09481: Inteligencia Artificial

Profesor del curso: Breyner Posso, Ing. M.Sc.
e-mail: breyner.posso1@u.icesi.edu.co

Programa de Ingeniería de Sistemas.
Departamento TIC.
Facultad de Ingeniería.
Universidad Icesi.
Cali, Colombia.

Agenda

1. Introducción.

2.Aprendizaje NO Supervisado.

3. Agrupamiento (clustering).

4. K-means.

5. Evaluación del agrupamiento.

1. Introducción

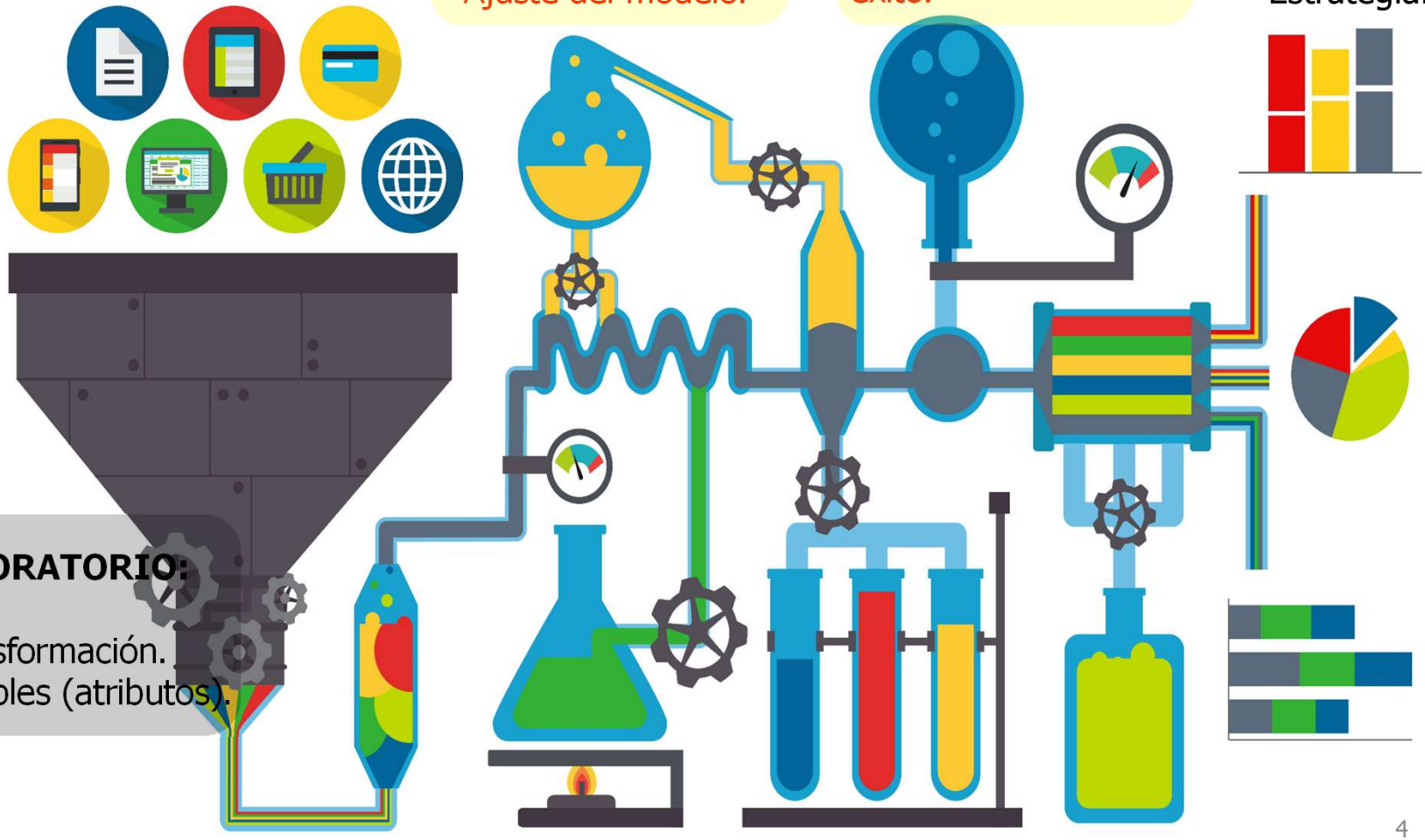
DATOS:
Materia prima.

MODELO:
Implementación del modelo de analítica.
Ajuste del modelo.

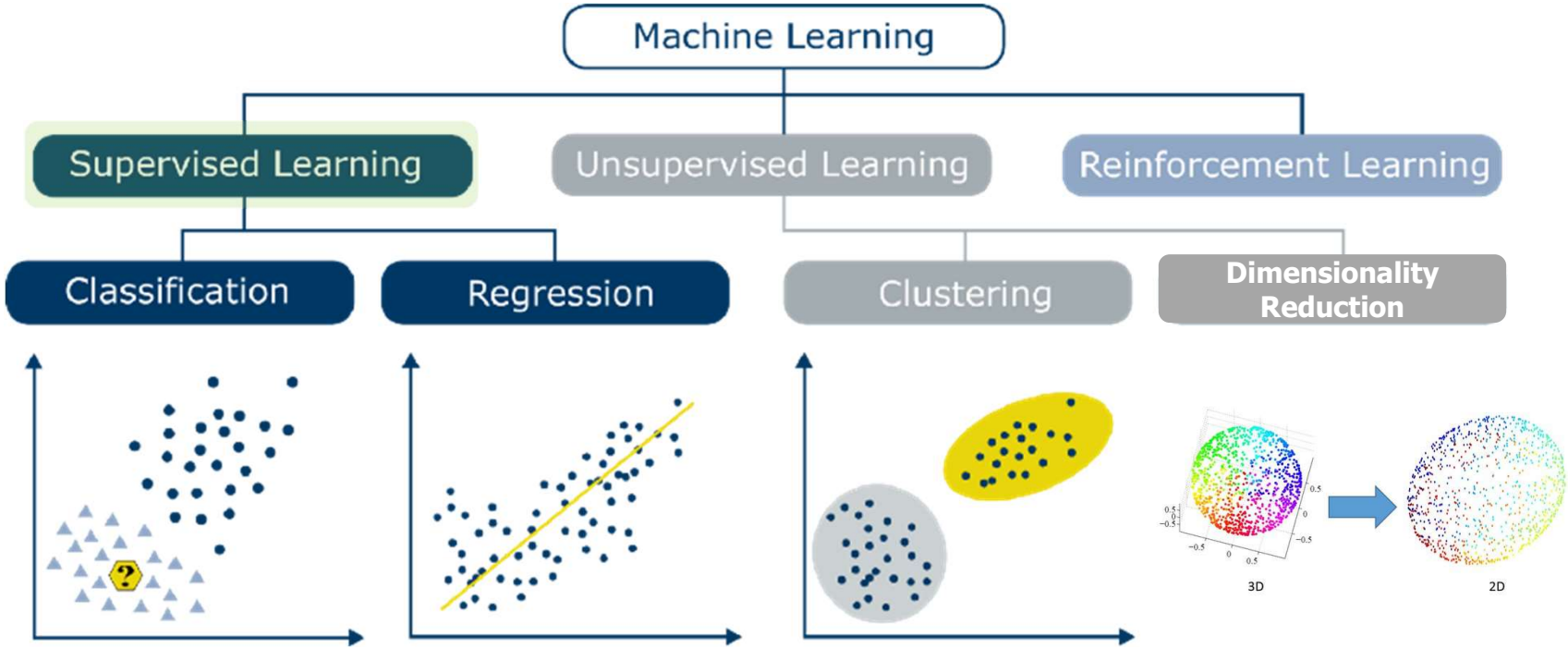
EVALUACION:
Viabilidad del negocio.
Validación criterios de éxito.

