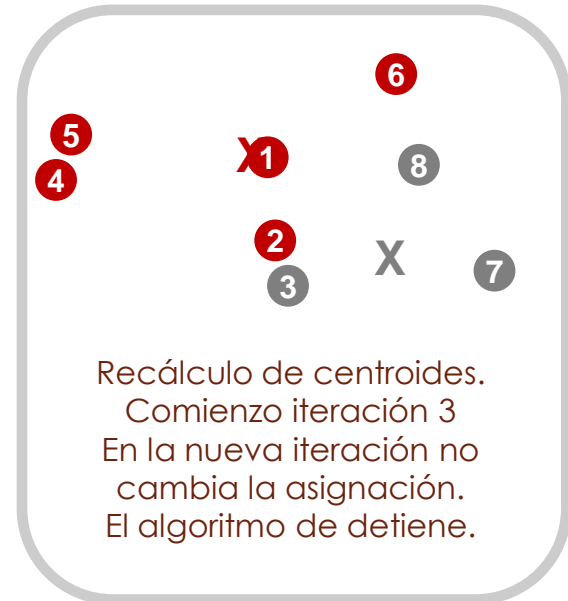
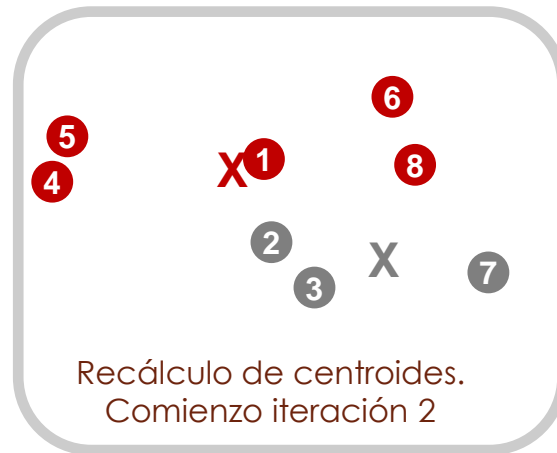
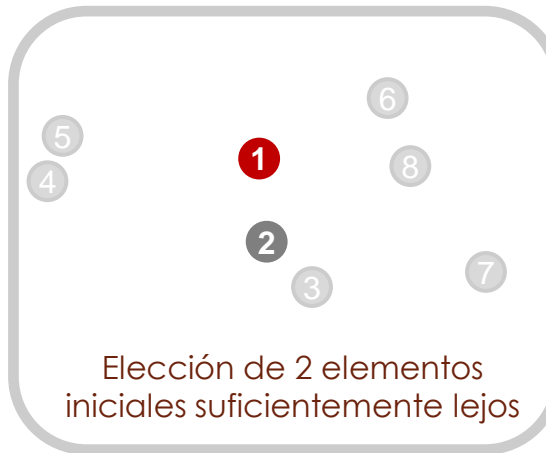
1. Tomar k centroides de partida (existen diferentes métodos).
2. Colocar cada individuo en el cluster con la semilla más próxima. Las semillas permanecen fijas para cada ciclo completo (un ciclo consiste en recorrer todo el conjunto de datos).
3. Calcular los nuevos centroides (medias de los datos).
4. Alternar los pasos 2 y 3 hasta que se alcance un número máximo de iteraciones o el cambio en la función objetivo sea menor que un α establecido de antemano. Empíricamente se ha probado que n° ciclos suele ser menor que 10.

Métodos particionales: Método de *Forgy*



El algoritmo es sensible a la elección de los
centroides de partida.

Métodos particionales: Método de Forgy



El algoritmo es sensible a la elección de los
centroides de partida.

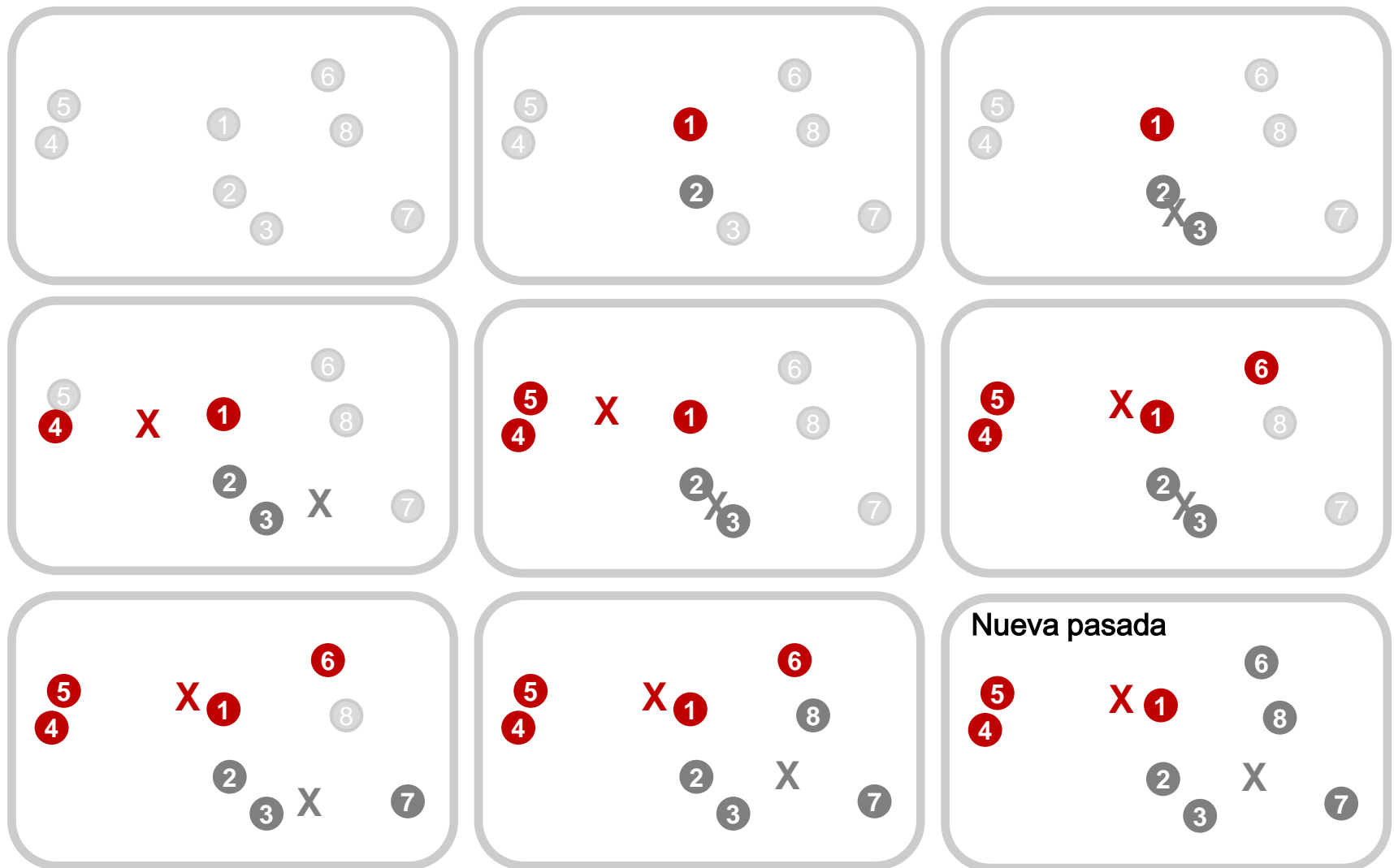
Métodos particionales: Método de MacQueen

La **técnica más conocida de cluster de optimización es el algoritmo de las k-medias** que presenta muchas variantes:

Método de MacQueen (1965).

1. Tomar k centroides de partida por alguno de los criterios anteriores (en su algoritmo original, *MacQueen* proponía tomar los k primeros).
2. Asignar cada uno de los $n - k$ elementos restantes al centroide más próximo. Tras cada asignación (no tras cada ciclo) recalcular el centroide del cluster obtenido.
3. Tomar los centroides de los *clusters* existentes y hacer una nueva pasada sobre los datos asignándolos al centroide más cercano.
4. El algoritmo original de *MacQueen* solo hace una iteración pero se puede iterar los pasos 2 y 3 hasta conseguir convergencia.

Métodos particionales: Método de MacQueen



Caso I (Algoritmo original de MacQueen): Se toma como centroides iniciales los dos primeros elementos. En los recuadros se presenta el resultado de hacer primero la asignación del siguiente elemento numerado y, posteriormente, recalculer el centroide.

Métodos particionales: Método de MacQueen

○ Algunas otros métodos particionales son:

- **PAM** (*Partitioning Around Mediods*).- En este caso los centroides son observaciones del dataset (no las medias de cada cluster). El objetivo es encontrar observaciones representativas del conjunto de datos con la menor disimilaridad posible con las demás observaciones del propio cluster.

