UTEC Posgrado

UTEC Posgrado

UTEC Posgrado

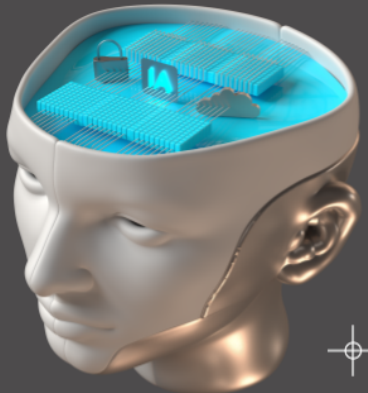
Una escuela para
Early Adopters

MACHINE LEARNING

APRENDIZAJE NO SUPERVISADO: K-MEANS

Motivación

Contexto y necesidad **del agrupamiento**



Aprendizaje supervisado vs. no supervisado

Supervisado

- Datos etiquetados: (x_i, y_i)

No supervisado

- Solo datos: $(x_1, x_2, \ldots, x_N)$

Aprendizaje supervisado vs. no supervisado

Supervisado

- Datos etiquetados: (x_i, y_i)
- Predecir respuesta Y dado X

No supervisado

- Solo datos: $(x_1, x_2, \ldots, x_N)$
- Descubrir estructura en $\Pr(X)$

Aprendizaje supervisado vs. no supervisado

Supervisado

- Datos etiquetados: (x_i, y_i)
- Predecir respuesta Y dado X
- Ejemplos: regresión, clasificación

No supervisado

- Solo datos: $(x_1, x_2, \ldots, x_N)$
- Descubrir estructura en $\Pr(X)$
- Ejemplos: clustering, PCA

Aprendizaje supervisado vs. no supervisado

Supervisado

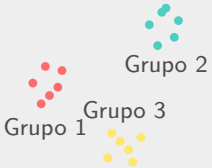
- Datos etiquetados: (x_i, y_i)
- Predecir respuesta Y dado X
- Ejemplos: regresión, clasificación
- Métrica clara de éxito

No supervisado

- Solo datos: $(x_1, x_2, \ldots, x_N)$
- Descubrir estructura en $\Pr(X)$
- Ejemplos: clustering, PCA
- Sin métrica directa de éxito

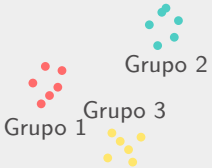
¿Por qué agrupar datos?

- **Segmentación de clientes**: perfiles de compra



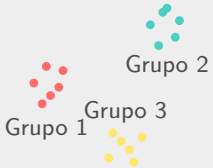
¿Por qué agrupar datos?

- **Segmentación de clientes**: perfiles de compra
- **Compresión de imágenes**: cuantización vectorial



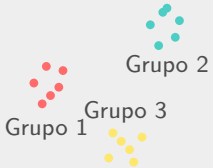
¿Por qué agrupar datos?

- **Segmentación de clientes**: perfiles de compra
- **Compresión de imágenes**: cuantización vectorial
- **Biología**: agrupamiento de genes por expresión



¿Por qué agrupar datos?

- **Segmentación de clientes**: perfiles de compra
- **Compresión de imágenes**: cuantización vectorial
- **Biología**: agrupamiento de genes por expresión
- **Documentos**: organización por temas



Objetivo del análisis de conglomerados

Definición: Agrupar N observaciones en K subconjuntos (conglomerados) tales que:

- Objetos **dentro** de un conglomerado sean **similares** entre sí

Objetivo del análisis de conglomerados

Definición: Agrupar N observaciones en K subconjuntos (conglomerados) tales que:

- Objetos **dentro** de un conglomerado sean **similares** entre sí
- Objetos en conglomerados **distintos** sean **diferentes**

Objetivo del análisis de conglomerados

Definición: Agrupar N observaciones en K subconjuntos (conglomerados) tales que:

- Objetos **dentro** de un conglomerado sean **similares** entre sí
- Objetos en conglomerados **distintos** sean **diferentes**

Formalmente: Minimizar la dispersión intra-conglomerado

$$W(C) = \frac{1}{2} \sum_{k=1}^{K} \sum_{C(i)=k} \sum_{C(i')=k} d(x_i, x_{i'})$$

Tipos de algoritmos de agrupamiento

Tipo	Descripción	Ejemplo
Combinatorio	Asignación directa sin modelo probabilístico	K-medias, K-medoides
Mezcla	Modelo de mezcla de distribuciones	EM Gaussiano
Jerárquico	Construye árbol de agrupamiento	Enlace simple, completo

Medidas de disimilitud

La elección de la medida de distancia es **crítica**:

- **Euclidiana cuadrática:** $d(x_i, x_{i'}) = \sum_{j=1}^{p}(x_{ij} - x_{i'j})^2 = \|x_i - x_{i'}\|^2$

Medidas de disimilitud

La elección de la medida de distancia es **crítica**:

- **Euclidiana cuadrática:** $d(x_i, x_{i'}) = \sum_{j=1}^{p}(x_{ij} - x_{i'j})^2 = \|x_i - x_{i'}\|^2$
- **Manhattan:** $d(x_i, x_{i'}) = \sum_{j=1}^{p}|x_{ij} - x_{i'j}|$

Medidas de disimilitud

La elección de la medida de distancia es **crítica**:

- **Euclidiana cuadrática:** $d(x_i, x_{i'}) = \sum_{j=1}^{p}(x_{ij} - x_{i'j})^2 = \|x_i - x_{i'}\|^2$
- **Manhattan:** $d(x_i, x_{i'}) = \sum_{j=1}^{p}|x_{ij} - x_{i'j}|$
- **Correlación:** $\rho(x_i, x_{i'}) = \frac{\sum_j (x_{ij}-\bar{x}_i)(x_{i'j}-\bar{x}_{i'})}{\sqrt{\sum_j (x_{ij}-\bar{x}_i)^2 \sum_j (x_{i'j}-\bar{x}_{i'})^2}}$

Medidas de disimilitud

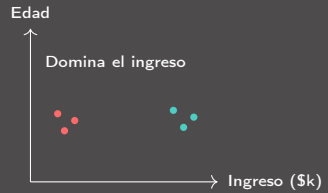
La elección de la medida de distancia es **crítica**:

- **Euclidiana cuadrática:** $d(x_i, x_{i'}) = \sum_{j=1}^{p}(x_{ij} - x_{i'j})^2 = \|x_i - x_{i'}\|^2$
- **Manhattan:** $d(x_i, x_{i'}) = \sum_{j=1}^{p}|x_{ij} - x_{i'j}|$
- **Correlación:** $\rho(x_i, x_{i'}) = \frac{\sum_j (x_{ij}-\bar{x}_i)(x_{i'j}-\bar{x}_{i'})}{\sqrt{\sum_j (x_{ij}-\bar{x}_i)^2 \sum_j (x_{i'j}-\bar{x}_{i'})^2}}$

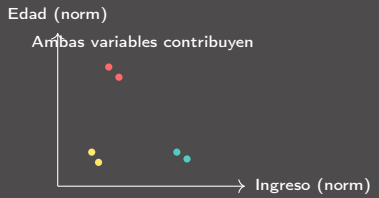
▷ K-medias usa distancia **euclidiana cuadrática**.

Efecto de la escala en el agrupamiento

Sin normalizar



Normalizado



Pregunta motivadora

¿Cómo podemos encontrar automáticamente

grupos "naturales" en los datos

sin etiquetas?

Pregunta motivadora

¿Cómo podemos encontrar automáticamente

grupos "naturales" en los datos

sin etiquetas?

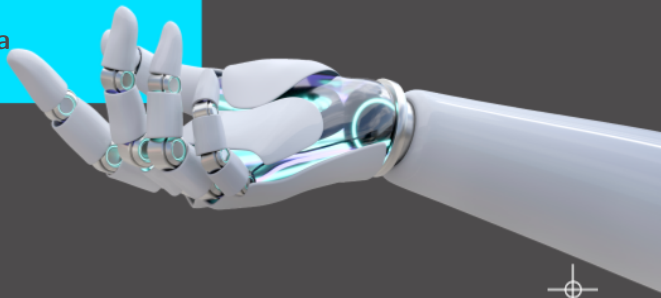
$\Longrightarrow$ **Algoritmo K-means**

UTEC Posgrado

Una escuela para
Early Adopters

Desarrollo

Algoritmo K-means **Teoría y práctica**



Intuición de K-means

Idea central: Particionar N puntos en K grupos minimizando la varianza intra-grupo.

1. Elegir K centros iniciales (centroides)

Intuición de K-means

Idea central: Particionar N puntos en K grupos minimizando la varianza intra-grupo.

1. Elegir K centros iniciales (centroides)
2. Asignar cada punto al centroide más cercano

Intuición de K-means

Idea central: Particionar N puntos en K grupos minimizando la varianza intra-grupo.

1. Elegir K centros iniciales (centroides)
2. Asignar cada punto al centroide más cercano
3. Recalcular los centroides como la media de cada grupo

Intuición de K-means

Idea central: Particionar N puntos en K grupos minimizando la varianza intra-grupo.

1. Elegir K centros iniciales (centroides)
2. Asignar cada punto al centroide más cercano
3. Recalcular los centroides como la media de cada grupo
4. Repetir pasos 2–3 hasta convergencia

Intuición de K-means

Idea central: Particionar N puntos en K grupos minimizando la varianza intra-grupo.

1. Elegir K centros iniciales (centroides)
2. Asignar cada punto al centroide más cercano
3. Recalcular los centroides como la media de cada grupo
4. Repetir pasos 2–3 hasta convergencia

▷ Convergencia garantizada (a un mínimo local).

Función objetivo de K-means

Buscamos el codificador C^* que minimice la dispersión intra-conglomerado:

$$W(C) = \sum_{k=1}^{K} N_k \sum_{C(i)=k} \|x_i - \bar{x}_k\|^2$$

donde:

- $\bar{x}_k$: media (centroide) del k-ésimo conglomerado
- $N_k = \sum_{i=1}^{N} I(C(i) = k)$: tamaño del conglomerado k
- $C(i)$: asignación del punto i al conglomerado k