DESPLIEGUE:
Resultados.
Conocimiento.
Estrategia.

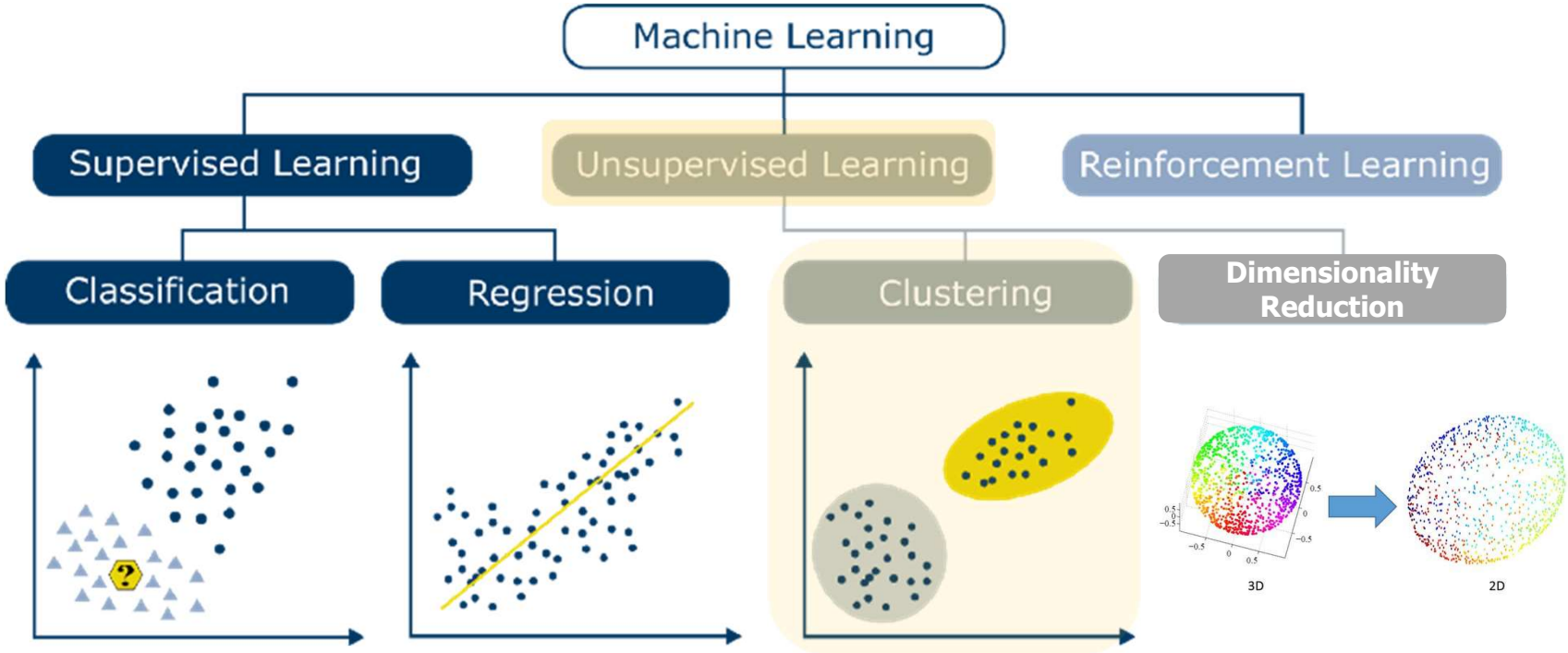
PREGUNTAS:
¿Cómo?
¿Cuáles?
¿Cuándo?
¿Por qué? *



ANALISIS EXPLORATORIO:
Limpieza de datos.
Preparación / transformación.
Selección de variables (atributos).