Se trata de un algoritmo más robusto que el K-MEANS pero de mayor coste computacional. **Su complejidad es $O(k*(n-k)^2)$.**

CLARA (*Clustering Large Applcations*).- Es una versión más eficiente computacionalmente del PAM. En cada iteración se seleccionan grupos aleatorios de tamaño $40+2*k$ y sobre cada uno de ellos aplica el PAM. **Su complejidad es $O(k*(40+k)^2+k*(n-k))$.**

CLARANS (*Clustering Large Applications Based on Randomized Search*).- Es una versión que aumenta su complejidad respecto a K-MEANS y CLARA, pero funciona bien para bases de datos de grandes dimensiones. **Su complejidad es $O(k*n^2)$.**

Algoritmo *fuzzy k-means* (*c-means*)

- Sea uno u otro el algoritmo utilizado, como resultado del mismo los elementos quedan asignados a un grupo.
- El algoritmo K-MEANS asigna cada observación a un grupo, pero en función de la distancia del elemento al centroide de dicho grupo se puede considerar que pertenece más o menos a él.
- En terminología difusa, la inversa de la distancia del elementos al centroide del grupo define el **grado de pertenencia** de dicho elemento a dicho grupo.
- Dichos valores pueden ser normalizados para que sumen 1 y definir algo parecido a la probabilidad de que el elemento pertenezca a cada grupo.
- En este contexto, surge el algoritmo **FUZZY K-MEANS** como una variante del algoritmo K-MEANS enfocada bajo la **óptica difusa**.
- En este caso, la función objetivo a minimizar es la suma de cuadrados intragrupal **ponderando la distancia al centroide de cada cluster por su grado de pertenencia al mismo**:

$$W(K) = \sum_{g=1}^G W_g = \sum_{g=1}^G \sum_{i=1}^{n_g} \sum_{j=1}^p (\mu_g(x_{ijg}))^m (x_{ijg} - \bar{x}_{jg})^2 \quad \text{con} \quad \sum_{g=1}^G \mu_g(x_{ijg}) = 1 \quad \forall x_{ijg}$$

Métodos particionales: Método de Bezdek

Método de Bezdek (1981).

1. Tomar c centroides de partida por alguno de los criterios anteriores.
2. Calcular la matriz de distancias a los centroides: $d_{ji} = ||x_j - v_i||^2$ y la matriz de pertenencias a ellos:
$$\mu_{ji} = \frac{1}{\sum_{l=1}^c \left(\frac{||x_j - v_l||^2}{||x_j - v_i||^2} \right)^{\frac{1}{m-1}}}$$

La función objetivo en la iteración t es: $J^{(t)} = \sum_{i=1}^c \sum_{j=1}^n \mu_{ji}^m d_{ji}$

3. Clasificar cada elemento en el clúster cuyo grado de pertenencia sea mayor y recalcular los

$$\text{centroides: } v_i = \frac{\sum_{j=1}^n (\mu_{ji}(x_j))^m x_j}{\sum_{j=1}^n (\mu_{ji}(x_j))^m} \quad 1 \leq i \leq c$$

4. Alternar los pasos 2 y 3 hasta que se alcance un número máximo de iteraciones o el cambio en la función objetivo sea menor que un α establecido de antemano ($|J(t) - J(t-1)| < \alpha$).

Algoritmo *fuzzy k-means* (*c-means*)

- El **parámetro de fuzzyficación** toma valores en el intervalo $(1, \infty)$, y permite al usuario elegir cómo de “soft” quiere que sea la clasificación. Cuanto más próximo a 1, más “hard” es la clasificación que se realiza:
- Este parámetro juega un papel muy importante en el algoritmo, condicionando el cálculo de los centroides y de los grados de pertenencia y, en consecuencia, la solución final:
 - En el **caso extremo ($m=1$)** tendríamos la función objetivo que plantea minimizar el ***k-means***.
 - **En la mayoría de las aplicaciones su valor suele ser 2.**
 - Cuando **m es demasiado grande**, los grados de pertenencia μ_i convergen a $1/c$ $\forall x \in X$, lo que vendría a significar que todos los elementos pertenecen con igual probabilidad a todos los clústers (**máxima confusión entre clusters**).
- El algoritmo *fuzzy k-means* **suele proporcionar mejores resultados en aquellos casos en los que hay outliers o mucho ruido**. En dichos casos los clusters tienden a solaparse (*overlapping clusters*) y no puede decidirse fácilmente si un objeto pertenece a uno u otro cluster.

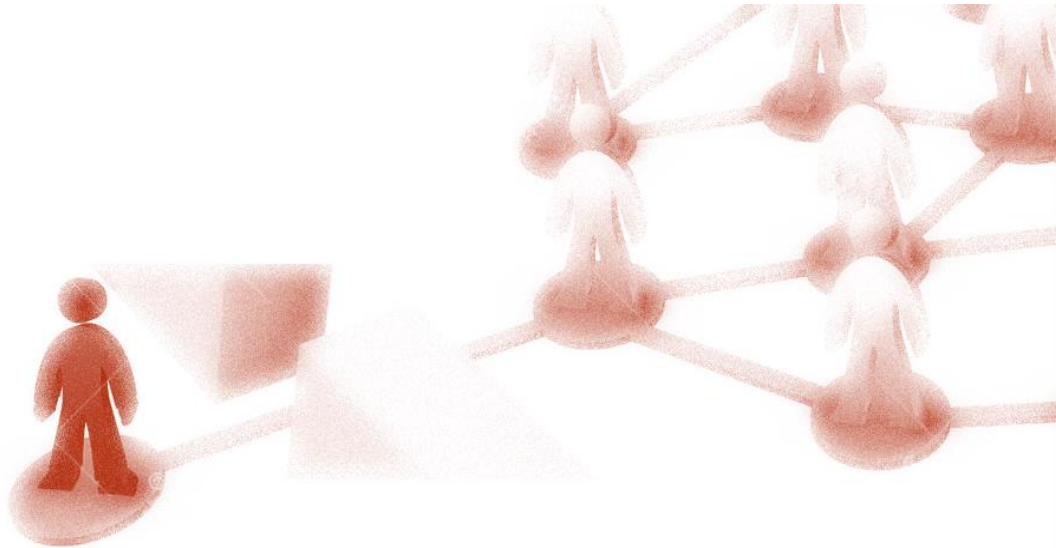
5.2 | Métodos jerárquicos

Métodos jerárquicos

- Los métodos jerárquicos **parten de una matriz de distancias** entre los elementos y construyen una jerarquía. Los resultados obtenidos en cada iteración quedan encajados en los resultados de la iteración siguiente generando una **estructura de árbol (dendrograma)** en el que es posible rastrear el proceso de agrupación o desagrupación desde su origen.
- Existen dos tipos:
 - **Métodos de aglomeración o agregativos.-** parten de tantos clusters como elementos y, en cada iteración, realizan una agrupación (de dos elementos, de un elemento con un grupo o de dos grupos). Son los más habituales.
 - **Métodos de división o disgregativos.-** parten del conjunto formado por todos los elementos y, en cada iteración, realizan una separación dando lugar a clusters más pequeños.
- Tener visibilidad de un resultado con $1, 2, \dots, n$ clusters **beneficia la toma de una decisión en cuanto al número de clusters a considerar.**
- Una consecuencia negativa es el **alto coste computacional** en el que incurren.

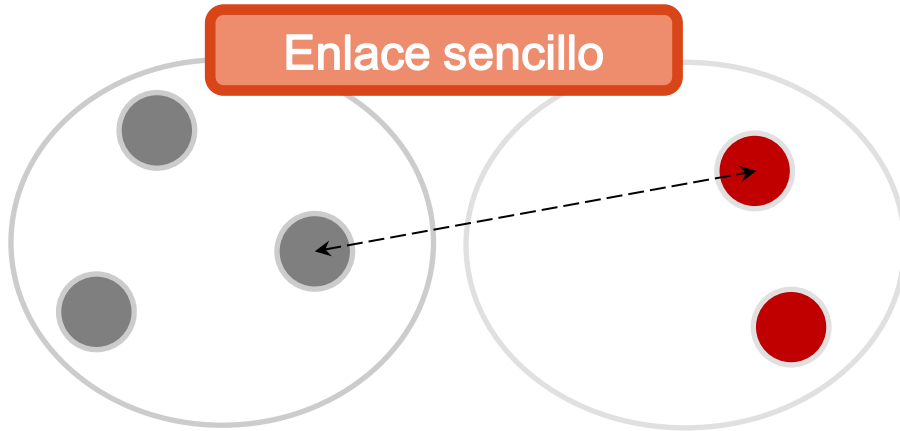
Métodos jerárquicos

- La matriz de distancias de partida informa sobre el grado de proximidad entre los elementos de partida pero conforme se van configurando los grupos, es preciso **definir la distancia de un elemento a un grupo o la de dos grupos entre sí.**
- En los **métodos particionales**, la distancia de un elemento a un grupo es la distancia del elemento al punto medio del grupo (**distancia centroide**). En los jerárquicos, además de ésta hay que **definir la distancia entre grupos.**



