APRENDIZAJE NO SUPERVISADO

Aprendizaje no supervisado

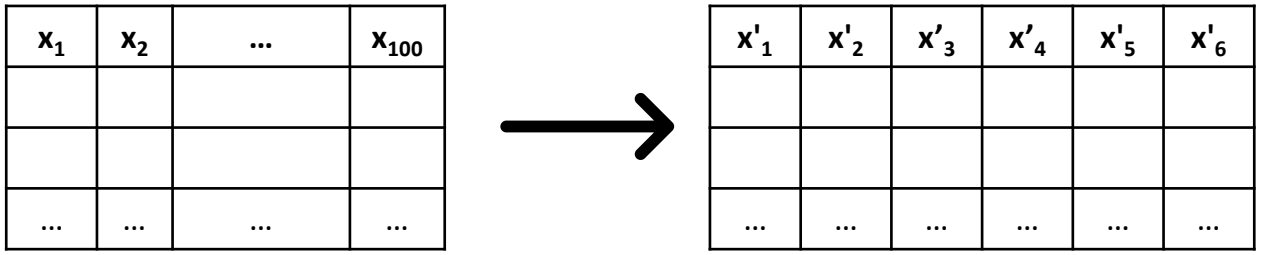
- Su objetivo es encontrar “estructura” en los datos proporcionados sin atender a ninguna categoría prefijada
  - Se les conoce también como técnicas exploratorias
- Típicamente se busca estructura en los individuos o en las variables
  - Estructura en los individuos: **técnicas de agrupamiento o *clustering***
  - Estructura en las variables: **técnicas de reducción de la dimensionalidad**
    - Nosotros nos centraremos en las técnicas de agrupamiento
- La estructura nos permite
  - “Reducir” nuestro conjunto de datos (reducir la dimensión de las variables o la de los individuos a grupos)
  - Ganar comprensión sobre los datos (a nivel de variables y/o de los individuos)
    - Ver qué variables están más (o menos) relacionadas entre sí y ver cómo se agrupan los individuos según su similitud

El problema de la dimensionalidad

- A menudo contar con muchas variables es un problema
  - "Maldición de la dimensionalidad": Podemos tener un número de dimensiones/variables muy elevado y no contar con suficientes ejemplos en muchas regiones de ese espacio multidimensional
    - Lo que "aprendamos" de esas regiones seguramente sea espúreo
  - Existen variables irrelevantes o redundantes
    - Algunas técnicas de aprendizaje no son capaces de elegir las variables relevantes para el problema y pueden confundirse si incluimos variables irrelevantes
  - Complica la visualización de los datos y su comprensión por el humano
    - ¿Cómo visualizar puntos con 100 o 1000 dimensiones?

Reducción de la dimensionalidad

- En las técnicas de reducción de la dimensionalidad
  - Se parte de un conjunto de variables m (es decir, m dimensiones)
  - Se busca reducirlo en un conjunto p mucho menor de factores que conserven el máximo posible de la información inicial (es decir, la variabilidad de las m variables)
  - En muchas de estas técnicas los factores se obtienen como una combinación de las variables originales
    - El Análisis de Componentes Principales (PCA) es la técnica clásica y usa combinación lineal
    - Existen variantes más sofisticadas para hacer combinaciones no lineales usando kernels



$$x'_1 = w_{1,1}x_1 + w_{1,2}x_2 + \cdots + w_{1,100}x_{100}$$
$$\cdots$$
$$x'_6 = w_{6,1}x_1 + w_{6,2}x_2 + \cdots + w_{6,100}x_{100}$$

Reducción de la dimensionalidad

- La interpretación de los p factores resultantes nos habla de dimensiones "ocultas" y de la estructura de las variables originales
  - De las variables originales habrá algunas que "contribuyen" más a un factor que a otros, lo que quiere decir que están más relacionadas con dicho factor
  - Las variables que contribuyen más en un factor están relacionadas entre sí
    - P.ej. En un conjunto de datos de pacientes variables como el peso y la altura estarán correlacionadas y podrían quedar agrupadas en un factor que nos hable del tamaño del individuo