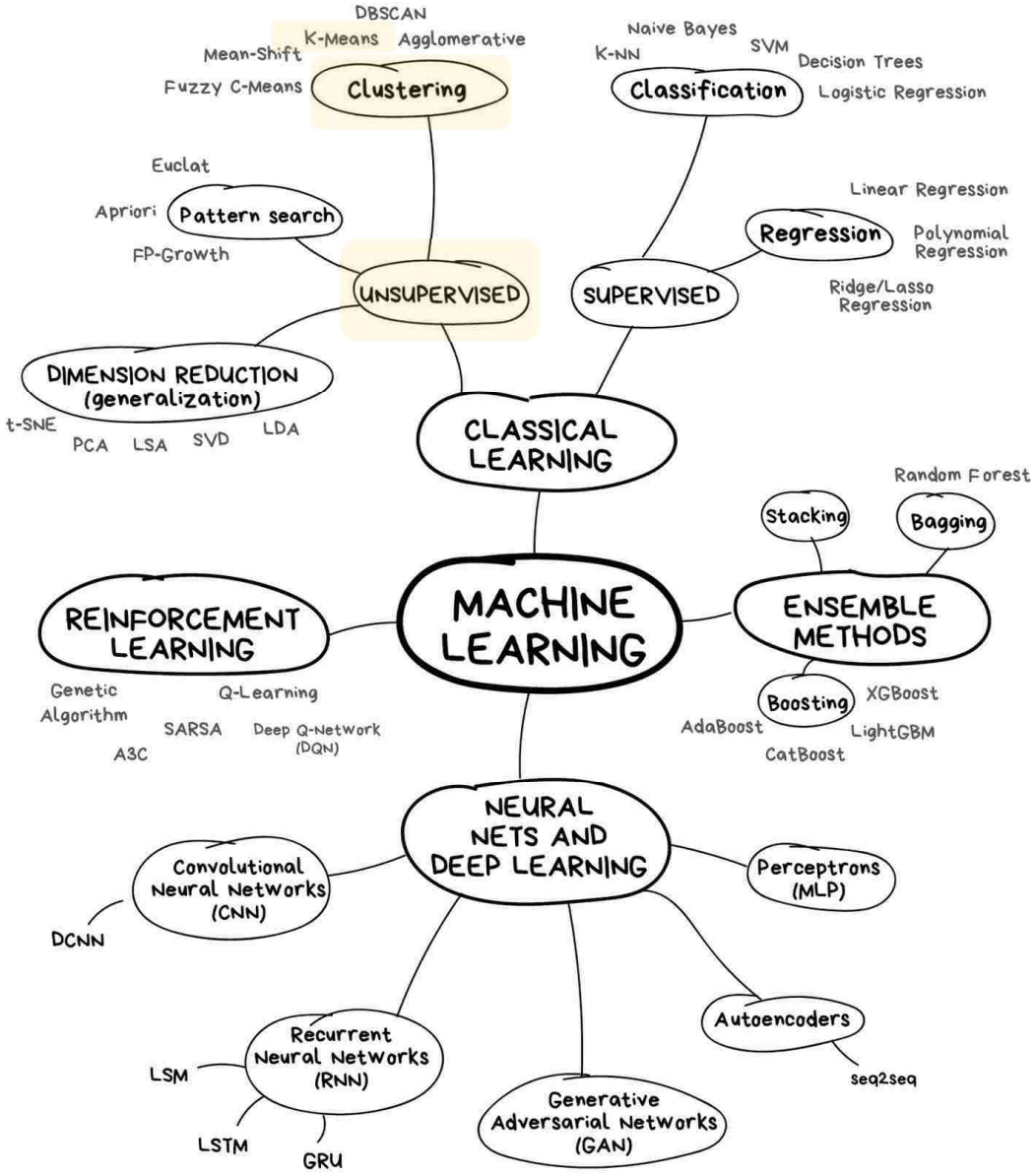


Métodos	KNN	Regresión Lineal Simple Regresión Lineal Múltiple Regresión Polinomial
Evaluación	Accuracy, Precision, Recall, F1 score, ROC, etc.	MSE, RMSE, R^2, etc.



Métodos	KNN	Regresión Lineal Simple Regresión Lineal Múltiple Regresión Polinomial
Evaluación	Accuracy, Precision, Recall, F1 score, ROC, etc.	MSE, RMSE, R^2, etc.

Desde el punto de los algoritmos



2. Aprendizaje No Supervisado

Aprendizaje No Supervisado

- Datos **no** están **etiquetados:** (x_1, x_2, …, x_n)

- El interés en el aprendizaje no supervisado está en descubrir estructura en los datos, un punto de vista nuevo, una simplificación o un resumen.

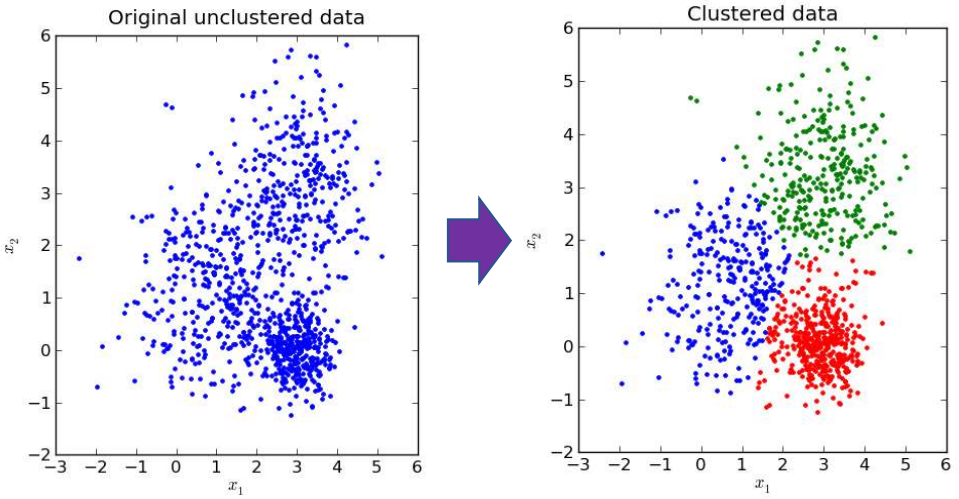
- Algunas tareas de interés incluyen:
  - Agrupamiento o segmentación (*clustering*).
  - Cambio de representación (e.g.: reducción de dimensiones, selección de factores).
  - Reglas de asociación.
  - Detección de anomalías (i.e.: excepciones).

- Es difícil validar los resultados obtenidos ya que no se tiene una salida deseada (e.g.: *ground-truth*" o "*gold standard*").

3. Agrupamiento (*clustering*)

Agrupamiento

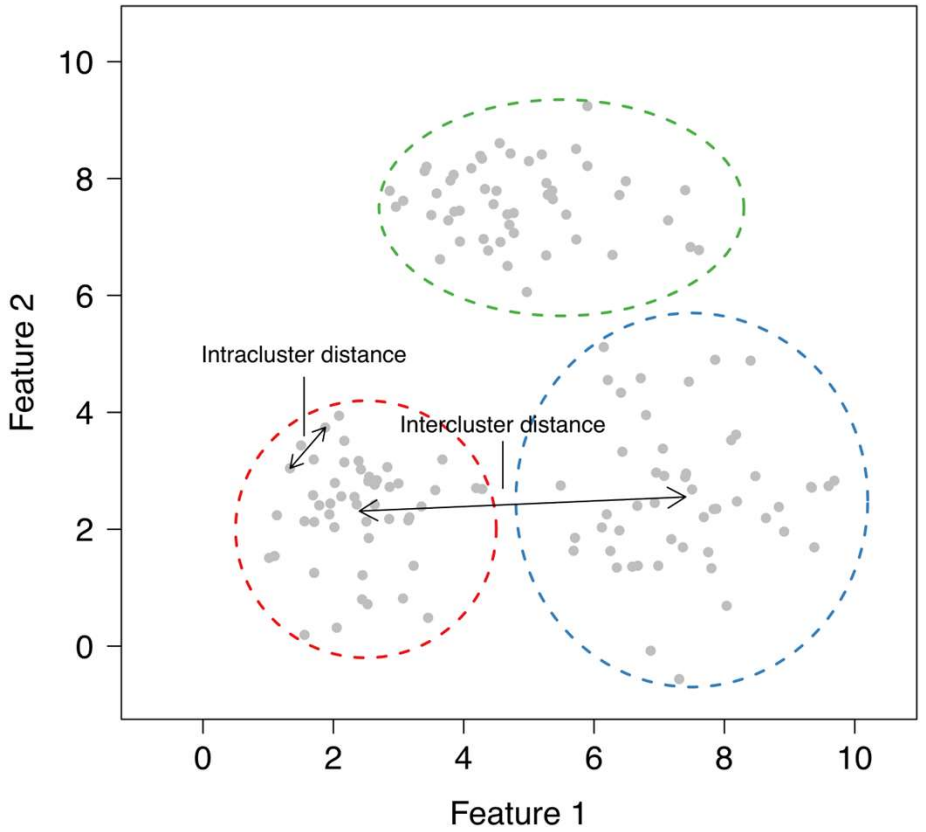
- No se tiene una variable de salida deseada.
- Se busca agrupar los datos similares para encontrar patrones globales en los datos.
- Se puede hacer un agrupamiento por similitud, proximidad, densidad.
- Se busca particionar un conjunto heterogéneo en *k* grupos de forma que los elementos de un grupo sean similares entre sí y diferentes a los elementos en otros grupos.
- Aplicaciones:
  - ✓Segmentación del mercado.
  - ✓Organización de un clúster de computadores.
  - ✓Análisis de redes sociales.
  - ✓Análisis de datos astronómicos.
  - ✓Etc. …