Métodos jerárquicos

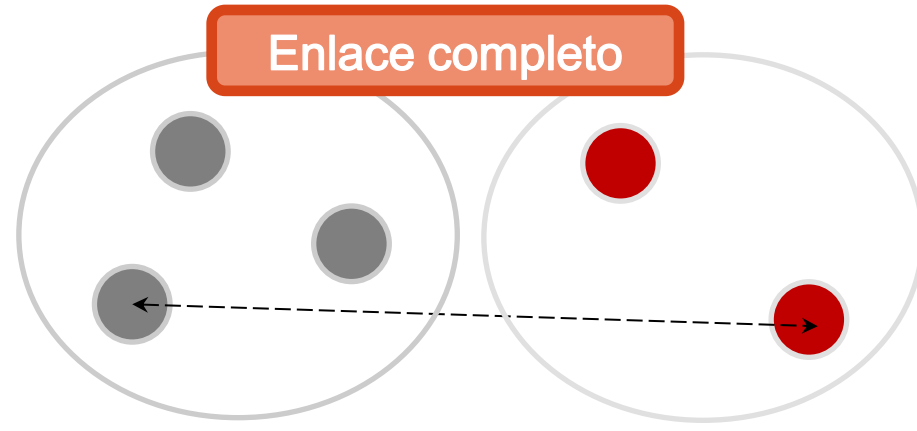
Enlace sencillo



☹️ Sensible a *outliers*

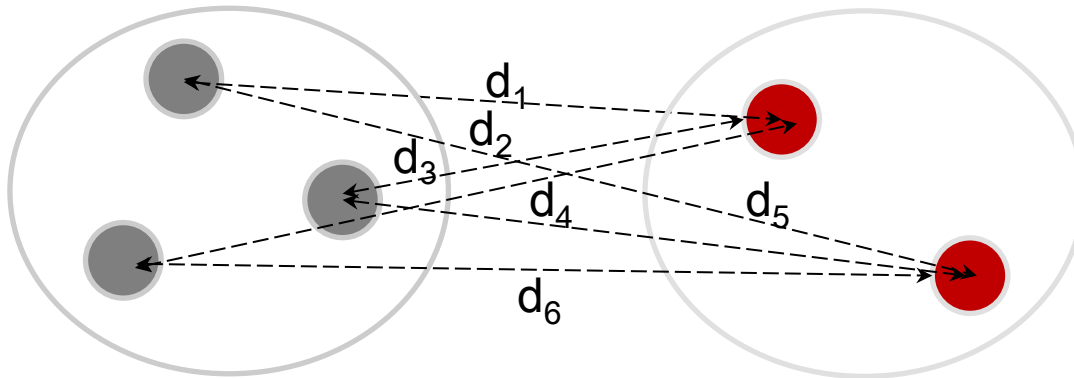
☹️ Tiende a generar *clusters* elongados (encadenados)

Enlace completo



☹️ Sensible a *outliers*

😊 Permite construir esferas de radio mínimo



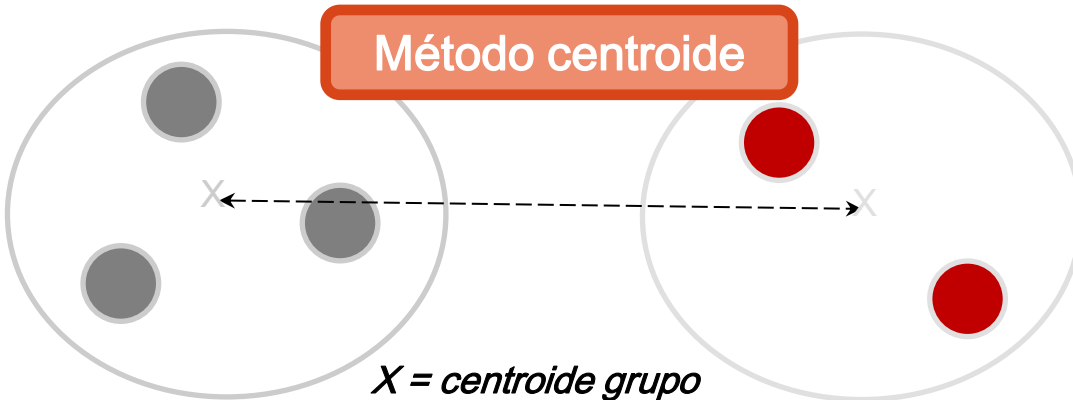
Enlace promedio

$$d(A, B) = \frac{d_1 + d_2 + d_3 + d_4 + d_5 + d_6}{6}$$

😊 Robustez a *outliers*

😊 Eficiente computacionalmente

Métodos jerárquicos. Enlaces habituales



$X = \text{centroide grupo}$

😊 Robustez a *outliers*

😊 Eficiente computacionalmente

Método WARD

Agrupando buscando el mínimo incremento de varianza.

$$W(G) = \sum_{g=1}^G W_g = \sum_{g=1}^G \sum_{i=1}^{n_g} \sum_{j=1}^p (x_{ijg} - \bar{x}_{jg})^2$$

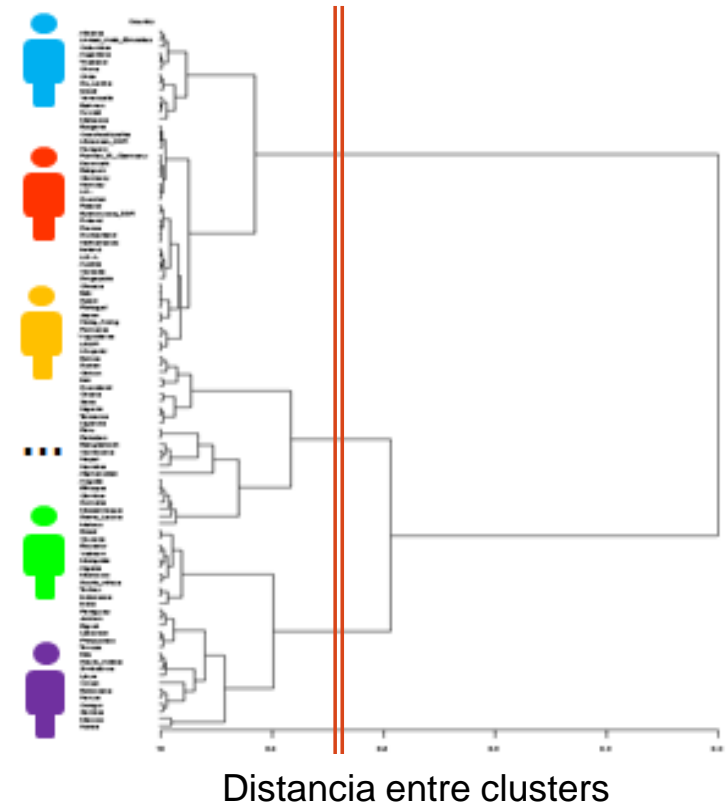
😞 Sensible a *outliers*

😞 Poco eficiente computacionalmente

😊 *Kaiper* y *Fisher* probaron que este método era capaz de acercarse más a la clasificación óptima que otros métodos

Métodos jerárquicos

- El **dendrograma** es un grafo conexo sin ciclos con un punto llamado raíz y tantos puntos extremos como elementos que equidistan de la raíz.
- Permite la **representación gráfica del proceso de agrupamiento** en forma de árbol, quedando las observaciones en uno de los ejes y algún **tipo de métrica** (R-square, varianza intragrupo, etc.) en otro.
- Al realizar un corte perpendicular al eje en el que se mide la métrica es posible conocer el valor de la misma para cierto nivel de agrupación, lo que proporciona una **herramienta** de ayuda para **decidir el número de clusters** a considerar.
- La decisión se basa en que el valor de la métrica no varíe sensiblemente al aumentar en una unidad el n° de *clusters*.



Ajustando 3 clusters (número de cortes), se puede conocer el valor de la métrica.

Una distancia grande es orientativa de ganancia y por tanto, del número de clusters a considerar.