APRENDIZAJE NO SUPERVISADO
TÉCNICAS DE CLUSTERING

Técnicas de agrupamiento o clustering

- El objetivo es agrupar los n individuos de nuestro conjunto de datos en una serie de grupos de forma que
  - Los individuos del mismo grupo sean lo más parecidos posible entre sí
  - Los individuos de grupos diferentes sean lo más diferentes entre sí
- De esta forma los grupos nos revelarán cierta "estructura" de los individuos de nuestro conjunto de datos, por ejemplo:
  - Cuáles son los grupos más numerosos y menos numerosos
  - Cuáles son los grupos más homogéneos y más dispersos
  - Qué individuos están más "alejados" de su grupo (outliers) o forman un grupo propio
- El concepto de parecido o de similitud suele requerir el uso de una medida de disimilitud o de distancia
  - Matemáticamente no toda disimilitud es una distancia
- Existen muchas familias de algoritmos de agrupamiento, aunque se suelen dividir en dos grandes grupos de los que veremos un ejemplo:
  - **Algoritmos de clustering jerárquico**
  - **Algoritmos de clustering basados en particiones**

Algoritmos de clustering jerárquico aglomerativo

- En un algoritmo de clustering jerárquico aglomerativo la estrategia general es la siguiente:
  1. Cada individuo empieza siendo un *cluster*, es decir, hay n *clusters*, tantos como individuos
  2. Repetir hasta que todos los individuos formen un único *cluster*
     - Agrupar los *clusters* más próximos en un único *cluster*

- Necesitamos definir:
  - La distancia (o disimilitud) que se usa para medir la proximidad (o similitud)
  - ¿Cómo se calcula la distancia entre *clusters* con más de un individuo?

- Existe una gran cantidad de distancias que pueden usarse
  - Su elección no debe ser casual ya que usar una distancia u otra hace que "priorices" unos aspectos frente a otros

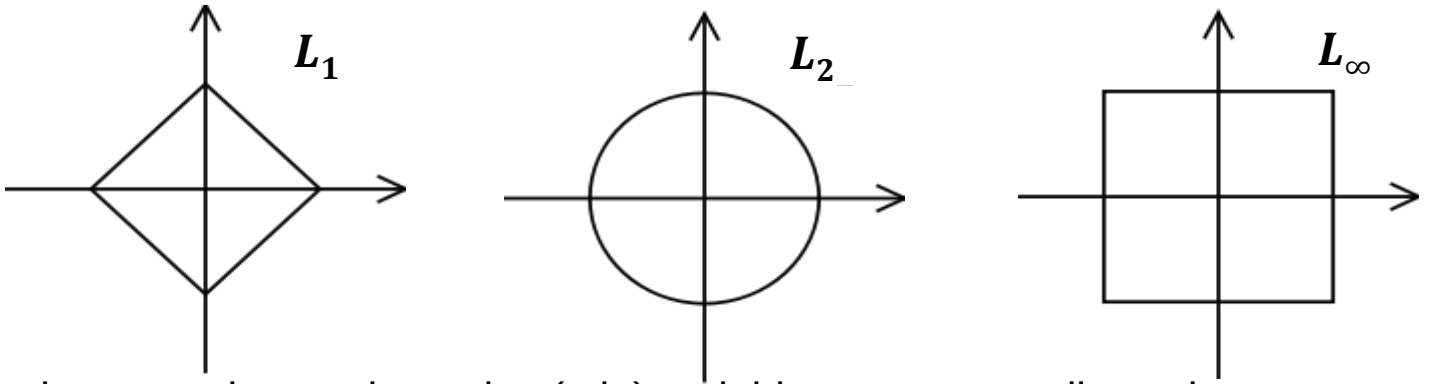
Distancias entre individuos

- La familia de métricas de Minkowski (L_p) suelen usarse habitualmente, especialmente sus variantes más famosas (Manhattan y Euclídea)
  - Sean dos individuos A y B descritos por m variables X_i con $i = 1, \dots, m$ definimos las distancias

Manhattan $L_1(A,B) = \sum_{i=1}^{m} |x_{Ai} - x_{Bi}|$

Euclídea $L_2(A,B) = (\sum_{i=1}^{m} (x_{Ai} - x_{Bi})^2)^{1/2}$

Chebychev $L_\infty(A,B) = max_i |x_{Ai} - x_{Bi}|$



En estas imágenes se muestran los puntos que se encuentran a la misma distancia en un espacio bidimensional con cada una de estas distancias