Dispersión total = Intra + Inter

La dispersión total T es **constante**:

$$T = W(C) + B(C)$$

Intra-conglomerado $W(C)$

$$W = \frac{1}{2}\sum_{k=1}^{K}\sum_{\substack{C(i)=k \\ C(i')=k}} d_{ii'}$$

$\downarrow$ Minimizar

Inter-conglomerado $B(C)$

$$B = \frac{1}{2}\sum_{k=1}^{K}\sum_{\substack{C(i)=k \\ C(i')\neq k}} d_{ii'}$$

$\uparrow$ Maximizar

Complejidad del problema

El número de asignaciones posibles de N puntos a K conglomerados es:

$$S(N,K) = \frac{1}{K!}\sum_{k=1}^{K}(-1)^{K-k}\binom{K}{k}k^N$$

N	K	$S(N,K)$
10	4	34 105
15	4	$\approx 10^7$
19	4	$\approx 10^{10}$
100	5	$\approx 10^{68}$

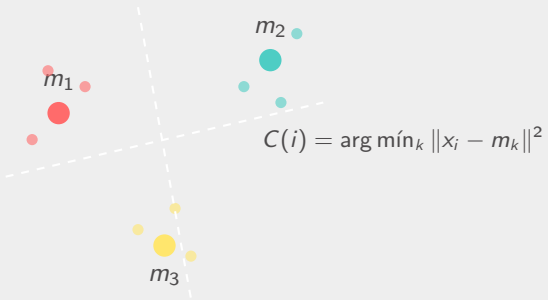
$\Longrightarrow$ Enumeración completa es **inviable**. Se usan heurísticas iterativas.

Algoritmo K-means

Entrada: Número de conglomerados K, datos $\{x_1, \ldots, x_N\}$
Salida : Asignación $C(i)$ y centroides $\{m_1, \ldots, m_K\}$

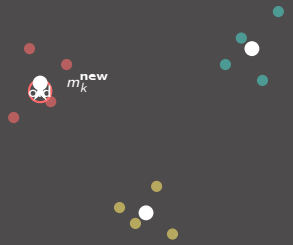
Inicializar centroides $m_1, \ldots, m_K$ (aleatoriamente)
repeat
  Paso de Asignación:
  for $i = 1$ **to** N **do**
    $C(i) \leftarrow \arg\min_{1 \leq k \leq K} \|x_i - m_k\|^2$
  end
  Paso de Actualización:
  for $k = 1$ **to** K **do**
    $m_k \leftarrow \frac{1}{N_k} \sum_{C(i)=k} x_i$
  end
until *las asignaciones C no cambien*
return $C, \{m_1, \ldots, m_K\}$

Paso 1: Asignación al centroide más cercano



Cada punto se asigna al centroide con menor distancia euclidiana.
Las regiones resultantes forman una **teselación de Voronoi**.

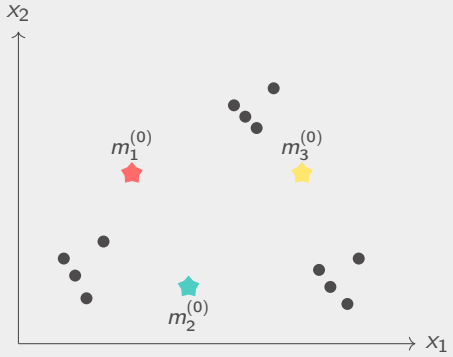
Paso 2: Actualización de centroides



$$m_k = \frac{1}{N_k} \sum_{C(i)=k} x_i$$

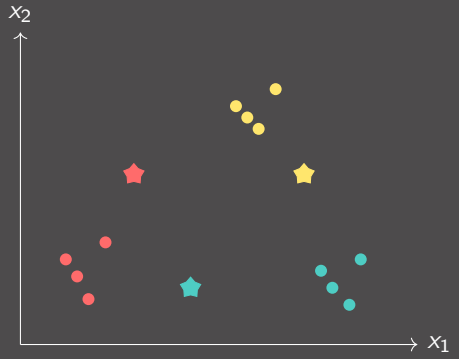
Cada centroide se recalcula como la **media** de los puntos asignados.

Ejemplo: Iteración 0 — Centroides iniciales



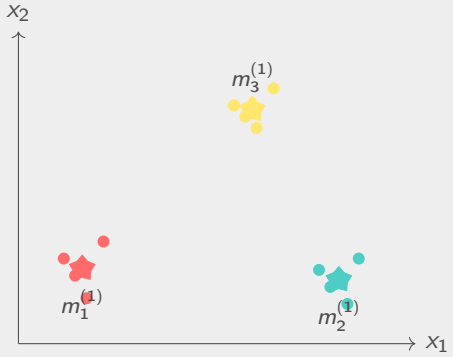
Se eligen $K = 3$ centroides de forma aleatoria.

Ejemplo: Iteración 1 — Asignación



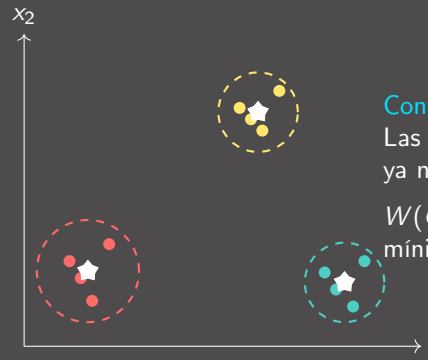
Cada punto se asigna al centroide más cercano por distancia euclidiana.

Ejemplo: Iteración 1 — Actualización



Los centroides se mueven a la **media** de sus puntos asignados.

Ejemplo: Convergencia



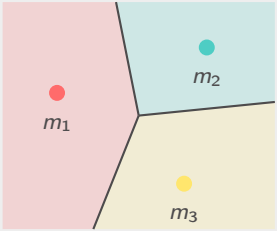
Convergió:
Las asignaciones ya no cambian.