Enlace sencillo

O_1	O_2	O_3	O_4	O_5	
	1	3	5	2	O_1
		3	3	4	O_2
			6	7	O_3
				9	O_4
					O_5

$$\Rightarrow r \equiv \{O_1, O_2\}$$

$$\left. \begin{aligned} d(r, O_3) &= \min\{d(O_1, O_3), d(O_2, O_3)\} = \min\{3, 3\} = 3 \\ d(r, O_4) &= \min\{d(O_1, O_4), d(O_2, O_4)\} = \min\{5, 3\} = 3 \\ d(r, O_5) &= \min\{d(O_1, O_5), d(O_2, O_5)\} = \min\{2, 4\} = 2 \end{aligned} \right\} \Rightarrow r'' \equiv \{r, O_5\}$$

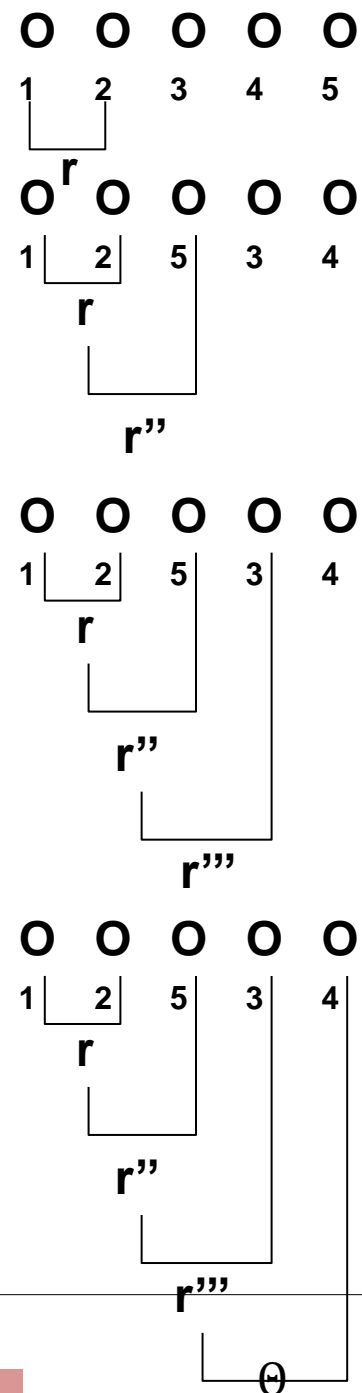
r	O_3	O_4	O_5	
	3	3	2	r
		6	7	O_3
			9	O_4
				O_5

$$\left. \begin{aligned} d(r'', O_3) &= \min\{d(r, O_3), d(O_5, O_3)\} = \min\{3, 7\} = 3 \\ d(r'', O_4) &= \min\{d(r, O_4), d(O_5, O_4)\} = \min\{3, 9\} = 3 \end{aligned} \right\} \Rightarrow r''' \equiv \{r'', O_3\}$$

r''	O_3	O_4	
	3	3	r''
		6	O_3
			O_4

$$d(r''', O_4) = \min\{d(r'', O_4), d(O_3, O_4)\} = \min\{3, 6\} = 3 \Rightarrow \Theta \equiv \{r''', O_4\}$$

r'''	O_3	
	3	r'''
		O_4



Enlace diámetro

O_1	O_2	O_3	O_4	O_5	
	1	3	5	2	O_1
		3	3	4	O_2
			6	7	O_3
				9	O_4
					O_5

$$\Rightarrow r \equiv \{O_1, O_2\}$$

$$\left. \begin{aligned} d(r, O_3) &= \max\{d(O_1, O_3), d(O_2, O_3)\} = \max\{3, 3\} = 3 \\ d(r, O_4) &= \max\{d(O_1, O_4), d(O_2, O_4)\} = \max\{5, 3\} = 5 \\ d(r, O_5) &= \max\{d(O_1, O_5), d(O_2, O_5)\} = \max\{2, 4\} = 4 \end{aligned} \right\} \Rightarrow r'' \equiv \{r, O_3\}$$

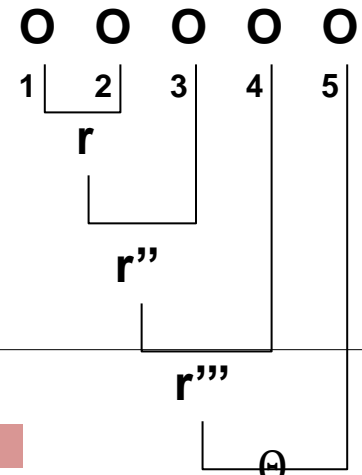
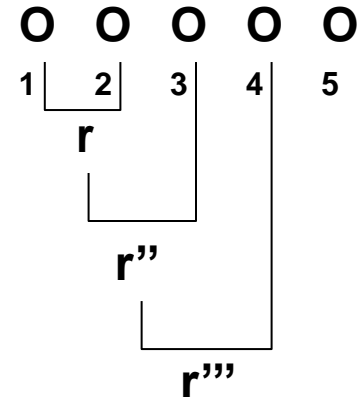
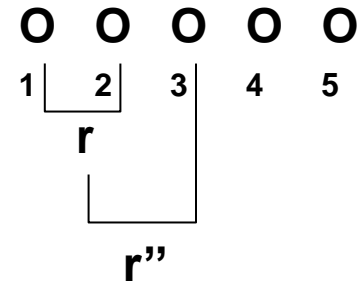
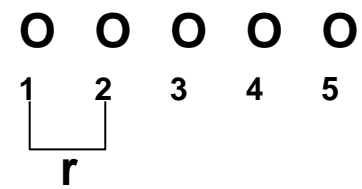
r	O_3	O_4	O_5	
	3	5	4	r
		6	7	O_3
			9	O_4
				O_5

$$\left. \begin{aligned} d(r'', O_4) &= \max\{d(r, O_4), d(O_3, O_4)\} = \max\{5, 6\} = 6 \\ d(r'', O_5) &= \max\{d(r, O_5), d(O_3, O_5)\} = \max\{4, 7\} = 7 \end{aligned} \right\} \Rightarrow r''' \equiv \{r'', O_4\}$$

r''	O_4	O_5	
	6	7	r''
		9	O_4
			O_5

$$d(r''', O_5) = \max\{d(r'', O_5), d(O_4, O_5)\} = \max\{7, 9\} = 9 \Rightarrow \Theta \equiv \{r''', O_5\}$$

r'''	O_5	
	3	r'''
		O_5



Enlace promedio

O_1	O_2	O_3	O_4	O_5	
	1	3	5	2	O_1
		3	3	4	O_2
			6	7	O_3
				9	O_4
					O_5

$$\Rightarrow r \equiv \{O_1, O_2\}$$

$$\left. \begin{aligned} d(r, O_3) &= \text{med}\{d(O_1, O_3), d(O_2, O_3)\} = \text{med}\{3, 3\} = 3 \\ d(r, O_4) &= \text{med}\{d(O_1, O_4), d(O_2, O_4)\} = \text{med}\{5, 3\} = 4 \\ d(r, O_5) &= \text{med}\{d(O_1, O_5), d(O_2, O_5)\} = \text{med}\{2, 4\} = 3 \end{aligned} \right\} \Rightarrow r'' \equiv \{r, O_3\}$$

r	O_3	O_4	O_5	
	3	4	3	r
		6	7	O_3
			9	O_4
				O_5

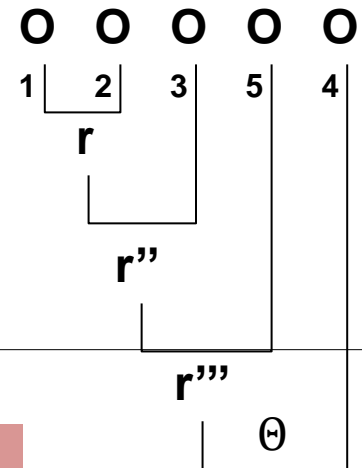
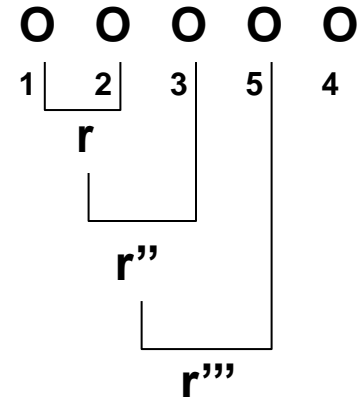
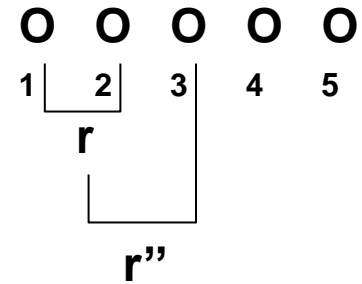
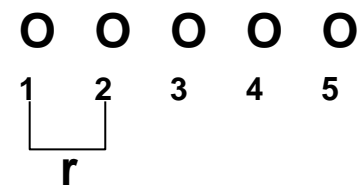
$$\left. \begin{aligned} d(r'', O_4) &= \text{med}\{d(O_1, O_4), d(O_2, O_4), d(O_3, O_4)\} = \text{med}\{5, 3, 6\} = 14/3 \\ d(r'', O_5) &= \text{med}\{d(O_1, O_5), d(O_2, O_5), d(O_3, O_5)\} = \text{med}\{2, 4, 7\} = 13/3 \end{aligned} \right\} \Rightarrow r''' \equiv \{r'', O_5\}$$

r''	O_4	O_5	
	14/3	13/3	r''
		9	O_4
			O_5

$$d(r''', O_4) = \frac{d(O_1, O_4) + d(O_2, O_4) + d(O_3, O_4) + d(O_5, O_4)}{4} = \frac{5 + 3 + 6 + 9}{4} = \frac{23}{4}$$