Fuente: http://pypr.sourceforge.net/kmeans.html

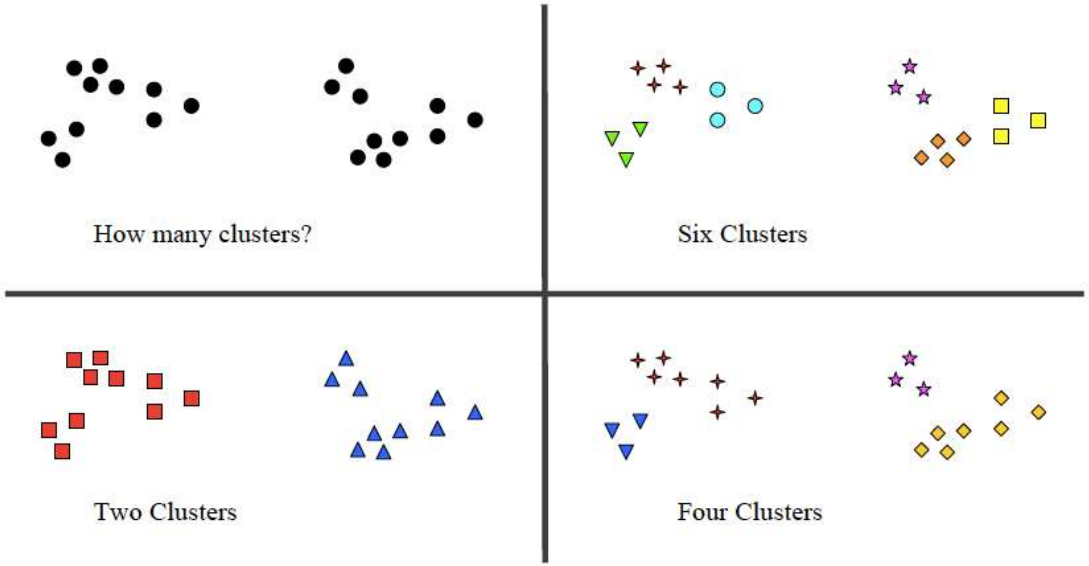
Agrupamiento por distancia [1/3]

- **Objetivo:** descubrir k grupos o segmentos desconocidos que:
  - ✓Minimicen la distancia dentro de los grupos.
  - ✓Maximicen la distancia por fuera de los grupos.
- Se basan en una noción de distancia**:**
  - ✓Se debe definir la medida a utilizar.
  - ✓Las unidades de los atributos tienen gran influencia, por ello se recomienda normalizarlos, o estandarizarlos.



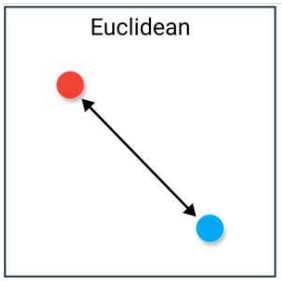
Agrupamiento por distancia [2/3]

- Se pueden buscar grupos de observaciones o de atributos usando los mismos algoritmos.
- No existe un método universal para definir el valor de ***k***, sólo heurísticos.
- El proceso requiere juicio humano y es difícil de automatizar.
- La interpretación de los resultados no se debe hacer de manera absoluta, sino como un punto de partida para un análisis posterior.
- Es posible que los datos no tengan una estructura, por lo cual su agrupación puede carecer de sentido.

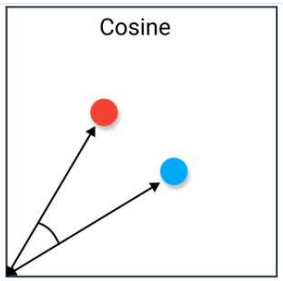


Fuente: http://governingstochastic.weebly.com/blog/category/clustering

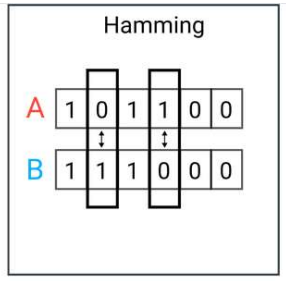
Agrupamiento por distancia [3/3]



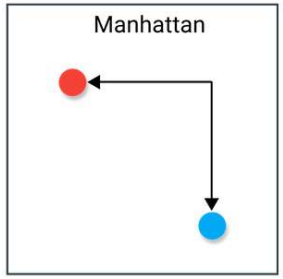
$$D(x,y) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}$$



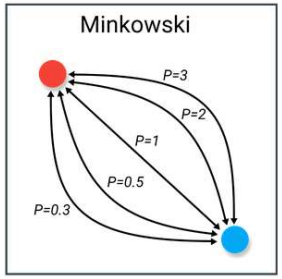
$$similarity(x,y) = \cos(\theta) = \frac{x \cdot y}{\|x\|\|y\|}$$



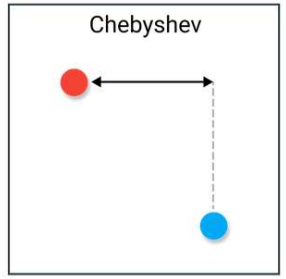
Número de valores diferentes entre dos vectores de la misma longitud.



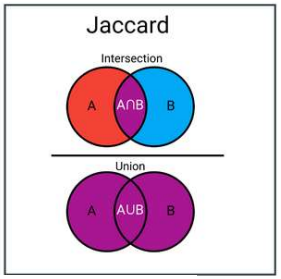
$$D(x,y) = \sum_{i=1}^{k}|x_i - y_i|$$



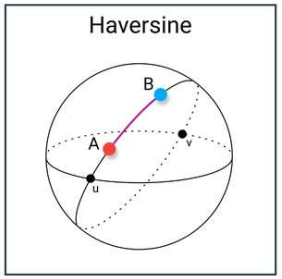
$$D(x,y) = \left(\sum_{i=1}^{n}|x_i - y_i|^p\right)^{\frac{1}{p}}$$



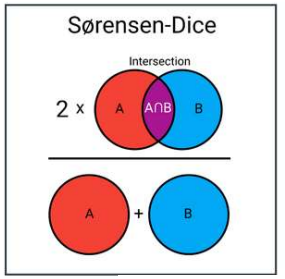
$$D(x,y) = \max_i(|x_i - y_i|)$$



$$D(x,y) = 1 - \frac{|x \cap y|}{|y \cup x|}$$



$$d = 2rarcsin\left(\sqrt{sin^2\left(\frac{\varphi_2 - \varphi_1}{2}\right) + cos(\varphi_1)cos(\varphi_2)sin^2\left(\frac{\lambda_2 - \lambda_1}{2}\right)}\right)$$

