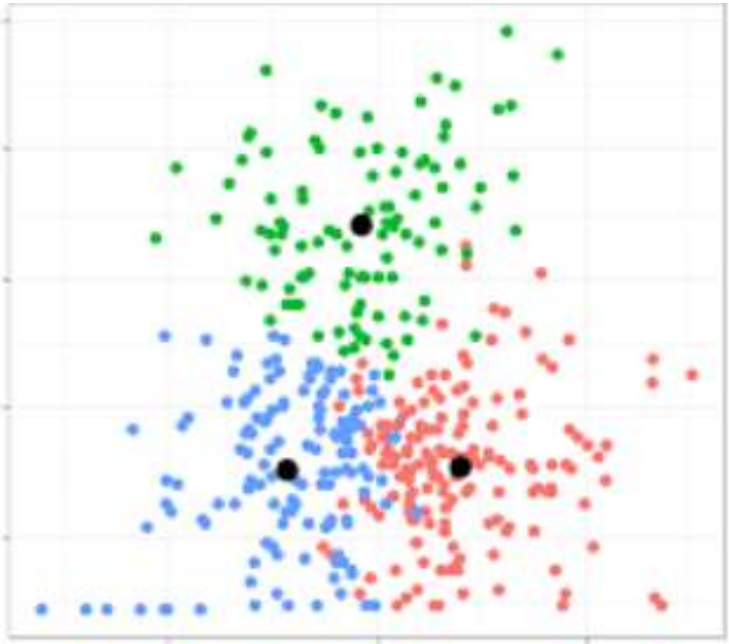
# Agrupamiento (*Clustering*): *K*-means

Aprendizaje Automático


Dr. Luis Eduardo Falcón Morales
ITESM
Campus Guadalajara

## Algoritmos de Agrupamiento (*clustering*)

- Los métodos de agrupamiento busca separar de manera automática un conjunto de datos en subconjuntos cuyos datos comparten alguna característica o características similares.
- Son algoritmos no supervisados.
- Son utilizados para sacar información de los datos aún cuando no se tenga conocimiento previo de ellos.
- A diferencia de los algoritmos de predicción o de clasificación, en estos no se construye ningún modelo. Es decir, son modelos no paramétricos.
- También son utilizados para reducir la dimensionalidad de los datos, al encontrar subgrupos de datos similares que puedan describirse con una cantidad menor de características (*features*).
- $K$-means es uno de los principales algoritmos de agrupamiento.



# Algoritmo *K-means*

Se tiene un conjunto $\Omega$ de $N$ puntos $m$ dimensional $\vec{x}_j = (x_{j1}, x_{j2}, \dots, x_{jm}) \in \mathbb{R}^m$, j $= 1,2, \dots, n$.

Se desea obtener una partición $S_1, S_2, \dots, S_K$ de $\Omega$, en $K$ subconjuntos disjuntos, para un entero $K$ dado.

Inicialización: Seleccionar de manera aleatoria $K$ centroides $\vec{\mu}_1, \vec{\mu}_2, \dots, \vec{\mu}_K$ .

Repetir hasta obtener la convergencia deseada o hasta alcanzar la cota del máximo número de iteraciones:

Definir $S_1 = S_2 = \cdots = S_K = \emptyset$

For $j = 1$ to $N$

Determinar $q \in \{1,2, \dots, K\}$ cuya distancia $\left\|\vec{x}_j - \vec{\mu}_q\right\|$ sea la mínima.

Asignar $\vec{x}_j$ al subconjunto $S_q$.

End For $\{j\}$

For $q = 1$ to $K$

Actualizar los centroides: $\vec{\mu}_q = \frac{1}{|S_q|} \sum_{\vec{x}_j \in S_q} \vec{x}_j$

End For $\{q\}$

donde $|S_q|$= cardinalidad del conjunto.

La función de costo $J$ definida para una partición $S_1, S_2, \ldots, S_K$ de $\Omega$ con centroides $\vec{\mu}_1, \vec{\mu}_2, \ldots, \vec{\mu}_K$, se define como:

$$J = \sum_{i=1}^{N} \sum_{j=1}^{K} \delta_{ij} \left\| \vec{x}_i - \vec{\mu}_j \right\|^2$$

Observamos que dicha función de costo viene a ser precisamente la traza de la matriz de dispersión de los datos.

donde además definimos el indicador binario $\delta_{ij}$ como sigue: $\delta_{ij} = \begin{cases} 1 & si \quad \vec{x}_i \in S_j \\ 0 & en\ otro\ caso \end{cases}$