  - La importancia que tienen las (o la) variables con mayor distancia aumenta en $L_1 < L_2 < L_\infty$
    - En L_1 damos igual valor a todas las diferencias, en L_2 el cuadrado hace que pesen más las diferencias grandes y en L_∞ solamente se tiene en cuenta la variable donde la diferencia es mayor
- Es importante tener en cuenta que cuando tenemos varias variables, la magnitud en la que se mueven sus valores puede afectar a la distancia
  - No es lo mismo medir distancia entre dos variables una en centímetros y otra en kilos, que si lo hacemos en metros y gramos
  - Para evitar estos efectos se suelen **escalar** o **estandarizar** los datos

Distancias entre *clusters*

- Existen varias estrategias para medir distancias entre *clusters*
  - Se utiliza una de ellas durante todo el algoritmo
  - La estrategia usada afectará a la forma final de los *clusters*
- Las más típicas son:
  - **Centroide**: Se toma la distancia entre los puntos medios (el vector medio) de los dos *clusters*
  - **Enlace simple (*single linkage*)**: Se toma la distancia entre los puntos más próximos de los dos *clusters*
  - **Enlace completo (*complete linkage*)**: Se toma la distancia entre los puntos más alejados de los dos *clusters*

Ejemplo con dos clusters en un espacio bidimensional

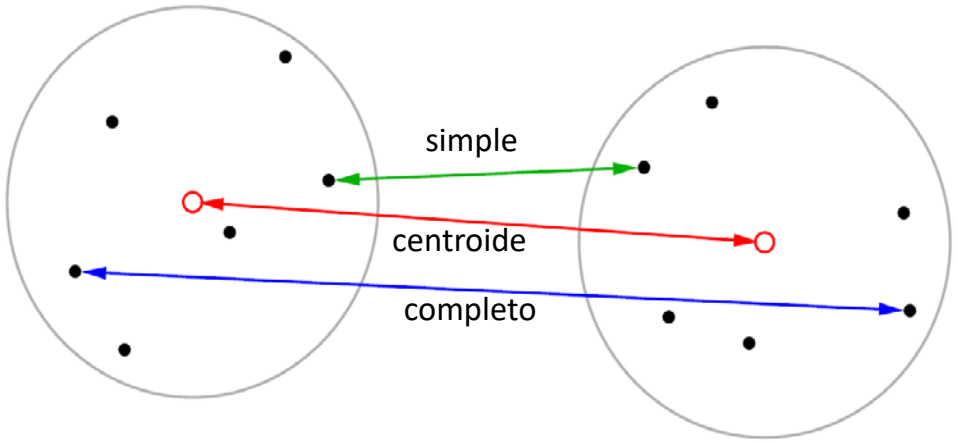


Imagen de C. Borgelt (2012). Intelligent Data Analysis

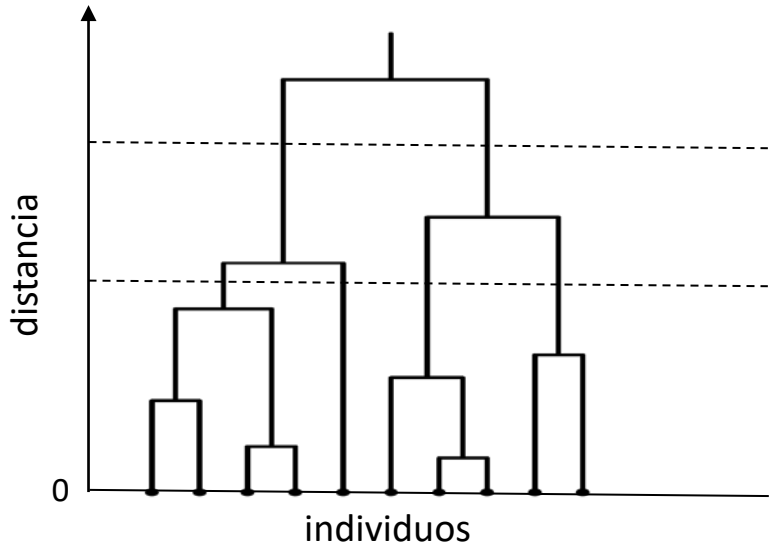
Algoritmo extendido

- FASE 1: Crear la matriz de distancias inicial D
  - Es una matriz simétrica (basta con usar una de las matrices triangulares)
- FASE 2: Agrupación de Individuos
  1. Partición inicial P_0: Cada objeto es un *cluster*
  2. Calcular la partición siguiente usando la matriz de distancias D
     - Elegir los dos *clusters* más cercanos
       - Serán la fila y la columna del mínimo de la matriz D
     - Agrupar los dos en un *cluster*
       - Eliminar de la matriz la fila y columna de los *clusters* agrupados
       - Generar la nueva matriz de distancias D
         - » Añadir una fila y una columna con el *cluster* nuevo
         - » Calcular la distancia del resto de *clusters* al cluster nuevo
  3. Repetir paso 2 hasta tener sólo un *cluster* con todos los individuos
     - Representar el dendograma (árbol de clasificación)