$W(C)$ alcanzó un mínimo local.

Teselación de Voronoi

La asignación de K-means particiona el espacio en regiones:

- Cada región contiene un centroide
- Todos los puntos más cercanos a ese centroide
- Los bordes son **equidistantes** entre centroides



Garantía de convergencia

- El paso de **asignación** reduce (o mantiene) $W(C)$:
  Cada x_i se asigna al m_k más cercano $\Rightarrow$ W no puede aumentar.

Garantía de convergencia

- El paso de **asignación** reduce (o mantiene) $W(C)$:
  Cada x_i se asigna al m_k más cercano $\Rightarrow$ W no puede aumentar.
- El paso de **actualización** reduce (o mantiene) $W(C)$:
  $\bar{x}_S = \arg\min_m \sum_{i \in S} \|x_i - m\|^2$ (media minimiza SSE)

Garantía de convergencia

- El paso de **asignación** reduce (o mantiene) $W(C)$:
  Cada x_i se asigna al m_k más cercano $\Rightarrow$ W no puede aumentar.
- El paso de **actualización** reduce (o mantiene) $W(C)$:
  $\bar{x}_S = \arg\min_m \sum_{i \in S} \|x_i - m\|^2$ (media minimiza SSE)
- $W(C) \geq 0$ y es monótonamente decreciente $\Rightarrow$ **converge**.

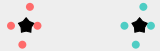
Garantía de convergencia

- El paso de **asignación** reduce (o mantiene) $W(C)$:
  Cada x_i se asigna al m_k más cercano $\Rightarrow$ W no puede aumentar.
- El paso de **actualización** reduce (o mantiene) $W(C)$:
  $\bar{x}_S = \arg\min_m \sum_{i \in S} \|x_i - m\|^2$ (media minimiza SSE)
- $W(C) \geq 0$ y es monótonamente decreciente $\Rightarrow$ **converge**.

Advertencia: La convergencia es a un **mínimo local**, no necesariamente global.

Problema de mínimos locales

Inicialización A



$W = 1{,}5$ (óptimo)

Inicialización B

$W = 8{,}2$ (subóptimo)

Solución: Ejecutar K-means múltiples veces con inicializaciones aleatorias y seleccionar la que tenga menor $W(C)$.

Ejemplo numérico: Datos en $\mathbb{R}^2$

Consideremos 6 puntos con $K = 2$:

Punto	x_1	x_2	Cluster
A	1.0	1.0	1
B	1.5	2.0	1
C	3.0	4.0	1
D	5.0	7.0	2
E	3.5	5.0	2
F	4.5	5.0	2

Centroides iniciales: $m_1 = (1{,}0, 1{,}0)$, $m_2 = (5{,}0, 7{,}0)$

Ejemplo: Cálculo de distancias

Distancias de cada punto a los centroides $m_1 = (1, 1)$ y $m_2 = (5, 7)$:

Punto	$\|x - m_1\|^2$	$\|x - m_2\|^2$	**Asignación**
$A(1, 1)$	0	52	Cluster 1
$B(1{,}5, 2)$	1,25	37,25	Cluster 1
$C(3, 4)$	13	13	Empate $\rightarrow$ 1
$D(5, 7)$	52	0	Cluster 2
$E(3{,}5, 5)$	22,25	6,25	Cluster 2
$F(4{,}5, 5)$	28,25	4,25	Cluster 2

Ejemplo: Actualización de centroides

Cluster 1: $\{A(1,1), B(1{,}5,2), C(3,4)\}$

$$m_1^{(1)} = \left(\frac{1+1{,}5+3}{3}, \frac{1+2+4}{3}\right) = (1{,}83, 2{,}33)$$

Cluster 2: $\{D(5,7), E(3{,}5,5), F(4{,}5,5)\}$

$$m_2^{(1)} = \left(\frac{5+3{,}5+4{,}5}{3}, \frac{7+5+5}{3}\right) = (4{,}33, 5{,}67)$$

Se repite el proceso de asignación con los nuevos centroides...

Propiedades de K-means

Propiedad	Detalle
Complejidad	$O(N \cdot K \cdot p \cdot I)$ por ejecución (I = iteraciones)
Convergencia	Garantizada a mínimo local
Tipo de datos	Variables cuantitativas
Distancia	Euclidiana cuadrática
Sensibilidad	A inicialización y valores atípicos
Parámetro	Número de clusters K (definido por usuario)

Elección del número de conglomerados K

- W_K decrece al aumentar K (siempre)

Elección del número de conglomerados K

- W_K decrece al aumentar K (siempre)
- Buscar un "quiebre" (codo) en la curva W_K vs. K

Elección del número de conglomerados K

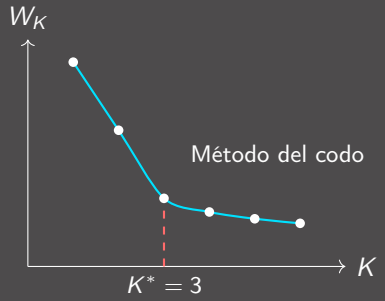
- W_K decrece al aumentar K (siempre)
- Buscar un "quiebre" (codo) en la curva W_K vs. K
- Para $K < K^*$: cada incremento reduce W significativamente