Se desea minimizar dicha función de costo cuadrática con respecto a los centroides $\vec{\mu}_j$, entonces, calculando la derivada parcial $\frac{\partial J}{\partial \vec{\mu}_j}$ para cada $j \in \{1,2, \ldots, K\}$ e igualando a cero:

$$\frac{\partial J}{\partial \vec{\mu}_j} = -\sum_{i=1}^{N} 2\delta_{ij}\left(\vec{x}_i - \vec{\mu}_j\right) = 0$$

Despejando vemos que efectivamente los centroides de cada subconjunto de la partición son los puntos que minimizan mejor la función de costo:

$$\vec{\mu}_j = \frac{\sum_{i=1}^{N} \delta_{ij} \vec{x}_i}{\sum_{i=1}^{N} \delta_{ij}}$$

- Un punto que esté a una misma distancia mínima de varios centroides se puede resolver de manera aleatoria.

- Existen varios criterios para elegir los $K$ centroides iniciales, entre los más comunes están:
  - Generar aleatoriamente $K$ puntos en los rangos adecuados de cada coordenada.
  - Seleccionar aleatoriamente $K$ puntos del conjunto de puntos dado.
  - Seleccionar, del conjunto de puntos dado, los $K$ puntos más distantes entre sí.

- Re-escalar los datos para que las distancias de todas coordenadas sean comparables entre sí.

- Existe diversas variantes del algoritmo $K\text{-}means$ para la actualización de los centroides. Algunas de dichas variantes son:
  - Recalcular todos los centroides cada vez que se reasigna un dato a una clase diferente a la que se encontraba.
  - Para cada dato $\vec{x}_j$, actualizar el centroide $\mu_q^{(old)}$ más cercano a dicho punto mediante la expresión:

$$\mu_q^{(new)} = \mu_q^{(old)} + \gamma_j \left( \vec{x}_j - \mu_q^{(old)} \right)$$

    donde $\gamma_j$ es una razón de aprendizaje que generalmente decrece monotónicamente al incrementarse $j$.

- En ocasiones se tiene una convergencia a un mínimo local, por lo que debe repetirse el proceso para diferentes valores iniciales.

- El costo computacional del algoritmo $K\text{-}means$ es $\sigma(KN)$, para un conjunto de $N$ puntos.

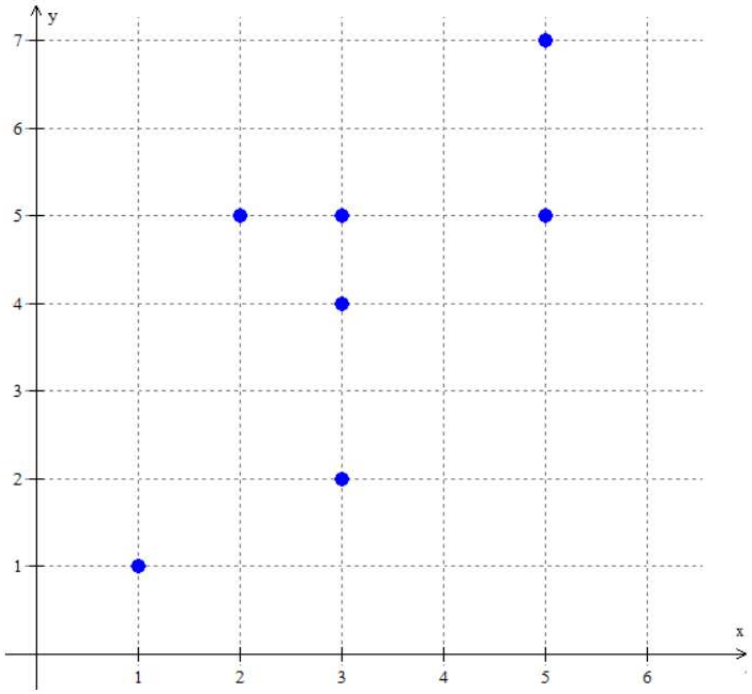
**Ejercicio:**

Aplicar el algoritmo $K\text{-}means$ al siguiente conjunto de datos, con $K = 2$:

$$\Omega = \{(1,1), (3,2), (2,5), (3,4), (3,5), (5,5), (5,7)\}$$

Considera la métrica euclidiana y como centroides iniciales: $\mu_1 = (2,4)$, $\mu_2 = (4,6)$