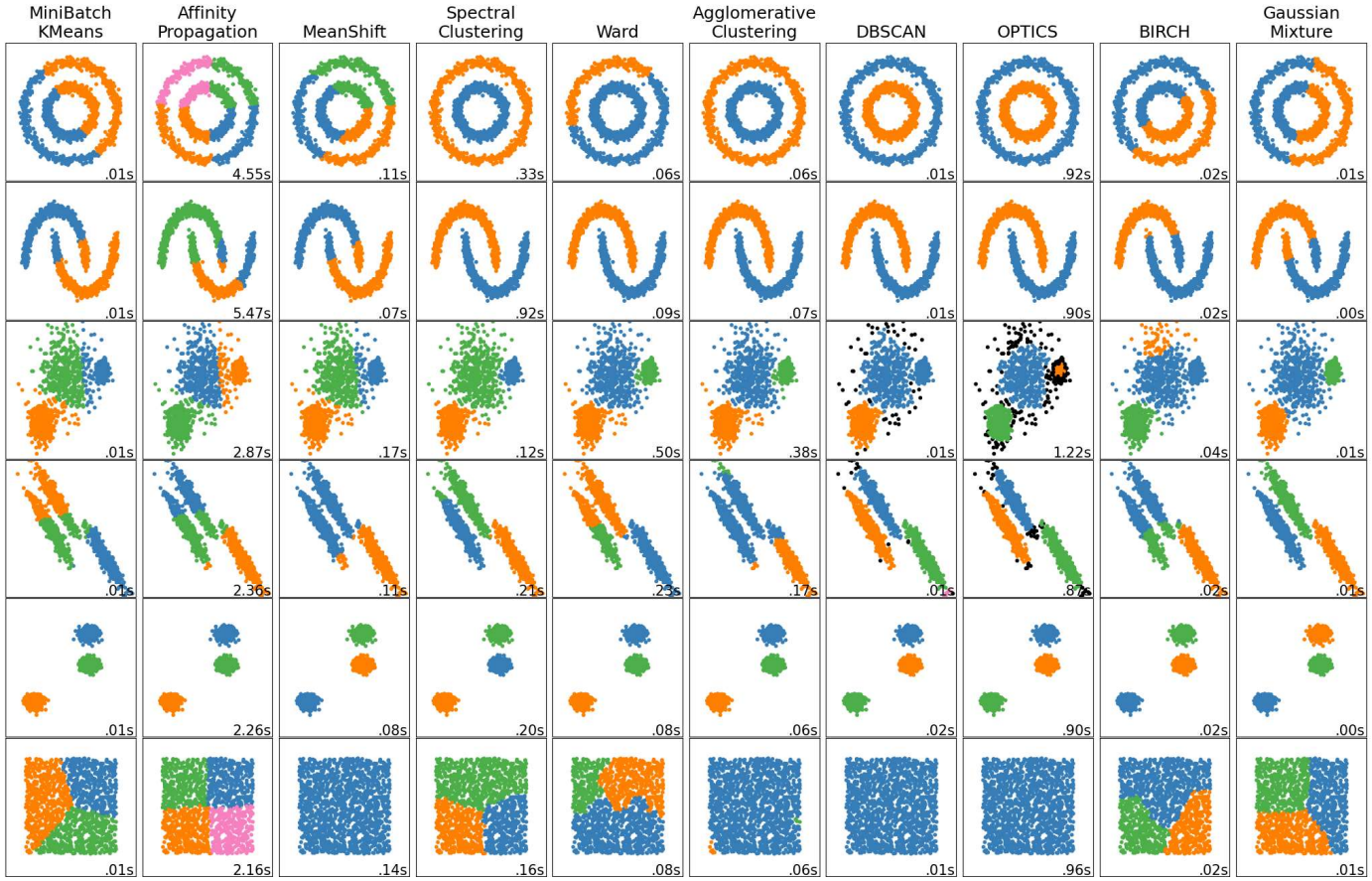


$$D(x,y) = \frac{2|x \cap y|}{|x| + |y|}$$

Agrupamiento: Algoritmos

Notas: Los parámetros de cada uno de estos pares de "algoritmo" y "conjunto de datos" se ajustaron para producir buenos resultados de agrupamiento.

Aunque estos ejemplos brindan alguna intuición sobre los algoritmos, es posible que esta intuición no aplique a datos de alta dimensionalidad.



Fuente: https://scikit-learn.org/stable/auto_examples/cluster/plot_cluster_comparison.html

4. K-means

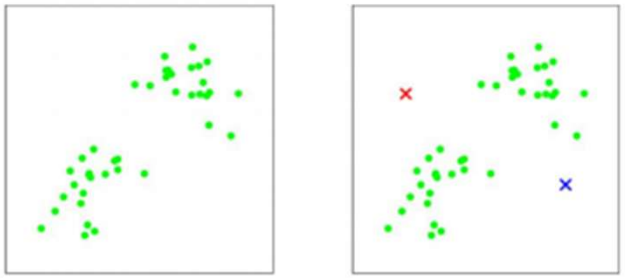
K-means: ¿Cómo funciona?

(a) Se tienen m puntos en el espacio n-dimensional.

(a)

K-means: ¿Cómo funciona?

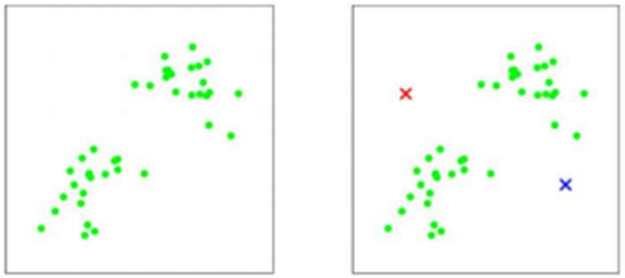
(a) Se tienen m puntos en el espacio n-dimensional.
(b) Se inicializan K centroides.



K-means: ¿Cómo funciona?

(a) Se tienen m puntos en el espacio n-dimensional.
(b) Se inicializan K centroides.