r'''	O_4	
	23/4	r'''
		O_4

$$\Rightarrow \Theta \equiv \{r''', O_4\}$$



Algunas métricas para decidir k

- Algunas **métricas** habituales para **decidir el número de clusters** son:
 - Utilizar un gráfico de codo/rodilla (**elbow / knee method**) evalúa la varianza intracluster $W(k)$ como una función del número k de clusters. Se busca un codo en el gráfico, dando a entender que, la consideración de un cluster adicional no mejora mucho la segmentación. Sin embargo, en ocasiones se trata de un criterio subjetivo.
 - En esta línea, $H(K) = (n - K - 1) \left[\frac{W(K)}{W(K+1)} - 1 \right]$. – permite evaluar cómo se reduce W al pasar de k a $k+1$ grupos (el numerador es positivo). Se compara la disminución de variabilidad al aumentar un grupo. Es un estadístico F parcial para contrastar si tiene valor hacer $K+1$ clusters. **Regla empírica (Hartigan, 1975): hacer $K+1$ grupos si esta cantidad es mayor que 10.**
 - El método de **Calinski y Harabasz, (1974)**, consiste en elegir el número de clusters que maximiza $CH(K) = \frac{B(K)/(K-1)}{W(K)/(n-K)}$ (como el estadístico F de un ANOVA).
 - $KL(K) = \left| \frac{DIFF(K)}{DIFF(K+1)} \right|$, $DIFF(K) = (K-1)^{2/p} W(K-1) - K^{2/p} W(K)$ (**Krzanowski y Lai, 1985**).- consiste en elegir el número de clusters que maximiza esta cantidad.

$W(K)$ = varianza intra-cluster (SCDG), $B(K)$ = varianza inter-cluster, p = nº variables, n = nº datos

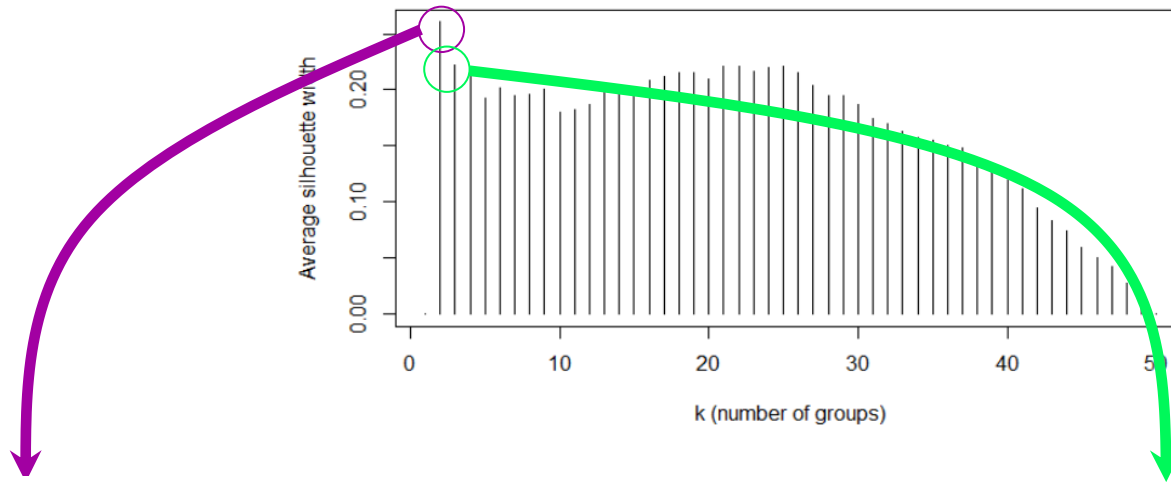
Algunas métricas para decidir k

- El **coeficiente de silueta (Silhouette) o puntuación** de silueta es otra métrica utilizada para evaluar la bondad del agrupamiento, determinando **lo bien que está clasificada cada observación dentro de su cluster**.
- Dada una observación "i", se define $s(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}}$ siendo:
 - $a(i)$ = *disimilaridad media entre i y todos los puntos de cluster al que pertenece.*
 - $d(i, C)$ = la disimilaridad de "i" a todas las observaciones de C
 - $b(i) = \min_C d(i, C)$ *disimilaridad media ente "i" y el cluster más cercano al que no pertenece*
 - Si "i" es la única observación de su cluster, $s(i) = 0$.
- $s(i)$ varía entre -1 y 1:

{	Valor próximo a 1: Obs. "i" bien clusterizada
	Valor próximo a 0: Obs. "i" entre dos clusters
	Valor próximo a -1: Obs. "i" probablemente mal clasificada.
- En cada cluster, se promedian los $s(i)$ de las observaciones que lo componen y además se calcula un valor promedio asociado a la segmentación de forma que:
 - Valor 1: Los clusters están bien separados entre sí y se distinguen claramente.
 - 0: Los clusters son indiferentes, e.d, la distancia entre ellos no es significativa.
 - -1: Los clusters están asignados de forma incorrecta.

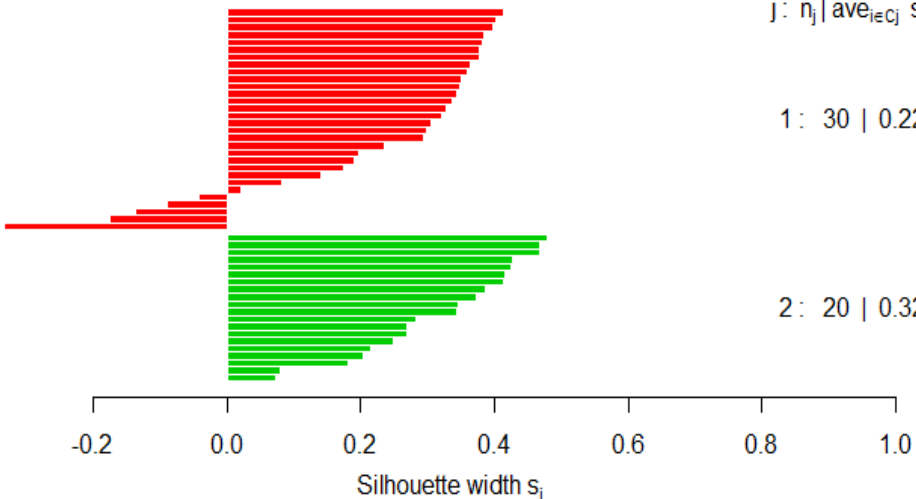
Algunas métricas para decidir k

Silhouette-optimal number of clusters



Silhouette plot

n = 50

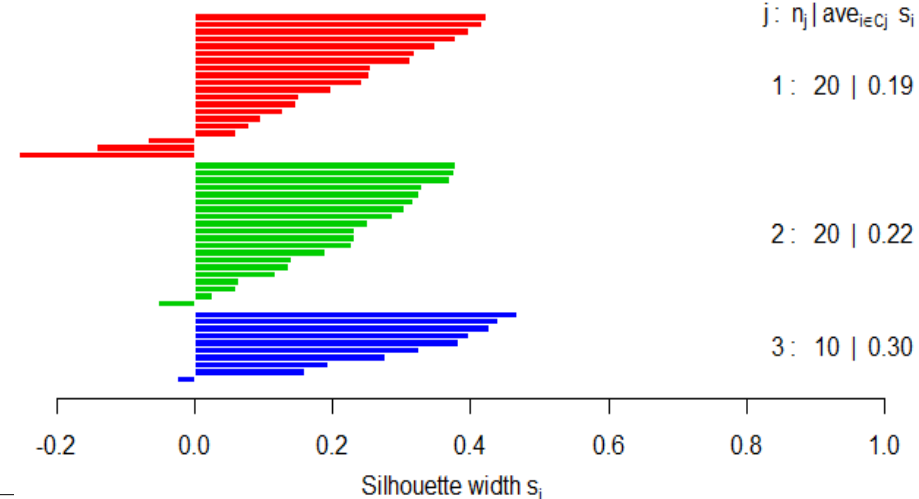


Average silhouette width : 0.26



Silhouette plot

n = 50



Average silhouette width : 0.22

Métodos bietápicos: Jerárquico + Particional

- Los **algoritmos de clustering bietápicos** tratan de solucionar el problema que supone establecer de antemano el número de clusters a generar.
- La estructura a la que responden dichos algoritmos son:
 - **Aplicar un algoritmo tipo k-medias con un número de clusters suficientemente alto.**
 - Calcular los centroides de los grupos obtenidos y considerar **un nuevo problema de clasificación en el que los elementos a agrupar son dichos centroides.**
 - La manejabilidad computacional del nuevo problema mediante un **algoritmo jerárquico** permiten utilizar una técnica de este tipo que ayuda a **decidir el número K de clusters a considerar** (mediante el criterio CCC por ejemplo).
 - **Aplicar el algoritmo k-means sobre el conjunto de datos original solicitando K clusters.** Como punto de partida se pueden utilizar los K centroides obtenidos con el algoritmo jerárquico.