- La complejidad con los enlaces simple y completo son computacionalmente costosas, $O(n^3)$, aunque existen implementaciones eficientes más livianas en $O(n^2)$

Dendrograma

- El dendrograma (*dendro* es árbol en griego) es una representación bidimensional de la jerarquía inferida por el algoritmo de *clustering* jerárquico
  - En un eje ponemos los individuos y los *clusters* (abcisas en nuestro caso)
    - El orden de los mismos favorecerá la representación del dendrograma para que no haya cruces
  - El otro eje representa la distancia (ordenadas en nuestro caso)
  - El gráfico representa que la unión entre dos *clusters* se produce a una distancia determinada
- El resultado es una jerarquía en la que podemos ver la estructura de agrupación que ha ido siguiendo el algoritmo



Según el punto de corte que usemos obtendremos un número diferente de clusters

La elección del punto de corte la suele marcar el analista o usar alguna heurística

Ejemplo del Algoritmo

- Dadas estas seis observaciones unidimensionales $\{2, 12, 16, 25, 29, 45\}$
- Podemos calcular su matriz de distancias (independientemente de la distancia elegida)

$$D = \begin{pmatrix} 0 & 10 & 14 & 23 & 27 & 43 \\ 10 & 0 & 4 & 13 & 17 & 33 \\ 14 & 4 & 0 & 9 & 13 & 29 \\ 23 & 13 & 9 & 0 & 4 & 20 \\ 27 & 17 & 13 & 4 & 0 & 16 \\ 43 & 33 & 29 & 20 & 16 & 0 \end{pmatrix}$$