Elección del número de conglomerados K

- W_K decrece al aumentar K (siempre)
- Buscar un "quiebre" (codo) en la curva W_K vs. K
- Para $K < K^*$: cada incremento reduce W significativamente
- Para $K > K^*$: reducciones menores (dividir grupos naturales)

Elección del número de conglomerados K

- W_K decrece al aumentar K (siempre)
- Buscar un "quiebre" (codo) en la curva W_K vs. K
- Para $K < K^*$: cada incremento reduce W significativamente
- Para $K > K^*$: reducciones menores (dividir grupos naturales)



Estadístico Gap

Propuesto por Tibshirani et al. (2001), compara $\log W_K$ con datos uniformes:

$$\text{Gap}(K) = \mathbb{E}^*[\log W_K] - \log W_K$$

Estadístico Gap

Propuesto por Tibshirani et al. (2001), compara $\log W_K$ con datos uniformes:

$$\text{Gap}(K) = \mathbb{E}^*[\log W_K] - \log W_K$$

- $\mathbb{E}^*$: esperanza bajo distribución uniforme de referencia

Estadístico Gap

Propuesto por Tibshirani et al. (2001), compara $\log W_K$ con datos uniformes:

$$\text{Gap}(K) = \mathbb{E}^*[\log W_K] - \log W_K$$

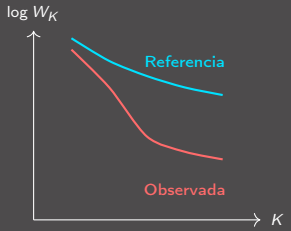
- $\mathbb{E}^*$: esperanza bajo distribución uniforme de referencia
- Elegir K^* como el menor K tal que:

$$K^* = \arg\min_K \{K \mid G(K) \geq G(K+1) - s'_{K+1}\}$$

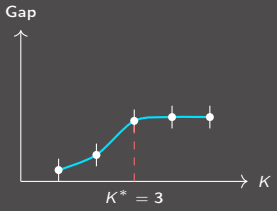
donde $s'_K = s_K \sqrt{1 + 1/B}$ y $B =$ número de simulaciones.

Estadístico Gap: Visualización

$\log W_K$ vs K



Curva Gap



K-means++: Mejor inicialización

Problema: Inicialización aleatoria puede producir malos resultados.

K-means++ (Arthur & Vassilvitskii, 2007):

1. Elegir el primer centroide m_1 uniformemente al azar de los datos

K-means++: Mejor inicialización

Problema: Inicialización aleatoria puede producir malos resultados.

K-means++ (Arthur & Vassilvitskii, 2007):

1. Elegir el primer centroide m_1 uniformemente al azar de los datos
2. Para cada punto x_i, calcular $D(x_i)$: distancia al centroide más cercano ya elegido

K-means++: Mejor inicialización

Problema: Inicialización aleatoria puede producir malos resultados.

K-means++ (Arthur & Vassilvitskii, 2007):

1. Elegir el primer centroide m_1 uniformemente al azar de los datos
2. Para cada punto x_i, calcular $D(x_i)$: distancia al centroide más cercano ya elegido
3. Elegir el siguiente centroide con probabilidad proporcional a $D(x_i)^2$

K-means++: Mejor inicialización

Problema: Inicialización aleatoria puede producir malos resultados.

K-means++ (Arthur & Vassilvitskii, 2007):

1. Elegir el primer centroide m_1 uniformemente al azar de los datos
2. Para cada punto x_i, calcular $D(x_i)$: distancia al centroide más cercano ya elegido
3. Elegir el siguiente centroide con probabilidad proporcional a $D(x_i)^2$
4. Repetir hasta tener K centroides

Algoritmo K-means++

Entrada: Datos $\{x_1, \ldots, x_N\}$, número de clusters K
Salida : Centroides iniciales $\{m_1, \ldots, m_K\}$

$m_1 \leftarrow$ punto aleatorio uniforme de $\{x_1, \ldots, x_N\}$
for $j = 2$ **to** K **do**
  for $i = 1$ **to** N **do**
    $D(x_i) \leftarrow \min_{l<j} \|x_i - m_l\|^2$
  end
  Elegir $m_j = x_i$ con probabilidad $\frac{D(x_i)^2}{\sum_{i'} D(x_{i'})^2}$
end
Ejecutar K-means estándar con $\{m_1, \ldots, m_K\}$

K-medoides: Alternativa robusta

K-medias

- Centroide = media del grupo
- Solo distancia euclidiana
- Sensible a outliers
- $O(NKpI)$

K-medoides

- Centroide = observación real
- Cualquier disimilitud
- Más robusto
- $O(N^2K)$ por iteración

El centro del cluster k en K-medoides:

$$i_k^* = \arg \min_{\{i:C(i)=k\}} \sum_{C(i')=k} D(x_i, x_{i'})$$

Algoritmo K-medoides

Entrada: Matriz de disimilitudes D, número de clusters K
Salida : Asignación $C(i)$ y medoides $\{i_1^*, \ldots, i_K^*\}$

1 Seleccionar K observaciones como medoides iniciales
2 **repeat**
3 | **Paso de Asignación:**
4 | **for** $i = 1$ **to** N **do**
5 | | $C(i) \leftarrow \arg\min_{1 \leq k \leq K} D(x_i, x_{i_k^*})$
6 | **end**
7 | **Paso de Actualización:**
8 | **for** $k = 1$ **to** K **do**
9 | | $i_k^* \leftarrow \arg\min_{\{i:C(i)=k\}} \sum_{C(i')=k} D(x_i, x_{i'})$
10 | **end**
11 **until** *las asignaciones C no cambien*

Comparación: K-means vs. K-medoides

Aspecto	K-means	K-medoides
Centro	Media (virtual)	Observación real
Distancia	Euclidiana	Cualquiera
Datos	Cuantitativos	Cuantitativos, ordinales, categóricos
Robustez	Baja (outliers)	Mayor
Complejidad	$O(NKpI)$	$O(N^2K)$
Interpretabilidad	Centroide abstracto	Punto real representativo

Ejemplo: Disimilitudes entre países

Datos de Kaufman y Rousseeuw (1990): 12 países con disimilitudes subjetivas.