5.3 | Métodos basados en densidades

Métodos basados en densidades. DBSCAN

- La **técnica más conocida de cluster basadas en densidad** es el **algoritmo DBSCAN** (*Density-Based Spatial Clustering of Applications with Noise*), debido a **Martin Ester, Hans-Peter Kregel, Jörg Sander and Xiaowei Xu (1996)**
- El **algoritmo DBSCAN depende de dos parámetros**:
 - ϵ .- radio que determina el tamaño de un vecindario.
 - min_points.- n° mínimo de puntos en el radio para conformar un cluster.

Nota: Se tiene en cuenta el propio punto al contabilizar dicho valor.
- La idea que subyace bajo el algoritmo es la siguiente:
 1. Para cada punto, construir un vecindario de radio ϵ y definir dicho punto como **CORE POINT** si en dicho vecindario hay al menos min_points puntos.
 2. Encontrar las “componentes conexas (conectadas)” de los **CORE POINTS** (CORE POINTS CONECTADOS). Dichas componentes constituirán los clusters.
 3. Tomar cada **NON-CORE POINT** y medir la distancia al cluster más cercano:
 - Si es menor que ϵ , asignarlo a dicho cluster, pasando a denominarse **BORDER_POINT**.
 - Si no, considerarlo como un **NOISE POINT**.

Métodos basados en densidades. DBSCAN

ALGORITMO DBSCAN

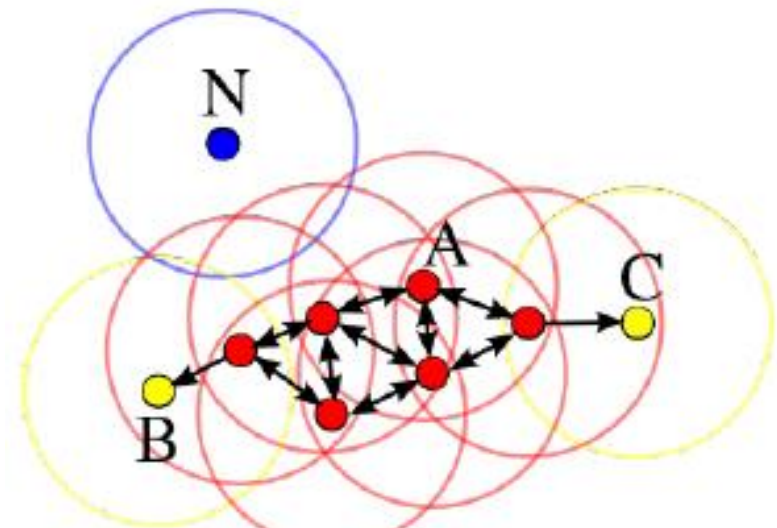
1. Elegir aleatoriamente un punto que no haya sido examinado y construir un vecindario de radio ϵ .
2. Si **no forma parte de ningún cluster**:
 - Si en el vecindario construido **hay al menos *min_points* puntos**, el punto pasa a ser denominado **CORE POINT** y se forma un cluster con ellos.
 - Si en el vecindario construido **no hay al menos *min_points* puntos**, el punto pasa a ser denominado **OUTLIER o NOISE (punto ruidoso)**.

Volver a 1.
3. Si **dicho punto forma parte ya de un cluster** generado anteriormente, pasará a calificarse de **BORDER POINT** si en su vecindario **no hay al menos *min_points* puntos**. **Volver a 1.**

Métodos basados en densidades. DBSCAN

Ejemplo con $min_samples = 4$.

- Todos los puntos rojos contienen al menos 3 puntos en el radio establecido, formando un cluster.
- Los puntos B y C son **alcanzables (*)** a través de ellos y por ello, aún cuando no tienen al menos min_points puntos en un radio suyo, forman parte de dicho cluster.
- El punto N es un nodo ruidoso porque no es alcanzable desde ningún punto.



- Un punto q es **directamente alcanzable** desde un CORE POINT p si la distancia entre ellos es menor que ϵ .
- En general, un punto q es **alcanzable** desde un CORE POINT p, si existe entre ellos un camino $p = p_1 = p_2 = \dots = p_n = q$ donde p_{i+1} es directamente alcanzable desde p_i . Esto implica que, salvo q (que puede ser un BORDER POINT), todos los puntos del camino son CORE POINTS.
- Todos los puntos **no alcanzables** desde otro punto son **NOISE POINTS**.

Métodos basados en densidades. DBSCAN

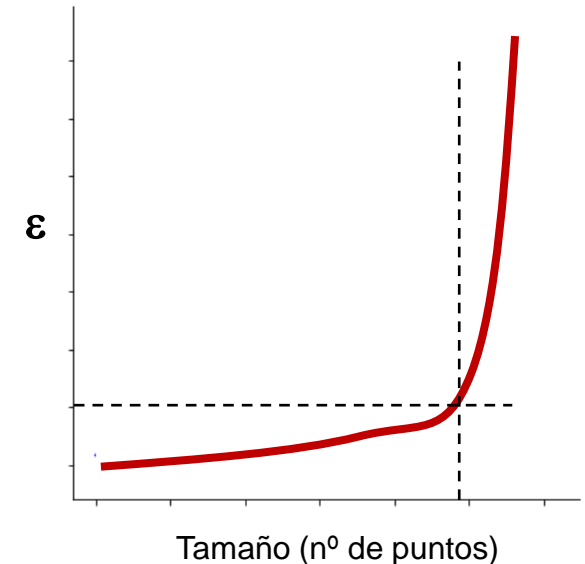
- Es preciso **determinar el valor de los parámetros ϵ y *min_points*.**
- Valores altos de ϵ hará que muchos datos estén en un mismo cluster, por lo que en general, **lo ideal es tomar valores pequeños para dicho parámetro**, teniendo en cuenta que valores demasiado bajos generará demasiados puntos ruidosos (no clusterizados).
- Respecto al parámetro ***min_points***, su valor debe ser al menos 3:
 - No tiene sentido que valga 1 (tantos clusters como puntos).
 - También se puede demostrar que si ***min_points* es menor o igual que 2, el resultado obtenido es el mismo que el que se obtendría con un cluster jerárquico con enlace sencillo.**
 - En general, si se trata de un dataset grande o que tenga muchos valores duplicados o que tenga datos “ruidosos”, interesa tomar un valor grande para él.
 - **OPTICS** (Partitioning Around Medoids).- Es una generalización de DBSCAN en la que el parámetro *min_points* es reemplazado por un valor que representa el mínimo tamaño que puede tener un cluster.

Métodos basados en densidades. DBSCAN

○ Un método para determinar dichos valores es el siguiente:

- **Se elige primer el valor de ϵ .** Para ello:

- **Se seleccionan diferentes tamaños de vecindades y se calcula el mínimo radio ϵ** que debería ponerse para que al construir vecindades de dicho radio se garantizase **alcanzar al menos dichos tamaños.**
- Se construye un **gráfico** que represente para cada uno de esos tamaños, dicho ϵ , y se busca en dicho gráfico una inflexión (**elbow/knee**): ¿A partir de qué valor al aumentar ϵ no se consigue que los clusters capturen muchos más vecinos?



- **Fijado ϵ , se elige el valor de min_points .** Para ello:

- Se va haciendo variar el valor de min_points y, en función de él se irán obteniendo diferente número de clusters. Utilizar alguna de las métricas ya conocidas (Calinski – Harabasz, silhouette, etc.) para tomar la decisión final.
- Si se quiere evitar el coste computacional de este último paso, como regla empírica se puede tomar $min_points = 2 * \text{número_variables}$.

