6.1 Introducción

En aprendizaje no supervisado trabajamos con datos sin etiquetas. El sistema "descubre" estructura (patrones, grupos) basándose en similitud entre ejemplos.

El significado de los grupos lo pone el diseñador: la máquina agrupa, tu interpretas (por ejemplo: "este grupos son tallas M", "este segmento son clientes frecuentes").

Conceptos clave: - Medida de similitud/distancia adecuada al problema - Función objetivo que cuantifique qué tan buena es la agrupación - Criterios para elegir el número de grupos

Ejemplos de usos: - Segmentación de clietnes - Detección de anomalías /outliers - Compresión de colores en imágenes - Agrupación de documentos por temática

6.2 Método K-medias

Queremos k clústeres donde cada punto pertenezca al centroide más cercano, minimizando la suma de distancias al centroide del clúster.

Minimizamos $J = \sum(\text{sobre clústeres}) \sum(\text{puntos del clúster}) ||x - \mu||^2$

1. Inicializa k centroides (aleatorios en principio)
2. Asignación: asigna a cada punto el centroide más cercano
3. Actualización: recalcula cada centroide como la media de los puntos asignados
4. Repite 2-3 hasta la convergencia

Ejemplo: Datos 2D (altura/peso simplificado):
$A(1,1), B(1.5,2), C(3,4), D(5,7), E(3.5,5), F(4.5,5), G(3.5,4.5)$.
$k = 2$, inicializa centroides en $A(1,1) y D(5,7)$.

- Asignación (distancia euclídea):
  - Cerca de A : A, B
  - Cerca de D : C, D, E, F, G
- Actualización de centroides:
  - $\mu 1 = media(A, B) = ((1 + 1.5)/2, (1 + 2)/2) = (1.25, 1.5)$
  - $\mu 2 = media(C, D, E, F, G) = (3.9, 5.1)$
- Siguiente iteración: vuelve a asignar con los nuevos centroides y repite hasta estabilizar.

6.3 Elegir el nú

A ojo podemos disintiguir agruapaciones en la imagen. Es un criterio empírico, no sabemos cómo realizamos estas agrupaciones en nuestra cabeza. ¿Cómo lo haría el método k-medias?

Suponemos que queremos identificar 2 grupos de datos. Partimos de los llamados centroides de cada uno de los grupos y los situamos aleatoriamente.