Para inicializar los centroides hay varias técnicas:

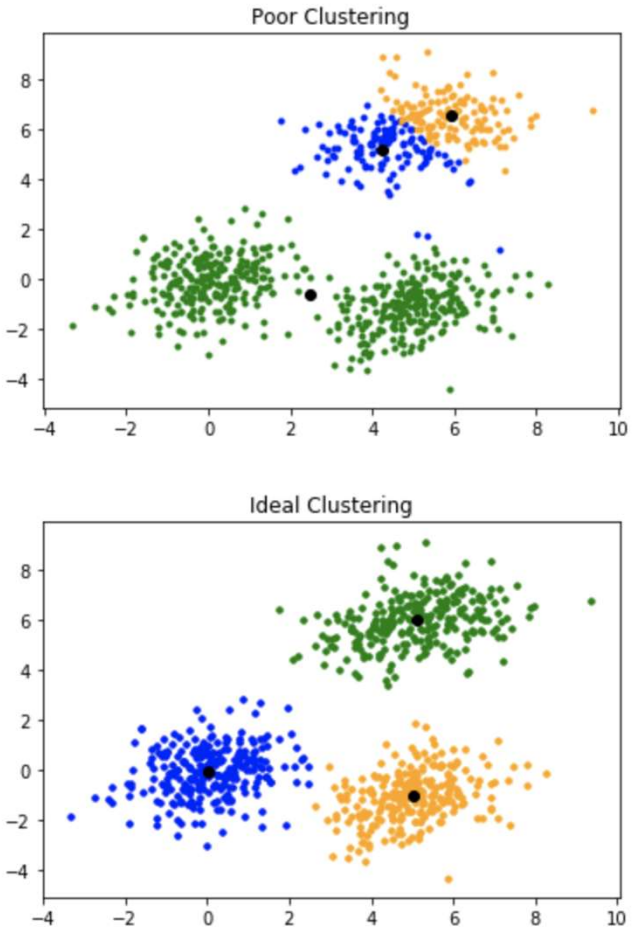


- De forma aleatoria tomando puntos en R^n, donde n es la dimensionalidad de las observaciones en los datos de entrada (número de atributos).
- Muestreando de forma aleatoria las **observaciones existentes**.
- Utilizando el algoritmo **K-means ++**.

K-means: ¿Cómo funciona?

(a) Se tienen m puntos en el espacio n-dimensional.
(b) Se inicializan K centroides.

K-means es muy sensible a la inicialización de los centroides.



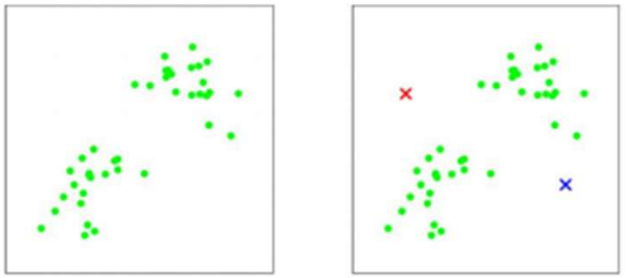
K-means: ¿Cómo funciona?

(a) Se tienen m puntos en el espacio n-dimensional.

(b) Se inicializan k centroides.

Para enfrentar el problema de una mala inicialización se utiliza el algoritmo **k-means++**.

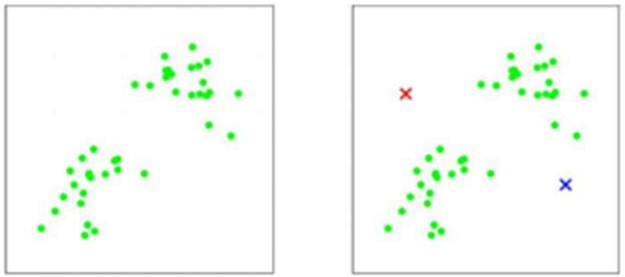
El objetivo de este algoritmo es inicializar los centroides lo más alejados posibles unos de otros.



K-means: ¿Cómo funciona?

(a) Se tienen m puntos en el espacio n-dimensional.

(b) Se inicializan K centroides.



Algoritmo **K-means++**.

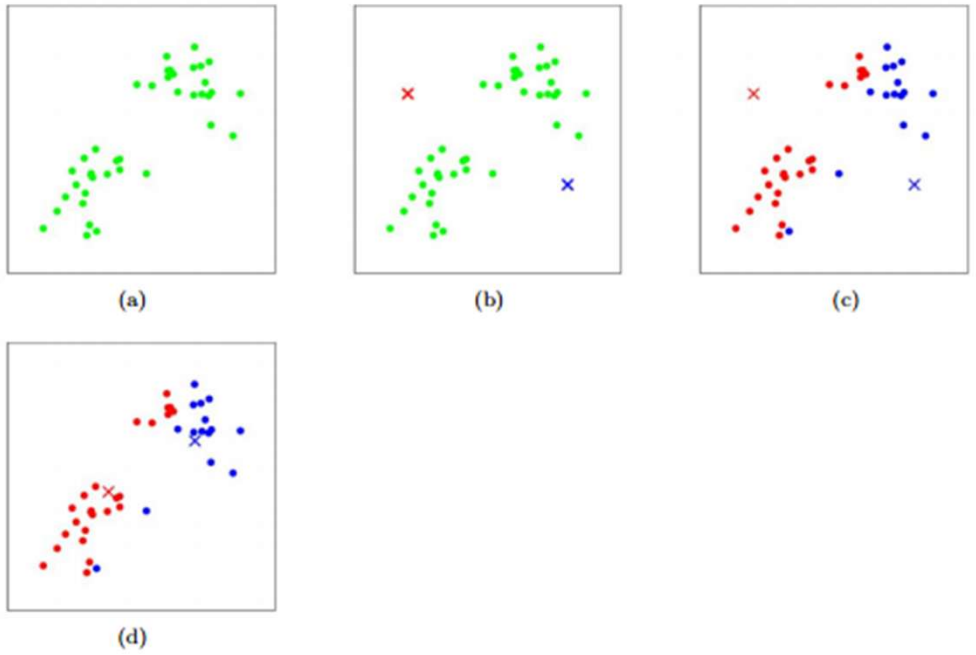
1. Se escoge aleatoriamente una observación como centroide.
2. Se calcula la distancia de cada observación al centroide escogido.
3. Se escoge de manera aleatoria una nueva observación como centroide, pero asociando una probabilidad a cada observación dada por la distancia calculada anteriormente.
4. Se repiten los pasos 3 y 4 hasta haber seleccionado *K* centroides.

K-means: ¿Cómo funciona?

(a) Se tienen m puntos en el espacio n-dimensional.
(b) Se inicializan K centroides.
(c) Se asigna cada observación en el conjunto de datos al grupo del centroide más cercano.