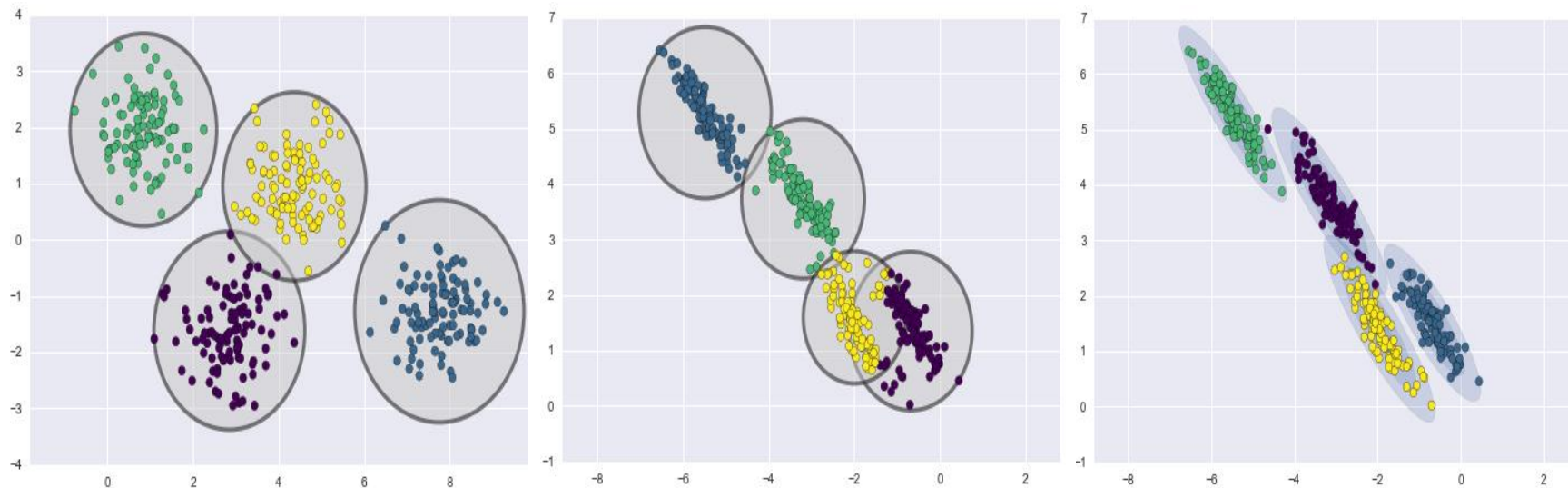
Métodos basados en densidades. DBSCAN

- El **principal inconveniente** del método es que si existen grandes diferencias de densidades en el dataset, la elección de los parámetros puede no ser adecuada.
- Sin embargo, el **método presente muchas ventajas** como son por ejemplo:
 - **No precisa de la determinación previa de un número de clusters.**
 - **No tiene una complejidad alta.** Es del orden **$O(n \cdot \log(n))$** .
 - **Es robusto a outliers.**
 - Aún cuando el orden en el que los datos son procesados puede influir en el resultado final (**no** es un método **completamente determinístico**), los **resultados no suelen ser muy diferentes.**

HDBSCAN (*Hierarchical Density-Based Spatial Clustering of Applications with Noise*).- es una **versión jerárquica** de DBSCAN en la que no existe la noción de puntos fronterizos, tratando éstos como NOISE POINT, lo que genera una **solución completamente determinística.**

Métodos basados en densidades. GMM

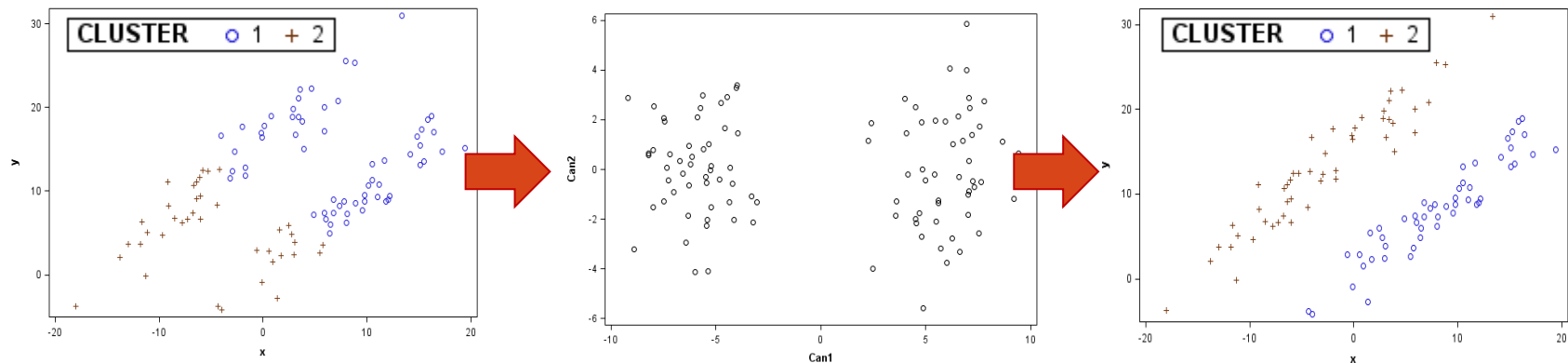
- En el método visto, los **radios contruidos alrededor de cada punto son circulares**, tendiendo a la **generación de clusters esféricos**, lo cual puede **funcionar bien cuando los datos tienen estructura circular**.
- Sin embargo, puedes presentarse **problemas** cuando por ejemplo tenemos **clusters elongados**. Este mismo problema lo presenta el propio K-MEANS.



- En dichos casos, se aconseja realizar **transformaciones** que permite hacerlos **más esféricos**.

Métodos basados en densidades. GMM

- Se puede **combinar un análisis de componentes principales y uno de correlación canónica** para proyectar los puntos en un nuevo espacio respecto al cual sí presentan dicho patrón).



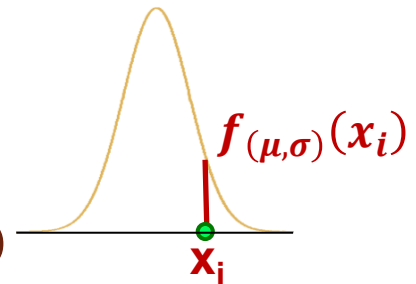
- El **inconveniente** es que, mientras que en dos/tres dimensiones se puede intuir visualmente si las observaciones presentan comportamientos elongados o esféricos, cuando el **número de dimensiones se dispara**, esta intuición no suele tenerse.
- En este contexto, surge el **Modelo de Mixturas Gaussianas (GMM)** que, aunque a veces es catalogado como un método de clustering, realmente **es un método de estimación de densidades, que tiene en cuenta la estructura de varianzas – covarianzas de los datos** (K-MEANS ni siquiera tiene en cuenta la varianza). **Es dicha matriz la que determina la forma de la distribución de los puntos.**

Métodos basados en densidades. GMM

- La idea consiste en **encontrar la combinación de gaussianas que maximiza la probabilidad de que una relación de puntos “i” esté representada por dichas gaussianas**: $\pi_{k,i} = p(x_i \in N(\mu_k, \sigma_k))$ $k=1,2, \dots, K$

- Dado un punto x_i y una distribución gaussiana $N(\mu, \sigma)$, se puede calcular cómo de verosímil es observar dicho punto si procede de dicha distribución. No es más que el valor de la densidad:

$$L(\mu, \sigma | x_i) = f_{(\mu, \sigma)}(x_i | \mu, \sigma) = f_{(\mu, \sigma)}(x_i) = p(x_i | x_i \in N(\mu_k, \sigma_k))$$



- Si en lugar de una, tuviéramos **k gaussianas**, se podría **calcular** igualmente dicha **verosimilitud como la suma de las probabilidades asociadas a cada una de las gaussianas**:

$$L((\vec{\mu}, \vec{\sigma}) | x_i) = f_{(\mu, \sigma)}(x_i) = \sum_{k=1}^K p(x_i \in N(\mu_k, \sigma_k)) * p(x_i | x_i \in N(\mu_k, \sigma_k))$$

- El resultado es **extendible a un conjunto** $X = (x_1, x_2, \dots, x_n)$ asumiendo **que la verosimilitud de observar cada uno de ellos es independiente de las otras y multiplicando dichas verosimilitudes**:

$$L((\vec{\mu}, \vec{\sigma}) | X) = \prod_{i=1}^n \sum_{k=1}^K p(x_i \in N(\mu_k, \sigma_k)) * p(x_i | x_i \in N(\mu_k, \sigma_k)) = \prod_{i=1}^n \sum_{k=1}^K \pi_{k,i} p(x_i | (\vec{\mu}_k, \Sigma_k))$$

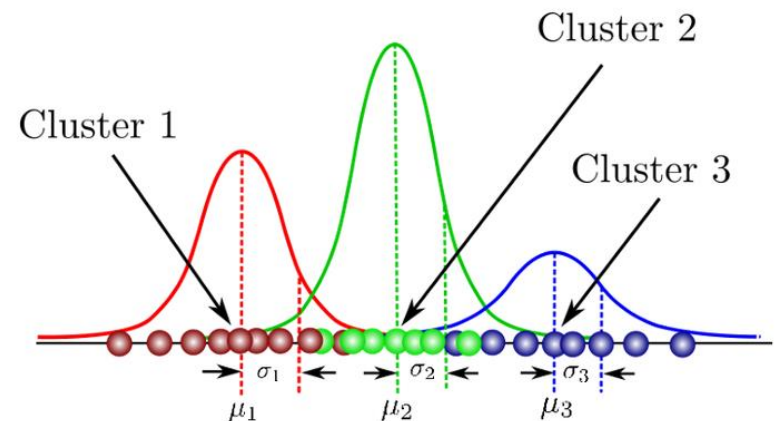
Métodos basados en densidades. GMM

- En este problema, las medias $\vec{\mu}$, las varianzas $\vec{\sigma}$ y los pesos $\vec{\pi}$ (con $\sum_{k=1}^K \pi_k = 1$) a asociar a cada gaussiana son los parámetros a **estimar para tratar de hallar**:

$$\max_{\vec{\mu}, \vec{\sigma}, \vec{\pi}} L((\vec{\mu}, \vec{\sigma})|X)$$

- Para ello, se aplica un **algoritmo de Maximización de Esperanza (EM)** que básicamente consiste en:

- Inicializar k distribuciones gaussianas.
- Calcular la probabilidad de pertenencia de cada punto a cada una de las distribuciones $p(x_i \in N(\mu_k, \sigma_k))$.
- Recalcular los parámetros de las distribuciones $N(\mu_k, \sigma_k)$ basándose en dichas probabilidades.
- Repetir el proceso hasta que se maximice $L((\vec{\mu}, \vec{\sigma})|X)$.