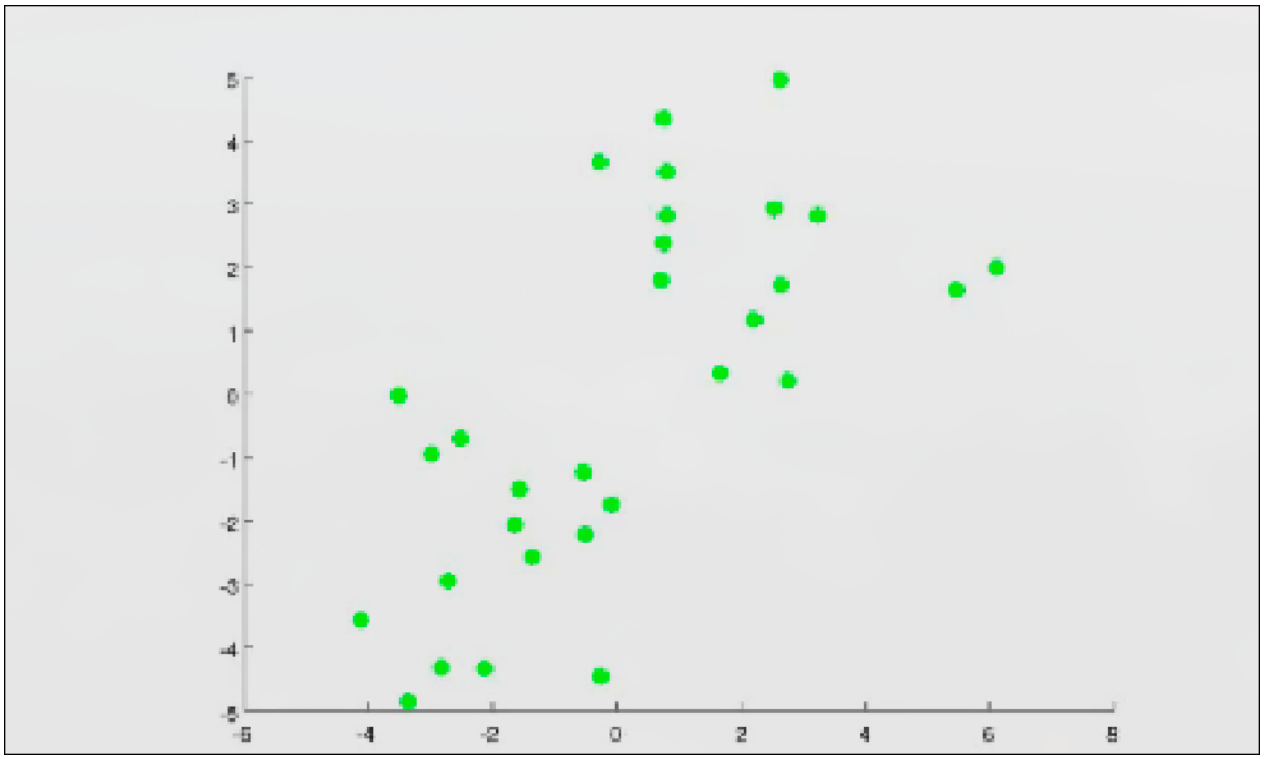

Figure 1: Pasted image 20251109161223.png

A partir de estos puntos de referencia, vemos que puntos están más cerca de uno y cuales más cerca del otro, identificando cada punto con un centroide. Depende del tipo de distancai que usemos y esta es una cuestión muy relevante que dependerá, en la mayoría de los casos, del problema que queremos resolver.

La iamgen anterior sería una primera división del conjunto de putnos de partida en los dos grupos que tratamos de identificar utilzando distnacia euclídea. EL poceso lógicamente no termina aquí, va iterando: - Ahora que ya tenemos disitnguidos dos conjutnos, vamos a calcular sus nuevos centroides: buscamos el punto medio de cada conjunto. - AL calcularlo, se van moviendo los centroides y puede ser que algún dato cacmbie de conjunto al estar más cerca del nuevo centroide del otro grtupo - Se repite el proceso hasta convergencia

La base de operaciones de este algoritmo es muy simple, si se analiza en detall y se utilizan otro ditpo de distancias, este puede llegar a ser más complejo. Algunas cuestiones: - Numero de agrupamientos en los que dividimos, nos lo pueden dar predefinido o no. Si no nos lo dan, veremos un criterio para elegirlo. - Donde inicializamos los centroides, ya que podrían variar el resultado. Se suele hacer aleatoriamente e influirá en la velocidad del algoritmo (número de iteraciones necesarias para converger). Por ejemplo, en la imagen - 1) A ojo, podemos distinguir 3 agrupamientos - 2) Si tenemos la suerte de inicializar bien los centroides desde un principio, el proceso será óptimo y rápido. - 3) Si inicializamos así los centroides, el rojo y el verde se repartirán esos 5 puntos y el resto se quedarán siempre en el azul. Mala distribución. - 4) Si partimos de esos centroides, no sería correcta la agrupación porque el cluster rojo quedaría sin puntos asociados.