- Según la estrategia de enlace usada obtendremos diferentes dendrogramas

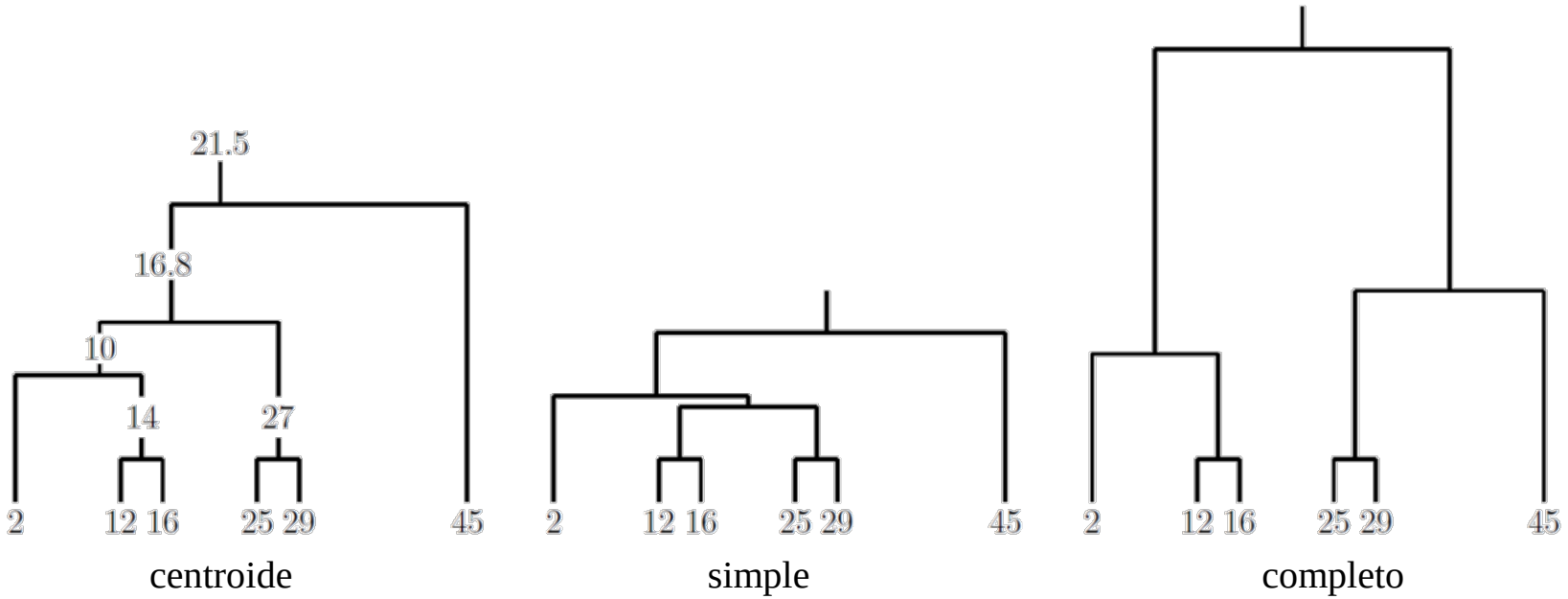
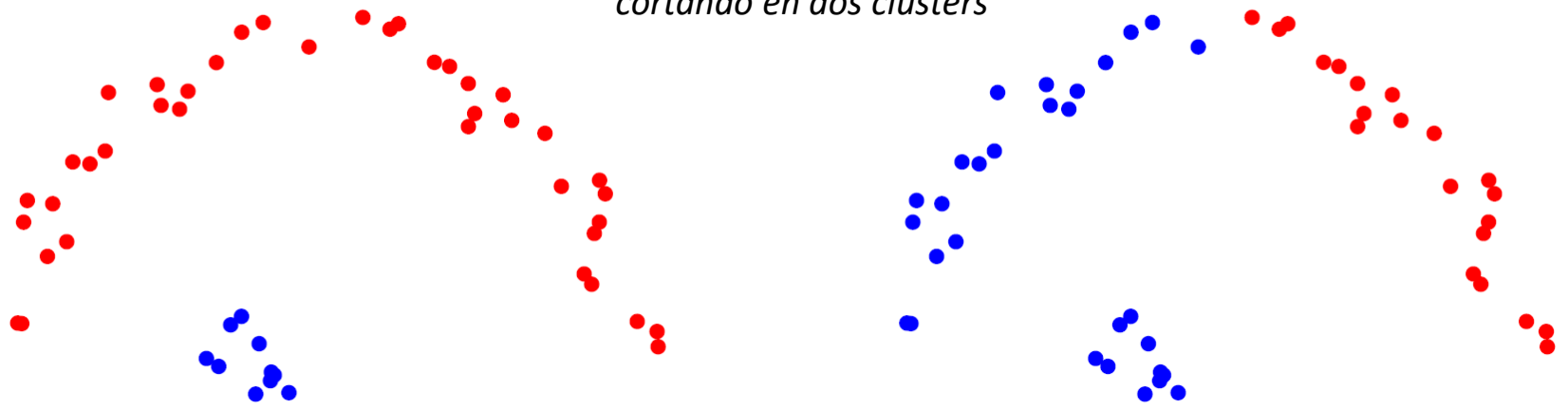


Imagen de C. Borgelt (2012). Intelligent Data Analysis

Impacto de las estrategias de clustering

- El enlace simple encadena observaciones próximas y tiende a generar cadenas
  - Funciona bien en clusters de formas diversas no necesariamente “homogéneas en todas sus direcciones” siempre y cuando los datos estén bien separados
- El enlace completo y el del centroide tienden a generar clusters más compactos
  - Funciona bien en clusters homogéneos y es más resistente al ruido en los datos

Ejemplo bidimensional cortando en dos clusters



simple

completo

Imagen de C. Borgelt (2012). Intelligent Data Analysis

Eligiendo el número adecuado de *clusters*

- Los problemas de clustering son aprendizaje no supervisado, por lo que **no hay una solución correcta** (una agrupación o un número de clusters correcto)
  - El objetivo es descubrir estructura en los datos y a priori todas son igual de correctas
  - El analista debe **interpretar** esa estructura inferida y ver qué le dice sobre el conjunto de datos que está analizando
    - Una solución es buena si los clusters son interpretables
- Hay varias aproximaciones para determinar el número de *clusters* haciendo un corte en el dendrograma
  - Usar una aproximación visual
  - Usar conocimiento experto y buscar *clusters* que sean interpretables
  - Especificar una distancia máxima a partir de la cual no formar *clusters*
  - Analizar la secuencia de distancias de los diferentes enlaces que se van haciendo al formarse la jerarquía y elegir un punto en el que la distancia se incremente mucho con respecto al enlace anterior
    - Existen heurísticas basadas en este concepto como el **diagrama del codo** que veremos más adelante
- En las dos primeras aproximaciones podríamos utilizar diferentes alturas de cortes para diferentes *subclusters*