	BEL	BRA	CHI	CUB	EGY	FRA
BRA	5.58					
CHI	7.00	6.50				
CUB	7.08	7.00	3.83			
EGY	4.83	5.08	8.17	5.83		
FRA	2.17	5.75	6.67	6.92	4.92	

- Solo disponemos de distancias $\Rightarrow$ K-means no es aplicable
- K-medoides con $K = 3$ agrupa correctamente por afinidad política

Resultado: 3-medoides en países

Cluster	Países
1	Bélgica, Francia, Israel, EE.UU., Egipto
2	Brasil, India, Zaire
3	Chile, Cuba, URSS, Yugoslavia

- Cluster 1: Democracias occidentales / aliados
- Cluster 2: Países en desarrollo
- Cluster 3: Países socialistas / afines

Caso real: Microarreglos de tumores

Datos: matriz 6830×64 (genes $\times$ muestras).
K-means con $K = 3$:

Cluster	Mama	SNC	Colon	Leucemia	Melanoma	Renal
1	3	5	0	0	1	9
2	2	0	0	6	7	0
3	2	0	7	0	0	0

Caso real: Microarreglos de tumores

Datos: matriz 6830×64 (genes $\times$ muestras).
K-means con $K = 3$:

Cluster	Mama	SNC	Colon	Leucemia	Melanoma	Renal
1	3	5	0	0	1	9
2	2	0	0	6	7	0
3	2	0	7	0	0	0

- Agrupa exitosamente muestras del mismo tipo de cáncer

Caso real: Microarreglos de tumores

Datos: matriz 6830×64 (genes $\times$ muestras).
K-means con $K = 3$:

Cluster	Mama	SNC	Colon	Leucemia	Melanoma	Renal
1	3	5	0	0	1	9
2	2	0	0	6	7	0
3	2	0	7	0	0	0

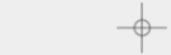
- Agrupa exitosamente muestras del mismo tipo de cáncer
- 2 muestras de "mama" en cluster 2 $\rightarrow$ resultó ser melanoma metastásico

Limitaciones observadas

1. **Sin orden dentro del cluster**:
   K-means no proporciona un ordenamiento lineal de los objetos dentro de cada grupo

Limitaciones observadas

1. **Sin orden dentro del cluster**:
   K-means no proporciona un ordenamiento lineal de los objetos dentro de cada grupo
2. **Inestabilidad al cambiar K**:
   Al cambiar K, las pertenencias pueden cambiar de manera arbitraria.
   Los $K + 1$ clusters no necesariamente están anidados dentro de los K anteriores.



Limitaciones observadas

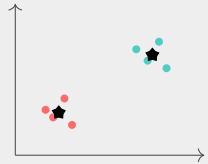
1. **Sin orden dentro del cluster**:
   K-means no proporciona un ordenamiento lineal de los objetos dentro de cada grupo
2. **Inestabilidad al cambiar K**:
   Al cambiar K, las pertenencias pueden cambiar de manera arbitraria.
   Los $K + 1$ clusters no necesariamente están anidados dentro de los K anteriores.
3. **Alternativa recomendada**:
   Agrupamiento jerárquico para datos de alta dimensión con estructura anidada.

Estrategias de inicialización

Método	Descripción	Ventaja/Desventaja
Aleatorio	Elegir K puntos al azar	Simple; puede ser malo
K-means++	Proporcional a D^2	Garantía teórica
Múltiples inicios	Repetir R veces, elegir mejor W	Más robusto; costoso
Progresivo	Agregar centros uno por uno minimizando W	Buena calidad; secuencial

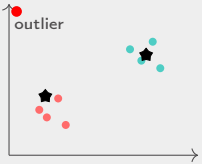
Sensibilidad a valores atípicos

Sin outlier



Centroides correctos.

Con outlier



Centroide desplazado.

La media es sensible a outliers $\Rightarrow$ considerar K-medoides.

Limitación: Formas de los clusters

K-means asume clusters **esféricos** (isotropos).