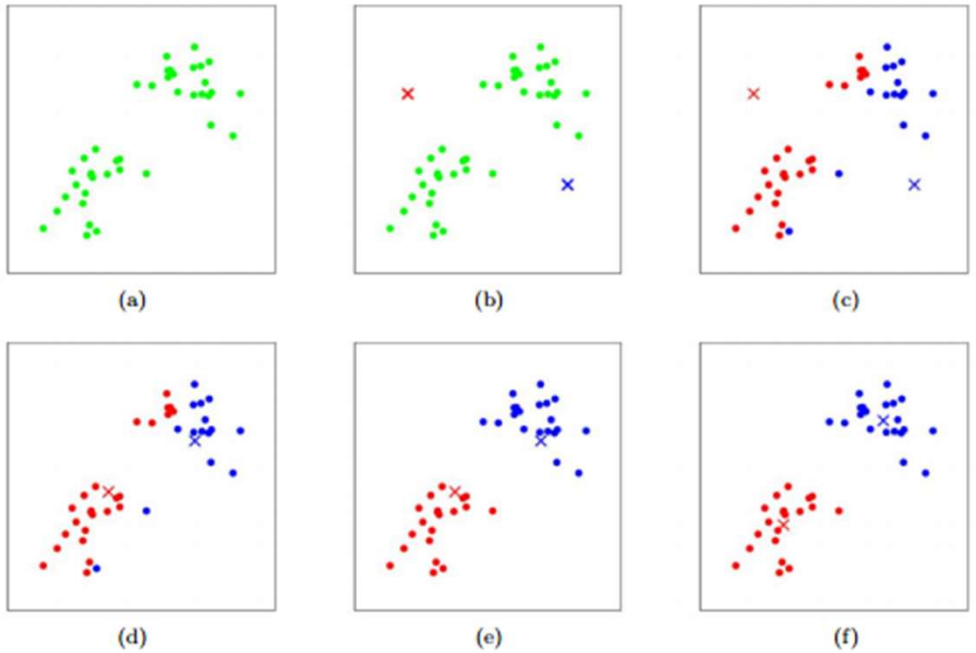
K-means: ¿Cómo funciona?

(a) Se tienen m puntos en el espacio n-dimensional.
(b) Se inicializan K centroides.
(c) Se asigna cada observación en el conjunto de datos al grupo del centroide más cercano.
(d) Se calculan nuevos centroides para cada grupo usando el promedio de las observaciones de cada grupo.



K-means: ¿Cómo funciona?

(a) Se tienen m puntos en el espacio n-dimensional.
(b) Se inicializan K centroides.
(c) Se asigna cada observación en el conjunto de datos al grupo del centroide más cercano.
(d) Se calculan nuevos centroides para cada grupo usando el promedio de las observaciones de cada grupo.
(e) Se repiten los pasos (c) y (d) hasta lograr la convergencia (i.e. hasta que los centroides dejen de moverse).
(f) Se obtienen K grupos.



K-means

Objetivo: minimizar la variación dentro de los grupos.

$$J = (1/m)\sum_{i=1}^{m} distancia(\boldsymbol{x_i} - centroide(\boldsymbol{x_i}))^2$$

Donde:

m: número de observaciones en el conjunto de datos.

$\boldsymbol{x_i}$: i-ésima observación.

$centroide(\boldsymbol{x_i})$

Notas:

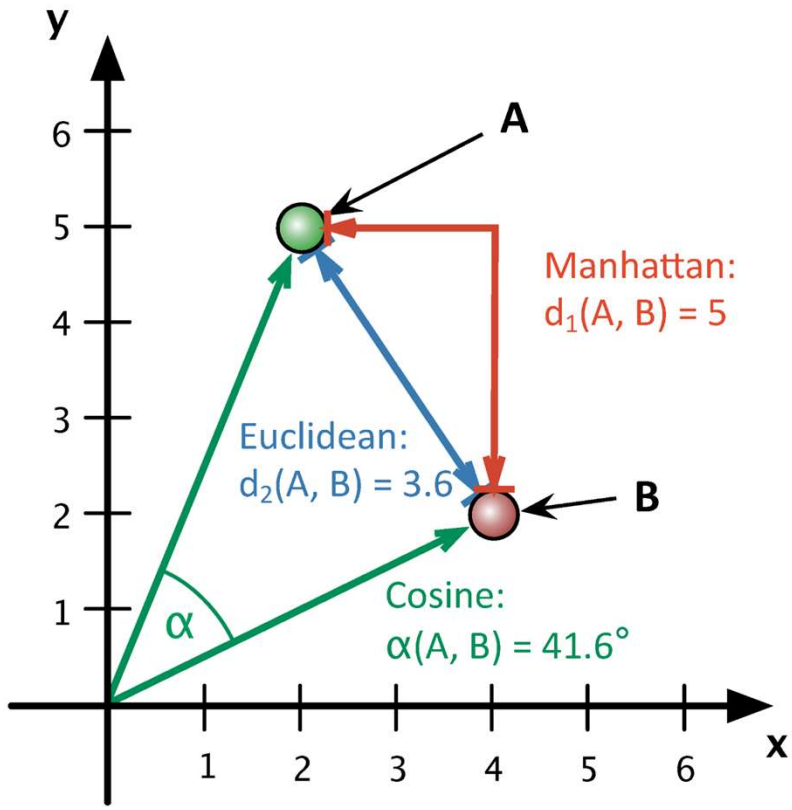
✓ Cada observación se asigna a un sólo grupo. Los grupos no se traslapan.

✓ Si un grupo queda sin observaciones asociadas, ese grupo se puede reinicializar de manera aleatoria, o se puede eliminar.

K-means: consideraciones [1/4]

¿Qué distancia escoger? ¿Euclidiana, Manhattan, otra?

R/ Depende del problema.



K-means: consideraciones [2/4]

Normalización o estandarización de atributos previo al uso de k-means:

Sea *x* un atributo de entrada:

Normalización con rango de salida [0,1]

$x_{max} = \max(x)$

$x_{min} = \min(x)$

Sii:

$(x_{max} - x_{min}) \neq 0$

Entonces:

$$x_{norm} = \left(\frac{x - x_{min}}{x_{max} - x_{min}} \right)$$

$x_{norm} \in [0,1]$

Normalización con rango de salida [a,b]

$a = \min(x_{norm})$

$b = \max(x_{norm})$

$x_{max} = \max(x)$

$x_{min} = \min(x)$

Sii:

$(x_{max} - x_{min}) \neq 0$

Entonces:

$$x_{norm} = a + \left(\frac{x - x_{min}}{x_{max} - x_{min}} \right)(b - a)$$

$x_{norm} \in [a,b]$

Nota: *a* y *b* se escogen con antelación.

Estandarización (*z-score*)

- Se parte de un supuesto de distribución normal.
- *x* es el valor actual del atributo.
- μ es la media aritmética del atributo *x*.
- σ es la desviación estándar del atributo *x*.
- *z* es la representación estandarizada.