Algoritmos *de clustering* basados en particiones

- El objetivo es dividir las n observaciones o individuos en un número de clusters k
  - Siendo el número $\boldsymbol{k}$ **un valor de entrada del algoritmo**
- Lo que hacen estos algoritmos es dividir nuestro espacio de representación m-dimensional en k regiones, siendo m las variables consideradas
  - Normalmente es una aproximación computacionalmente menos costosa que la jerárquica
- Las particiones se realizan optimizando un criterio ya sea globalmente o localmente (es decir, en un subconjunto de individuos)
- Existen muchos algoritmos de este tipo de los que el *k*-medias (*k-means* en inglés) es el más famoso

Algoritmo de k-medias

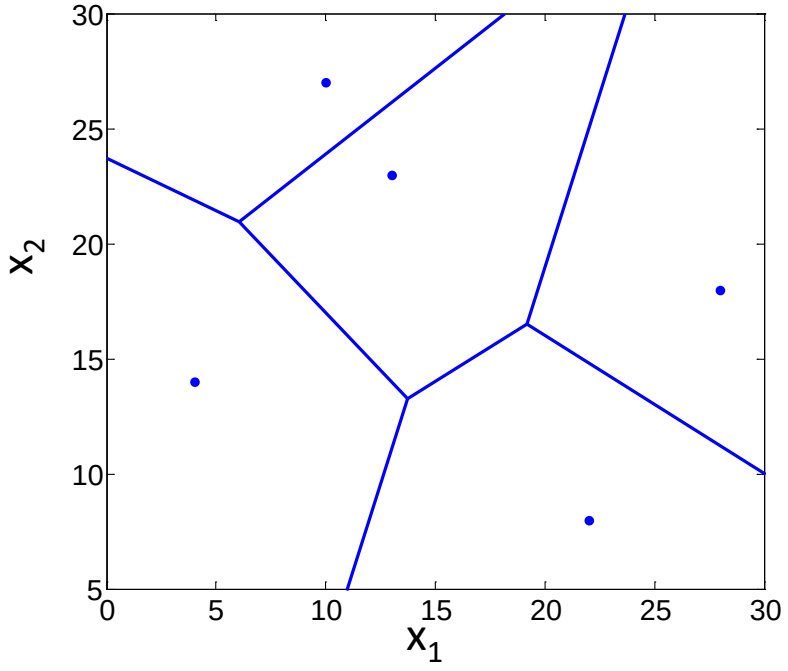
- Una vez fijado el valor de k, se realizan los siguientes pasos:
  1. Inicializa los k centros (o centroídes) de los *clusters* de forma aleatoria, típicamente
     - Generando k puntos aleatorios en el espacio m dimensional, o
     - Seleccionando aleatoriamente k individuos
  2. Repite el siguiente proceso hasta que los centros no cambien
     1. Fase de asignación: Asigna a cada punto la pertenencia al *cluster* que esté más cercano
        - Requiere el uso de una distancia (normalmente se usa la **distancia euclídea**)
     2. Actualiza el centro de los *clusters*, también llamado prototipo
        - Se calcula como el **individuo medio** de todos los individuos que pertenecen a ese *cluster*, es decir, se calcula la media de todas las variables consideradas de los individuos de cada *cluster*
        - El individuo medio es, además, el punto que minimiza la distancia euclídea entre sí mismo y los demás individuos del *cluster*