Para clusters elípticos o irregulares, considerar:

- Mezcla de Gaussianas (EM) con covarianza libre
- DBSCAN (basado en densidad)
- Clustering espectral

Mini-batch K-means

Para **grandes volúmenes de datos** ($N \gg 10^6$):

1. En cada iteración, tomar un subconjunto aleatorio (mini-batch) de tamaño $b \ll N$

Mini-batch K-means

Para **grandes volúmenes de datos** ($N \gg 10^6$):

1. En cada iteración, tomar un subconjunto aleatorio (mini-batch) de tamaño $b \ll N$
2. Asignar los puntos del mini-batch al centroide más cercano

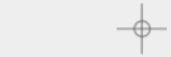
Mini-batch K-means

Para **grandes volúmenes de datos** ($N \gg 10^6$):

1. En cada iteración, tomar un subconjunto aleatorio (mini-batch) de tamaño $b \ll N$
2. Asignar los puntos del mini-batch al centroide más cercano
3. Actualizar centroides usando promedio ponderado incremental:

$$m_k \leftarrow m_k + \frac{1}{n_k}(x_i - m_k)$$

donde n_k es el conteo acumulado de asignaciones al cluster k



Mini-batch K-means

Para **grandes volúmenes de datos** ($N \gg 10^6$):

1. En cada iteración, tomar un subconjunto aleatorio (mini-batch) de tamaño $b \ll N$
2. Asignar los puntos del mini-batch al centroide más cercano
3. Actualizar centroides usando promedio ponderado incremental:

$$m_k \leftarrow m_k + \frac{1}{n_k}(x_i - m_k)$$

donde n_k es el conteo acumulado de asignaciones al cluster k

	K-means	Mini-batch
Datos por iteración	N	b
Calidad	Óptima local	Aproximada
Velocidad	Lenta	Rápida

Evaluación: Coeficiente de silueta

Para cada punto i asignado al cluster C_k:

- $a(i)$ = distancia promedio a los demás puntos de su cluster

Evaluación: Coeficiente de silueta

Para cada punto i asignado al cluster C_k:

- $a(i)$ = distancia promedio a los demás puntos de su cluster
- $b(i) = \min_{k' \neq k}$ distancia promedio a puntos de otro cluster

Evaluación: Coeficiente de silueta

Para cada punto i asignado al cluster C_k:

- $a(i)$ = distancia promedio a los demás puntos de su cluster
- $b(i)$ = $\min_{k' \neq k}$ distancia promedio a puntos de otro cluster
- Coeficiente de silueta:

$$s(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}} \in [-1, 1]$$

Evaluación: Coeficiente de silueta

Para cada punto i asignado al cluster C_k:

- $a(i)$ = distancia promedio a los demás puntos de su cluster
- $b(i) = \min_{k' \neq k}$ distancia promedio a puntos de otro cluster
- Coeficiente de silueta:

$$s(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}} \in [-1, 1]$$

$s(i)$	Interpretación
≈ 1	Bien asignado a su cluster
≈ 0	En la frontera entre dos clusters
≈ -1	Probablemente mal asignado

Resumen: Elección de K

Método	Idea	Limitación
Método del codo	Quiebre en W_K vs K	Subjetivo; codo no siempre claro
Estadístico Gap	Comparar con referencia uniforme	Costoso computacionalmente
Silueta	Cohesión vs separación	$O(N^2)$; no escala bien
Validación cruzada	No aplica directamente	W_K siempre decrece con K

K-means vs. agrupamiento jerárquico

K-means

- Requiere fijar K
- Partición plana
- Rápido: $O(NKpI)$
- Sensible a inicialización
- No anidado

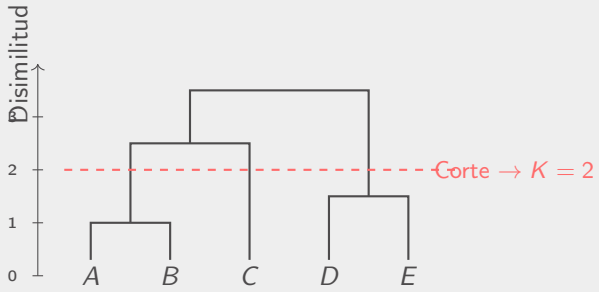
Jerárquico

- No requiere fijar K
- Dendrograma (jerárquico)
- Lento: $O(N^2 \log N)$
- Determinístico
- Estructura anidada

Enlace intergrupal más común:

- **Simple**: $d_{SL} = \min_{i \in G, i' \in H} d_{ii'}$
- **Completo**: $d_{CL} = \max_{i \in G, i' \in H} d_{ii'}$
- **Promedio**: $d_{GA} = \frac{1}{N_G N_H} \sum_{i \in G} \sum_{i' \in H} d_{ii'}$

Dendrograma: Visualización jerárquica



Cortando el dendrograma a una altura dada se obtiene una partición en K clusters.

Preprocesamiento para K-means

1. **Estandarizar variables:**
   $x'_{ij} = \frac{x_{ij} - \bar{x}_j}{s_j}$ para que cada variable tenga $\mu = 0$, $\sigma = 1$

Preprocesamiento para K-means

1. **Estandarizar variables:**
   $x'_{ij} = \frac{x_{ij} - \bar{x}_j}{s_j}$ para que cada variable tenga $\mu = 0$, $\sigma = 1$
2. **Eliminar valores atípicos extremos**
   O usar K-medoides si hay outliers