- A diferencia del K- MEANS que es un método de clasificación HARD en el que cada punto va asociado a un único cluster, una de las **ventajas** del método **GMM** es que **devuelve de manera directa la probabilidad de que un punto pertenezca a cada uno de los clusters**, algo que de alguna manera se perseguía con el FUZZY K-MEANS (algoritmo de clasificación SOFT).

Tipos de algoritmos. Ventajas y desventajas. Útil

Model	Pros	Cons	Use Cases
K means	<i>Quickest centroid based algorithm</i>	<i>Suffers when there is noise in the data</i>	<i>Even cluster size, flat geometry, not too many clusters and general-purpose</i>
	<i>Very lucid and can scale up for large amount of data sets</i>	<i>Outliers can never be identified</i>	
	<i>Reduces intra-cluster variance measure</i>	<i>Even though it reduces intra-cluster variance, it faces local minimum problem</i>	
		<i>Not ideal for data sets of non-convex shapes</i>	
		<i>Complicated to predict best K value</i>	
Agglomerative Clustering	<i>Embedded flexibility regarding level of granularity using dedrogram</i>	<i>Computationally expensive</i>	<i>Possibly connectivity constraints, non Euclidean distances and many clusters</i>
	<i>Can handle of any forms of similarity or distance</i>	<i>Can't handle outliers</i>	
		<i>Ward's algorithm usually generates equal size clusters</i>	
DBSCAN	<i>Resistant to outliers</i>	<i>Highly sensitive to the two parameters- Eps and Min points</i>	<i>Uneven cluster sizes and non-flat geometry</i>
	<i>Can handle clusters of different shapes and sizes</i>	<i>DBSCAN cannot cluster data sets well with large variances in densities</i>	
	<i>Not required to specify the number of clusters</i>		
GMM	<i>Robust to outliers</i>	<i>The algorithm is highly complex and can be slow</i>	<i>Good for density estimation and flat geometry</i>
	<i>Provides the BIC score for selecting paramteres</i>		
	<i>Converges fast given good initialisation</i>		

Tipos de algoritmos. Ventajas y desventajas. Útil

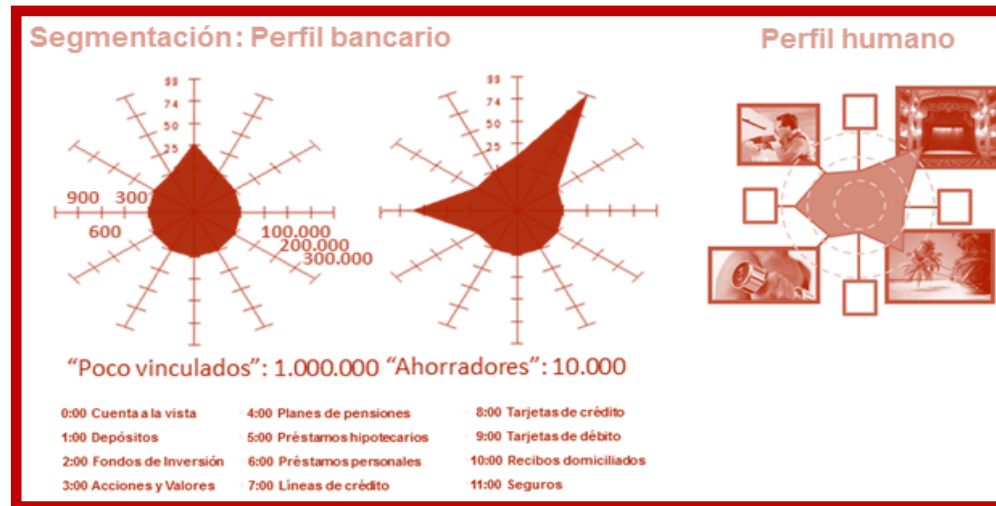
- Los algoritmos vistos quedan enmarcados dentro de una clasificación + amplia:

Name	Algorithm	Key –idea	Type of Data	Advantages	Disadvantages
Partitional	K-means	Mean Centroid	numerical	-Simple -Most popular	-Sensitive to outliers -Centroids not meaningful in most problems
	PAM	Mediod -centriod		robust to outliers	Cluster should be pre-determined
	CLARA			Applicable for large data set	Sensitive to outliers
	CLARANS			Handles outliers effectively	High cost
Density Based	DBSCAN	Fixed size	numerical	-Resistant to noise -Can handle clusters of various shapes and sizes.	-Cannot handle varying densities
	OPTICS	Variable size		-Good for data set with large amount of noise -Faster in computation	-Needs large no.of parameters
	DENCLUE			-Solid mathematical foundation	- Needs large no.of parameters
	RDBC			-More effective in discovering varied shape clusters -Handles noise effectively	-Cost Varying
Hierarchical agglomerative	CURE	Partition samples	Numerical	-Robust to outliers -Appropriate for handling large dataset	Ignores information about inter-connectivity of objects
	BIRCH	multidimensional	Numerical	-suitable for large databases -scales linearly	-Handles only numeric data -sensitive to data records
	ROCK	Notion of links	categorical	-Robust -Appropriate for large dataset	space complexity depends on initialization of local heaps
	S-link	Closest pair of points	-	it does not need to specify no.of clusters	-Termination condition needs to be satisfied. -Sensitive to outliers
	Ave-link	Centriod of clusters	-	It considers all members in cluster rather than single point	It produces clusters with same variance.
	Com-link	Farthest pair of points	-	Not strongly affected by outliers	It has problem with convex shape clusters.
Grid	STING	Multiple grids	Numerical	-Allows parallelization and multiresolution	-Does not define appropriate level of granularity
	WaveCluster		Numerical	- High-quality clusters - Successful outlier handling	-Cost Varying.
	CLIQUE	Density based grids		-Dimensionality reduction - Scalability -Insensitive to noise	-Prone to high dimensional clusters

6 | Validación de resultados

Validación de resultados

- La validación de resultados pasa por comprobar que:
 - Al **representar los centroides** de los clusters se reconozcan **grupos claramente diferentes y que tengan sentido: que permita asignarles un nombre (bautizar)**.



- El **número de grupos y el volumen de éstos sea razonable** (por ejemplo, si en función de ellos se tiene intención de articular acciones comerciales).
- **No se echa en falta ningún grupo** esperable antes de realizar el análisis.

7 | Caso de uso I

Segmentación de Estados en función de su nivel de criminalidad

Caso de uso. Segmentación de estados en función de su nivel de criminalidad

- El archivo `crime.csv` contiene datos de criminalidad por 10.000 habitantes en cada uno de los estados de EEUU de acuerdo a los siguientes criterios:
 - Asesinatos (variable `MURDER`)
 - Violaciones (variable `RAPE`)
 - Robos (variable `ROBBERY`)
 - Asaltos (variable `ASSAULT`)
 - Agresiones (variable `BURGLARY`)
 - Hurtos (variable `LARCENY`)
 - Robos de coches (variable `AUTO_THEFT`)
- Se plantea realizar una segmentación en función de dichas variables:
 - Estandarizar los datos.
 - Eliminar posibles *outliers* aplicando la transformación logarítmica.
 - Aplicar los diferentes métodos de segmentación vistos.

8 | Práctica Segmentación bancaria

Práctica: Segmentación bancaria

- Una entidad bancaria está interesada en **conocer el perfil de los clientes de su cartera respecto a la tenencia y saldo de sus productos** de activo y pasivo:

PRODUCTOS DE AHORRO

- Cuenta a la vista (*checkingAccount*).- saldo medio en el último año.
- Depósitos (*deposit*).- saldo medio en el último año
- Acciones (*shareOfStock*).- saldo medio en el último año
- Planes de pensiones (*pensionPlan*).- saldo medio en el último año

PRODUCTOS DE ACTIVO

- Hipotecas (*mortgage*).- saldo deudor medio en el último año
- Préstamos personales (*loan*).- saldo deudor medio en el último año
- Tarjetas (*cards*).- importe medio gastado con tarjetas de crédito en el último año
- Seguros (*insurance*).- número de seguros contratados (muy vinculado a la tenencia de hipoteca)

VINCULACIÓN

- Recibos domiciliados (*billPayment*).- nº medio de recibos domiciliados en el último año
- Domiciliación nómina (*salary*).- indicador de tenencia de domiciliación de nómina

- **Entregar un .HTML / .PDF en el que se presente:**

- El procesamiento de datos previos para llevar a cabo un análisis de segmentación.
- Los pasos llevados a cabo para realizar un ajuste bietápico, presentado las decisiones tomadas para determinar el número de clusters establecido.
- Una descripción de los grupos obtenidos.

- La **información** está **contenida** en la tabla **segmentacionBanca.csv**.



Afi Escuela
de Finanzas

danielvelezserrano@gmail.com