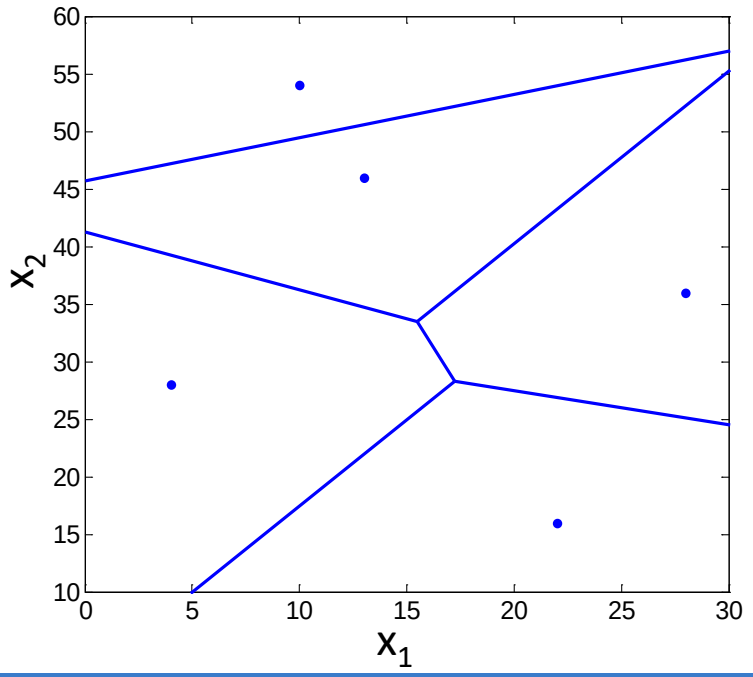
Algoritmo de k-medias

- El algoritmo de k-medias converge porque llega a un punto en el que los centros no cambian más
  - Y es bastante rápido (salvo en casos excepcionales)
- Sin embargo, la solución para un mismo k puede ser dependiente de la inicialización de los centros
  - Se puede ejecutar varias veces y ver si hay configuraciones más frecuentes o que nos convenzan más

Las regiones de Voronoi

- El algoritmo de k medias genera k particiones del espacio m-dimensional en el que se representan los individuos
- Estas regiones vienen determinadas por los k centros (o prototipos)
  - Cualquier punto dentro de una región determinada pertenece al *cluster* generado por dicho centro
- Por ejemplo, en los siguientes espacios bidimensionales con 5 prototipos se obtienen las siguientes regiones
  - En derecha, hemos hecho un cambio de escala en X_2 (multiplicando por 2)





¡La forma de las regiones cambia! Las unidades de medida de las variables (es decir, su escala) afecta a la forma de la región