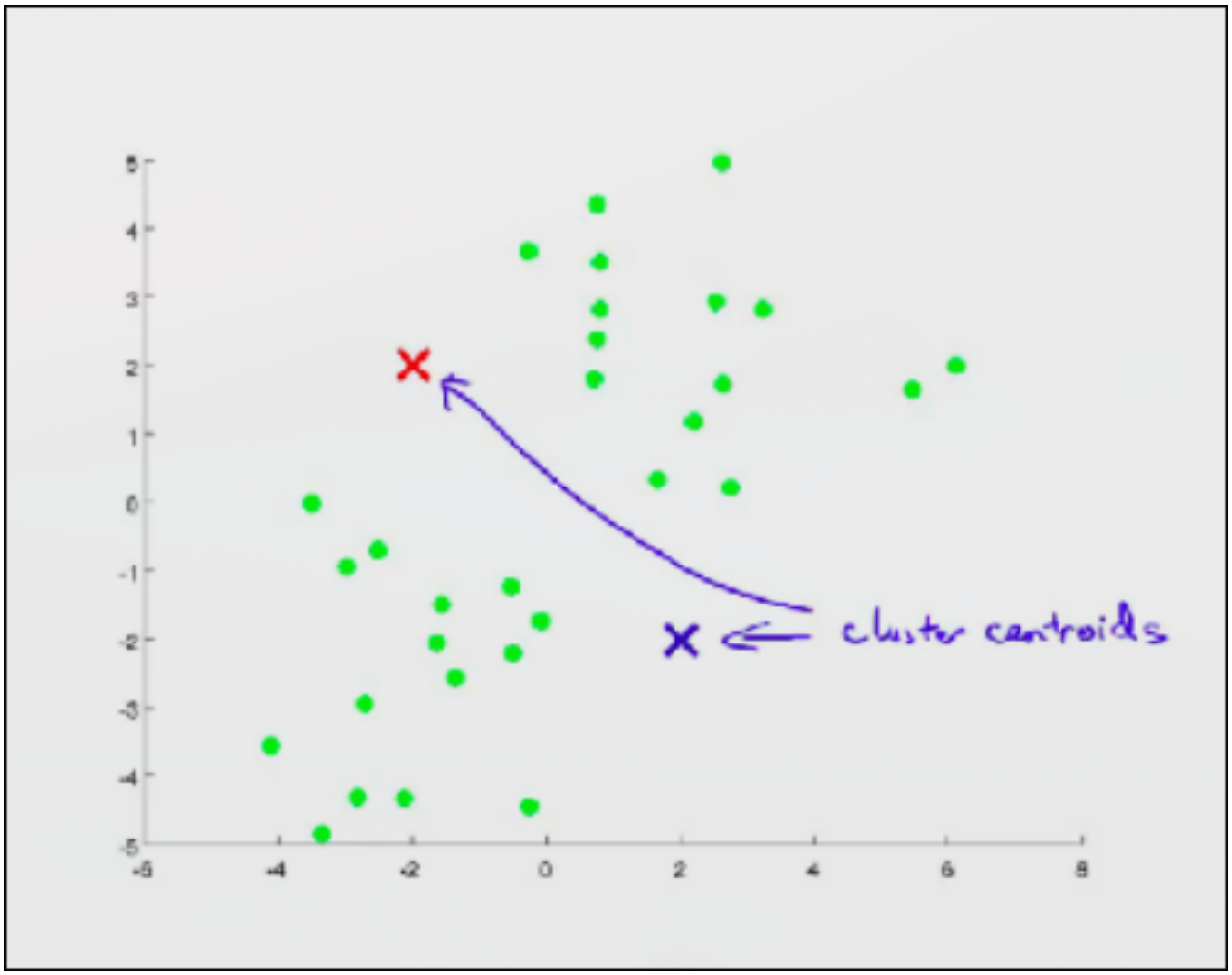

Figure 2: Pasted image 20251109161341.png

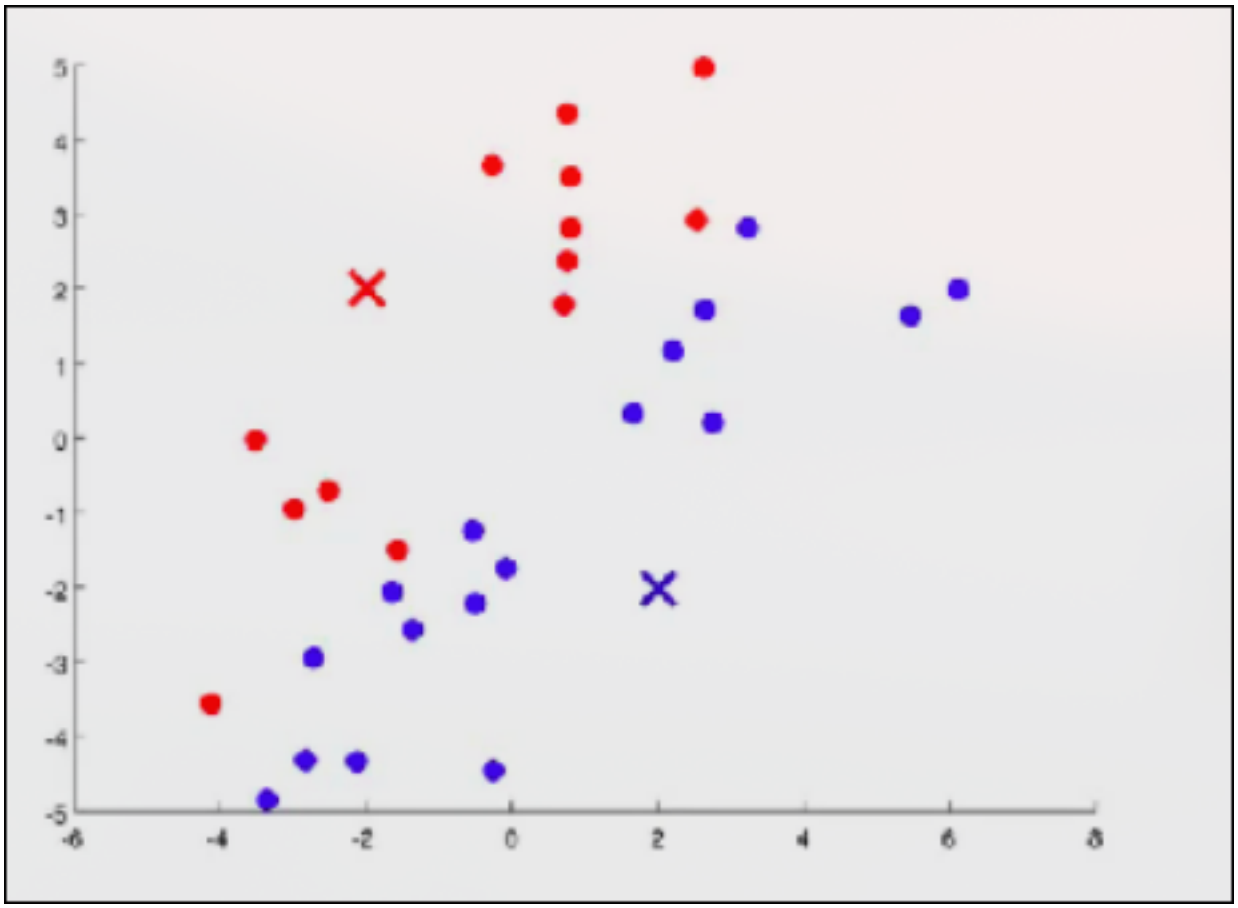

Figure 3: Pasted image 20251109161444.png

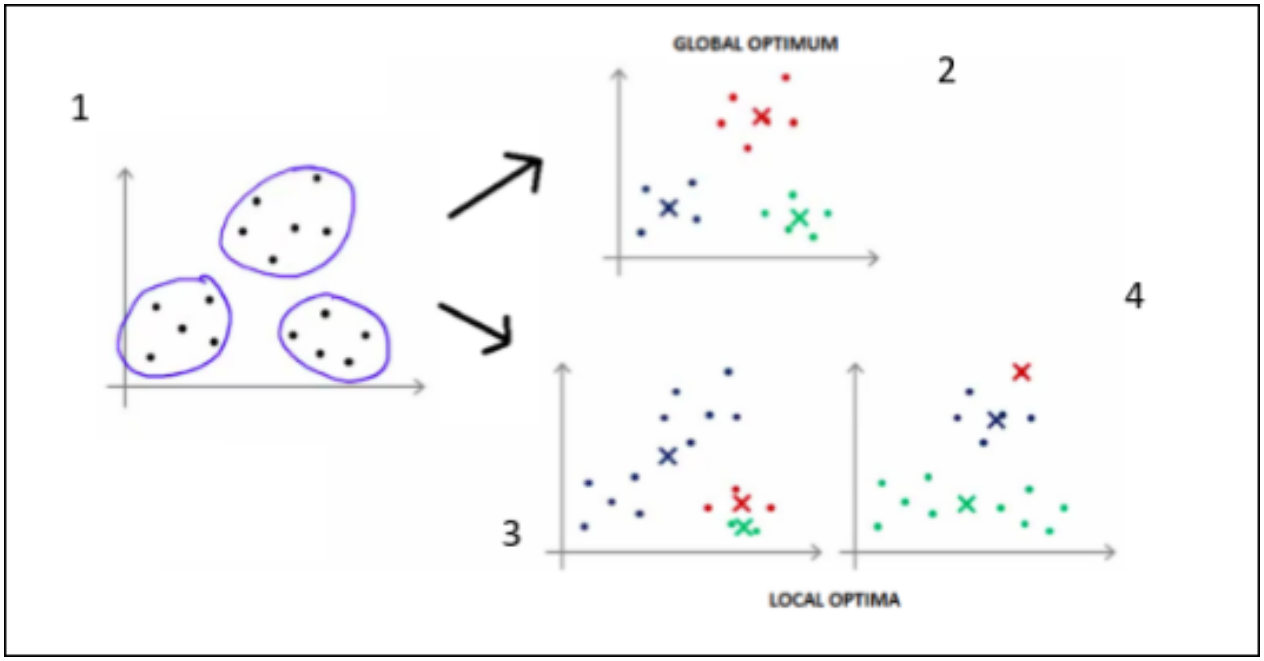

Figure 4: Pasted image 20251109161815.png

Para evitar problemas respecto a la inicialización de los clusters, podemos repetir el proceso muchas (cientos o miles, considerando el coste computacional que esto puede suponer) veces partiendo de distintos centroides y quedarnos con la mejor solución.