---

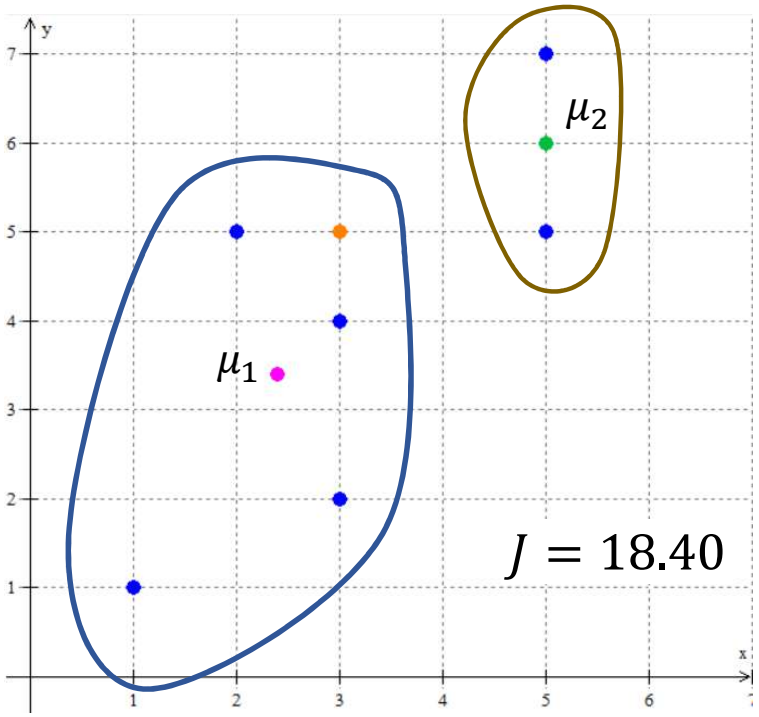



Existe un punto que está a la misma distancia de los dos centroides iniciales, resolverlo para ambos casos y mostrar cuál de ellos tiene el valor mínimo con la función de costo:

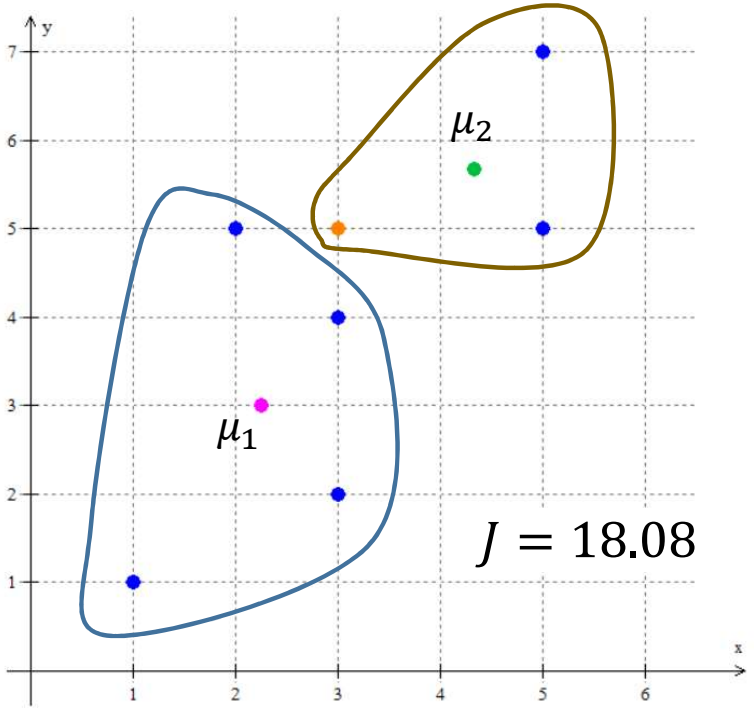
$$J = \sum_{i=1}^{N} \sum_{j=1}^{K} \delta_{ij} \left\| \vec{x}_i - \vec{\mu}_j \right\|^2$$

Soluciones Ejercicio: El punto $(3, 5)$ está a la misma distancia de ambos centroides iniciales, entonces se tendrían dos opciones:

Asignando el punto $(3, 5)$ a $\mu_1$ en la primera iteración:




Asignando el punto $(3, 5)$ a $\mu_2$ en la primera iteración:




Función de Costo: $J = \sum_{i=1}^{N} \sum_{j=1}^{K} \delta_{ij} \|\vec{x}_i - \vec{\mu}_j\|^2$

## Algoritmos Semi-Supervisados

Existe una gran variedad de casos de algoritmos semi-supervisados.

Uno de los casos de los algoritmos semi-supervisados se aplica a grandes rasgos como sigue:

- Se tiene como entrada un conjunto de datos no etiquetados.
- Etiquetar los datos aplicando un algoritmo de agrupamiento.
- Con dichas etiquetas, aplicar ahora un algoritmo supervisado para hacer predicciones con los datos.