El algoritmo k-medias en acción

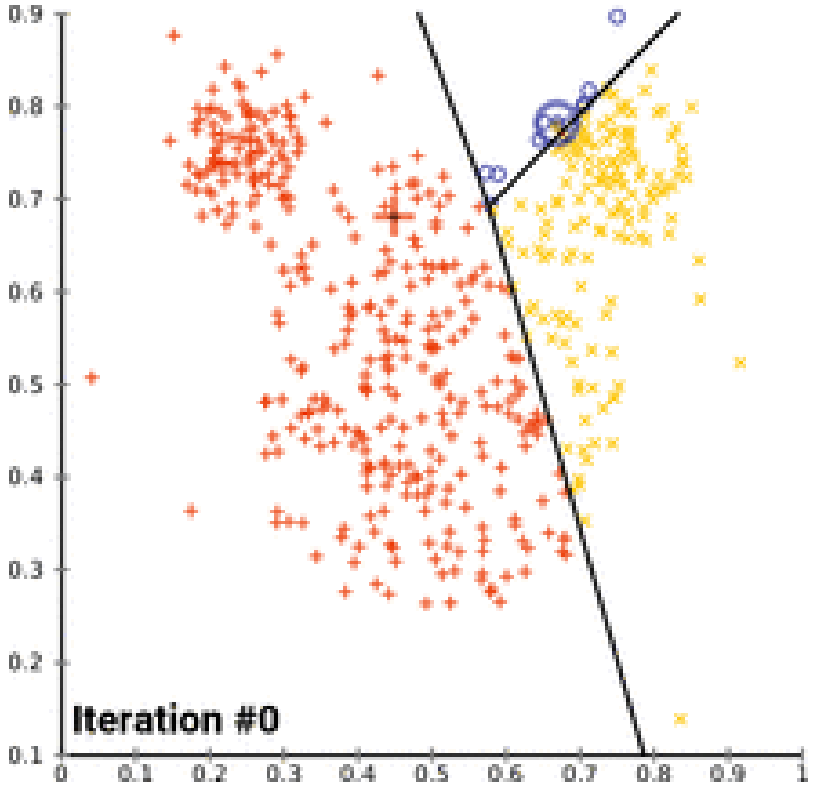


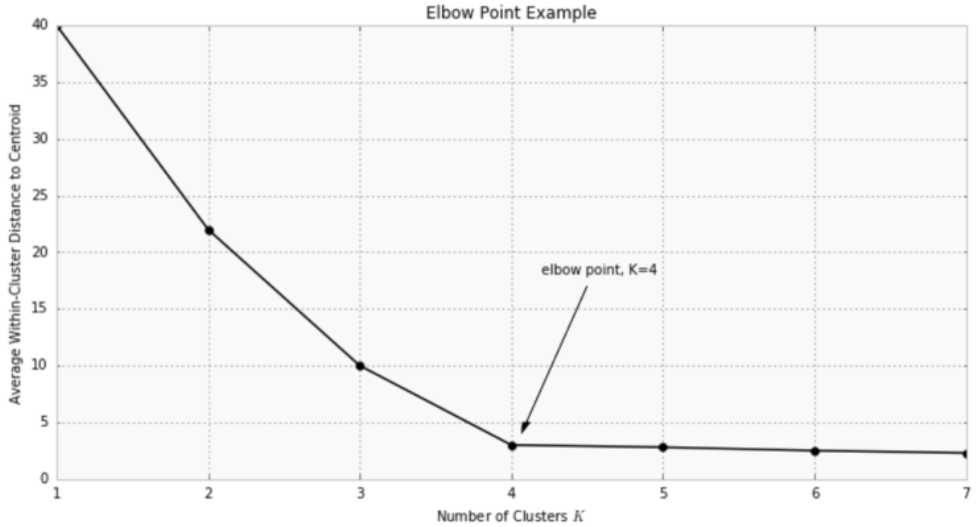
Imagen de Wikimedia Commons

Eligiendo el mejor resultado clustering

- ¿Cuántos clusters elegimos? Para el mismo número de clusters, ¿con qué resultado de los obtenidos con distintos algoritmos o con las distintas parametrizaciones de los algoritmos nos quedamos?
  - Como es aprendizaje no supervisado, no hay una solución (es decir, partición) correcta
  - Pero hay varias aproximaciones para ayudarnos a obtener una buena solución
- Una opción es usar conocimiento experto y buscar clusters que sean interpretables
  - Analizando los prototipos de los *clusters* y los individuos que pertenecen a ellos, así como lo dispersos o compactos que son (usando, p.ej. la desviación típica) y buscando que tengan sentido en el dominio del problema
- Existen índices que miden lo "compacto" de una solución, por ejemplo
  - **Dunn**: cociente entre la distancia mínima inter-*cluster* (entre los centros) y la distancia máxima intra-*cluster* (entre dos individuos de un *cluster*)
    - Cuanto mayor es el valor para una partición, mejor
  - **Davies-Bouldin** es un cociente entre la dispersión de los *clusters* y la separación de los mismos.
    - Cuanto más pequeño el valor obtenido para una partición, mejor
  - **Coeficiente de silueta**: similar a los anteriores, pero toma un valor entre 1 y -1, donde valores cercanos a 0 indican la existencia de clusters solapados, valores negativos indica una asignación incoherente y valores positivos indican particiones buenas.
- Hay otras aproximaciones más sofisticadas
  - Por ejemplo: usar validación cruzada (la veremos más adelante) para comprobar si la partición obtenida en los diferentes conjuntos de entrenamiento es homogénea con arreglo a un índice de calidad

Eligiendo el número adecuado de *clusters*: el diagrama del codo

- Dado un algoritmo de clustering, podemos determinar el número adecuado de clusters usando el diagrama del codo
  - Consiste en representar en el eje X el número de clusters, en el Y un índice de calidad y elegir el punto de X donde se aprecia un cambio de tendencia en Y
  - En general las métricas del eje Y serán mejores cuanto más clusters haya, pero puede haber un punto donde aumentar el número de clusters no produce una mejora significativa (el "codo")



- En el eje Y se suele poner:
  - Cuando usamos un algoritmo basado en particiones, la suma de las distancias al centroide
  - Cuando usamos un algoritmo jerárquico, el valor de la distancia del dendrograma (el eje Y del dendrograma) para cada número de clusters
- También se pueden usar los índices de Dunn, Daves-Boulding, el coeficiente de silueta, etc aunque no está garantizado que siempre se forme un codo, porque la serie no es "monótona".
  - El gráfico que se formará cambia de orientación en el eje Y según los valores óptimos del índice sean los más altos o los más bajos