¿Cuál es esta mejor solución? A ojo, podemos ver en la imagen anterior cual es la mejor solución, la 2. Pero esto no nos vale computacionalmente. Deberemos tener un criterio de coste o rendimiento. Podemos suponer que este criterio normalmente es la solución que mejora una función de coste: por ejemplo, podemos decir que el coste o error será mayor cuanto más separados estén los puntos de un agrupamiento respecto a su centroide. Sumando las distancias euclídeas, en este caso, de los puntos a su centroide, podemos tener una medida del coste de la solución. Así también podríamos ver computacionalmente que la mejor solución es la 2, ya que es la que minimiza la función de error en este caso.

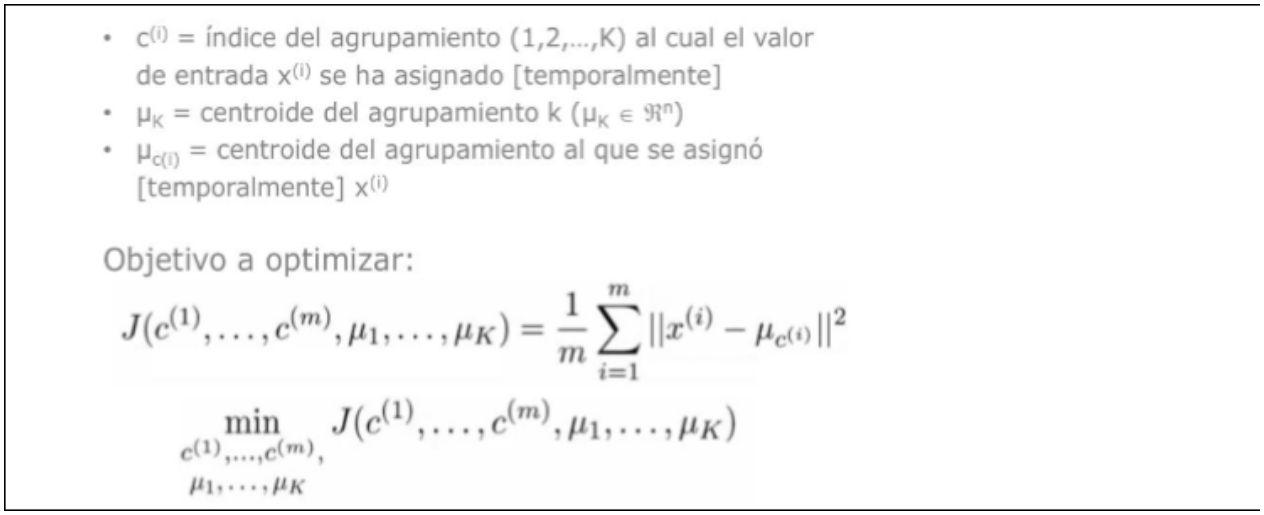

Figure 5: Pasted image 20251109161932.png

Tenemos un proceso de agrupamiento cuyo objetivo es maximizar la semejanza entre sí de los puntos pertenecientes a un agrupamiento, es decir, minimizar la distancia entre los puntos asociados en un agrupamiento y maximizarla con los de otros agrupamientos.