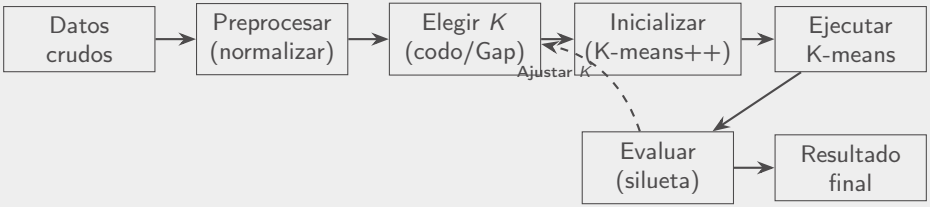
Preprocesamiento para K-means

1. **Estandarizar variables:**
   $x'_{ij} = \frac{x_{ij} - \bar{x}_j}{s_j}$ para que cada variable tenga $\mu = 0$, $\sigma = 1$
2. **Eliminar valores atípicos extremos**
   O usar K-medoides si hay outliers
3. **Manejar valores faltantes:**
   Imputar con media/mediana, o excluir pares con datos faltantes

Preprocesamiento para K-means

1. **Estandarizar variables:**
   $x'_{ij} = \frac{x_{ij} - \bar{x}_j}{s_j}$ para que cada variable tenga $\mu = 0$, $\sigma = 1$
2. **Eliminar valores atípicos extremos**
   O usar K-medoides si hay outliers
3. **Manejar valores faltantes:**
   Imputar con media/mediana, o excluir pares con datos faltantes
4. **Reducir dimensión (opcional):**
   PCA si $p \gg N$ para reducir ruido y complejidad

Pipeline completo de K-means



Conclusiones

Resumen y reflexiones **Próximos pasos**

Resumen: Algoritmo K-means

1. **Entrada:** N puntos en $\mathbb{R}^p$, número de clusters K

Resumen: Algoritmo K-means

1. **Entrada:** N puntos en $\mathbb{R}^p$, número de clusters K
2. **Inicializar:** K centroides (preferiblemente con K-means++)

Resumen: Algoritmo K-means

1. **Entrada:** N puntos en $\mathbb{R}^p$, número de clusters K
2. **Inicializar:** K centroides (preferiblemente con K-means++)
3. **Iterar:**
   - Asignar cada punto al centroide más cercano
   - Recalcular centroides como media del grupo

Resumen: Algoritmo K-means

1. **Entrada:** N puntos en $\mathbb{R}^p$, número de clusters K
2. **Inicializar:** K centroides (preferiblemente con K-means++)
3. **Iterar:**
   - Asignar cada punto al centroide más cercano
   - Recalcular centroides como media del grupo
4. **Convergencia:** Cuando las asignaciones no cambian

Resumen: Algoritmo K-means

1. **Entrada:** N puntos en $\mathbb{R}^p$, número de clusters K
2. **Inicializar:** K centroides (preferiblemente con K-means++)
3. **Iterar:**
   - Asignar cada punto al centroide más cercano
   - Recalcular centroides como media del grupo
4. **Convergencia:** Cuando las asignaciones no cambian
5. **Evaluar:** Silueta, método del codo

Ventajas y desventajas de K-means

Ventajas

- Simple e intuitivo
- Escalable: $O(NKpI)$
- Convergencia garantizada
- Fácil de implementar
- Versátil (VQ, preprocesamiento)

Desventajas

- Requiere especificar K
- Sensible a inicialización
- Solo mínimos locales
- Asume clusters esféricos
- Sensible a outliers
- Solo variables cuantitativas

¿Cuándo usar K-means?

Escenario	Recomendación
Clusters esféricos, datos numéricos	K-means
Outliers presentes	K-medoides / PAM
Datos categóricos	K-modos / K-prototipos
Clusters no convexos	DBSCAN / Spectral Clustering
Estructura jerárquica	Agrupamiento aglomerativo
Incertidumbre en asignaciones	Mezcla de Gaussianas (EM)

Conceptos clave aprendidos

1. **Aprendizaje no supervisado**: sin etiquetas, descubrir estructura

Conceptos clave aprendidos

1. **Aprendizaje no supervisado**: sin etiquetas, descubrir estructura
2. **K-means**: minimiza dispersión intra-conglomerado $W(C)$

Conceptos clave aprendidos

1. **Aprendizaje no supervisado**: sin etiquetas, descubrir estructura
2. **K-means**: minimiza dispersión intra-conglomerado $W(C)$
3. **Convergencia**: garantizada a mínimo local

Conceptos clave aprendidos

1. **Aprendizaje no supervisado**: sin etiquetas, descubrir estructura
2. **K-means**: minimiza dispersión intra-conglomerado $W(C)$
3. **Convergencia**: garantizada a mínimo local
4. **Inicialización**: K-means++ mejora la calidad

Conceptos clave aprendidos

1. **Aprendizaje no supervisado**: sin etiquetas, descubrir estructura
2. **K-means**: minimiza dispersión intra-conglomerado $W(C)$
3. **Convergencia**: garantizada a mínimo local
4. **Inicialización**: K-means++ mejora la calidad
5. **Elección de K**: método del codo, silueta

Conceptos clave aprendidos

1. **Aprendizaje no supervisado**: sin etiquetas, descubrir estructura
2. **K-means**: minimiza dispersión intra-conglomerado $W(C)$
3. **Convergencia**: garantizada a mínimo local
4. **Inicialización**: K-means++ mejora la calidad
5. **Elección de K**: método del codo, silueta
6. **Alternativas**: K-medoides (robusto), EM (suave), jerárquico (anidado)

Referencias

- Hastie, Tibshirani & Friedman (2009). *The Elements of Statistical Learning*, Cap. 14.
- Arthur & Vassilvitskii (2007). *k-means++: The Advantages of Careful Seeding*.

UTEC Posgrado

UTEC Posgrado