Pensando así podríamos decir que cada punto sea un agrupamiento distinto y por lo tanto sería optimo, ya que no habría distancia entre los puntos del agrupamiento, al ser solo un punto. Evidentemente, no diseñamos un sistema para que al final nos diga que tenemos tantos agrupamientos como puntos del conjunto de datos. Habrá que llegar a una solución de compromiso. ¿Cuántos clusters vamos a diferenciar? Podemos tener un número predefinido o obtener dicho número utilizando un criterio, que minimice o maximice, dependiendo de cada caso, el criterio establecido.

Ejemplo: no podemos diseñar camisetas a medida de cada consumidor. Debemos tener un cierto número de tallas y dicho número de tallas debe ser adecuado para ajustarse al público. No podemos tener demasiadas tallas, ya que esto supondría un gasto para el comerciante, pero tampoco demasiado pocas ya que debemos adaptarnos a los consumidores. Se podría coger una representación de la población a la que se dirige para la solución de este problema

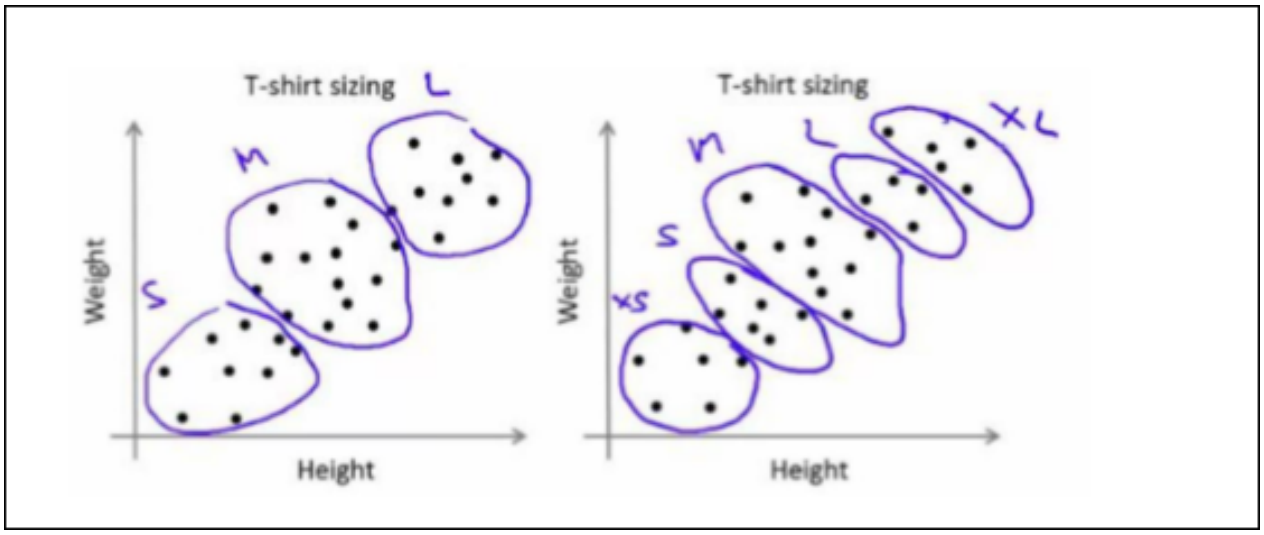

Figure 6: Pasted image 20251109162043.png

NOTA: En la imagen anterior podemos ver reflejada la idea de la que hablábamos al principio: el diseñador es el que le da sentido a los clusters, asignando una talla a cada uno y decidiendo que dicha talla debe cubrir a todas las personas incluidas en el cluster.

NOTA: En la imagen anterior podemos ver reflejada la idea de la que hablábamos al principio: el diseñador es el que le da sentido a los clusters, asignando una talla a cada uno y decidiendo que dicha talla debe cubrir a todas las personas incluidas en el cluster. Si no nos dan un número predefinido de clusters, deberemos buscar el mínimo número que represente bien el conjunto de datos. ¿Qué criterio usamos? Tenemos una función de coste, como la anterior, la que minimizaba la distancia euclídea de cada punto respecto a su centroide. ¿Dónde estaría una solución de compromiso entre el número de agrupamientos y la reducción de la función de coste? Se puede ir viendo, representando el coste respecto al número de clusters utilizado y nos va a quedar una gráfica así:

Para pocos clusters, el coste será alto. Según vamos añadiendo clusters, el coste cae muy rápidamente hasta llegar a un punto de inflexión donde ya, aunque vaya añadiendo más y más clusters, prácticamente ya no se reduce el coste o se reduce muy lentamente. Ese punto de inflexión o “codo” es el que se suele usar muchas veces como criterio práctico, se dice que esa solución de compromiso que buscábamos está ahí. A partir del codo, consideramos que el coste computacional adicional no vale la pena teniendo en cuenta la pequeña mejora del coste. En este caso, el número óptimo de clusters es 3.

Es bastante fácil que existan outliers o puntos atípicos en el conjunto de datos, debidos o no a errores de medida. Quedarán fuera de rango y distorsionarán la solución que obtendremos. Estos puntos se pueden identificar antes o durante el proceso de agrupamiento y descartarlos

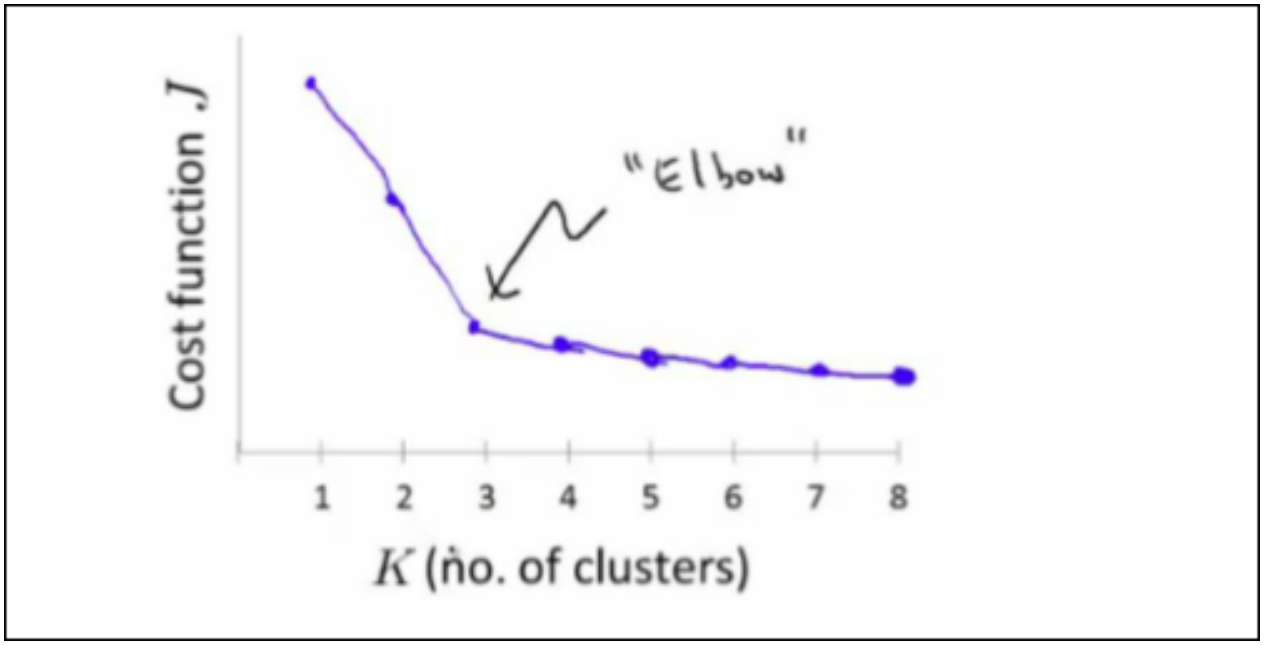

Figure 7: Pasted image 20251109162124.png