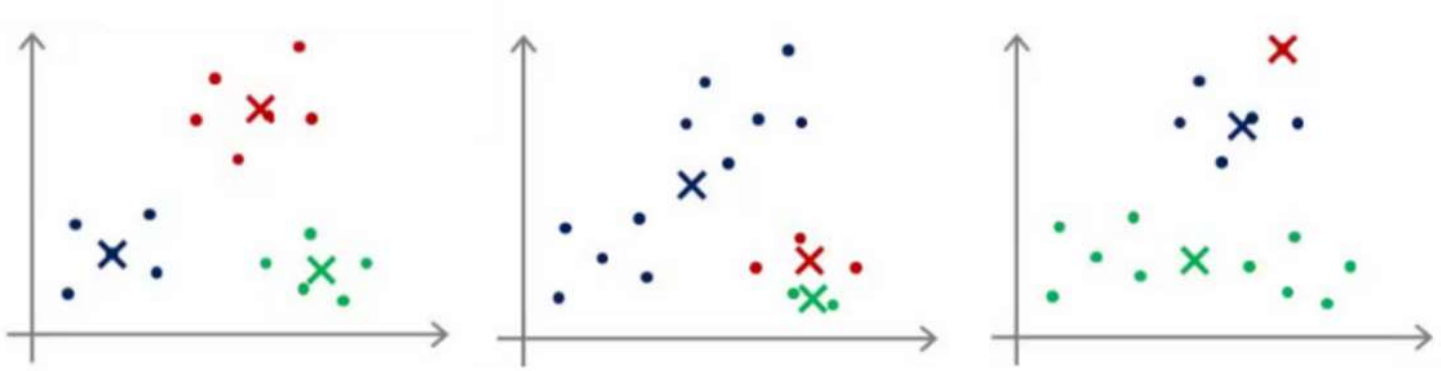
$$z = \frac{x - \mu}{\sigma}$$

Para σ diferente de 0.

K-means: consideraciones [3/4]

¿Cómo evitar los óptimos locales?

R/ Se puede ejecutar varias veces el algoritmo k-means con diferentes inicializaciones de centroides, y entonces seleccionar el agrupamiento que produce el mínimo valor de la función de costo J.



K-means: consideraciones [4/4]

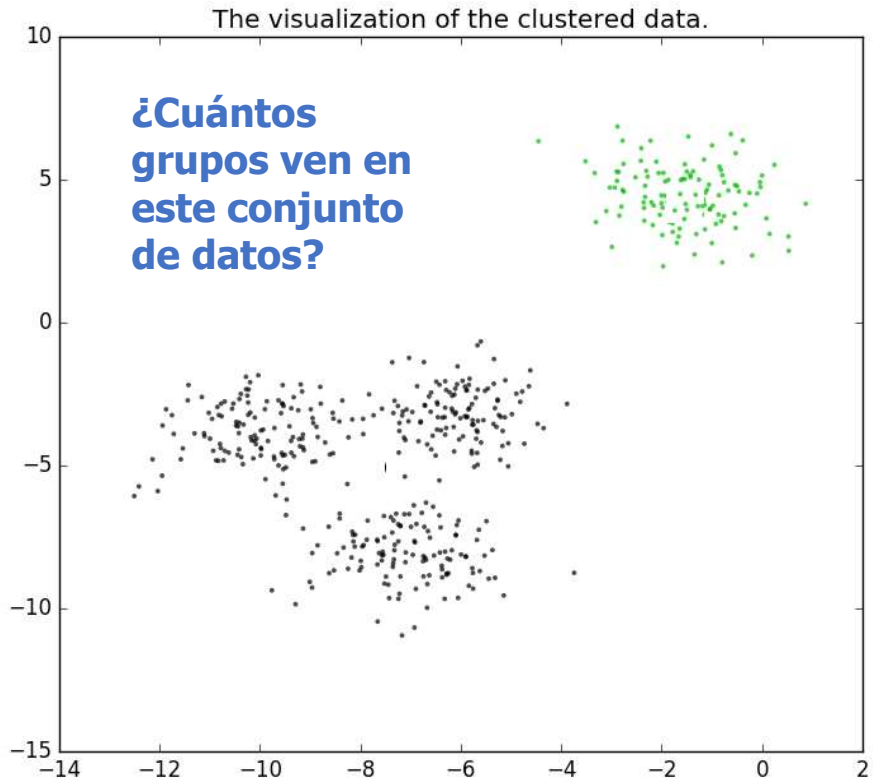
- Es un algoritmo de partición.
- Es muy fácil de implementar.
- Es más rápido que el agrupamiento jerárquico.
- Sólo trabaja con atributos numéricos pues requiere el cálculo del centroide.
- Es muy sensible a las anomalías en los datos (*outliers*).
- No sirve para identificar grupos con formas no convexas.

5. Evaluación del Agrupamiento

Selección de K

¿Cómo se estima el número de grupos *K*?

1. Método del codo (*elbow*).
2. Método de la silueta (*silhouette*) (sklearn.metrics.silhouette_score).
3. Índice Calinski-Harabasz. (sklearn.metrics.calinski_harabasz_score).



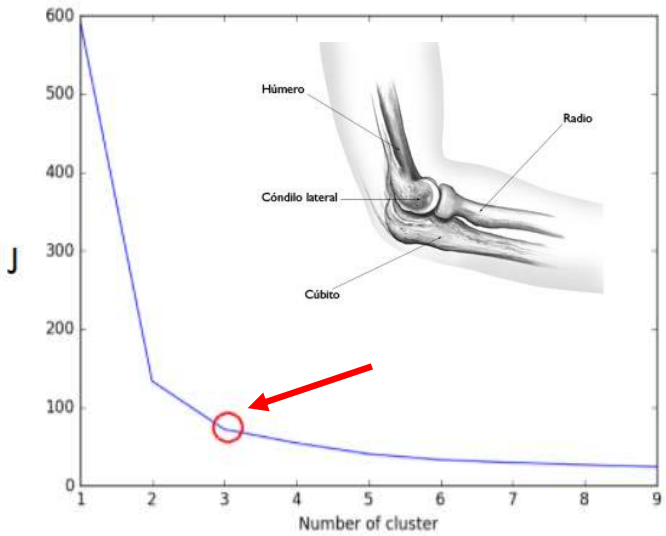
Fuente: http://scikit-learn.org/stable/auto_examples/cluster/plot_kmeans_silhouette_analysis.html

Selección de K: método del codo

- **Heurísticos:**
  - Existen diferentes métodos.
  - Dependen del juicio del analista, y requieren conocimiento del negocio.

- **Método del codo:**
  - Graficar una curva con los valores de la función de costo *J* para distintos valores de *K.*
  - Escoger el último valor de *K* que genera una reducción "significativa" en la curva.

$$J = (1/m) \sum_{i=1}^{m} distancia(\boldsymbol{x_i} - centroide(\boldsymbol{x_i}))^2$$



Selección de K: método de la silueta

$$silueta = \frac{1}{m}\sum_{p} s(p)$$

Valor promedio de la silueta de **todos los datos**. Se calcula como el promedio de los valores de las siluetas de las *m* observaciones en el conjunto de datos. Se calcula con sklearn.metrics.silhouette_score

$$silueta(C_i) = \frac{1}{m_i}\sum_{p \in C_i} s(p)$$

Valor promedio de la silueta del **grupo o *clúster* C_i**. Se calcula como el promedio de los valores de las siluetas de las observaciones en C_i. En esta expresión m_i representa el número de observaciones asociadas al grupo *i*.

Donde:

$$s(p) = silueta(p) = \frac{b(p) - a(p)}{\mathbf{max}(\, b(p),\ a(p)\,)}$$

El valor de silueta de **una observación *p***.
Se calcula con sklearn.metrics. silhouette_samples.

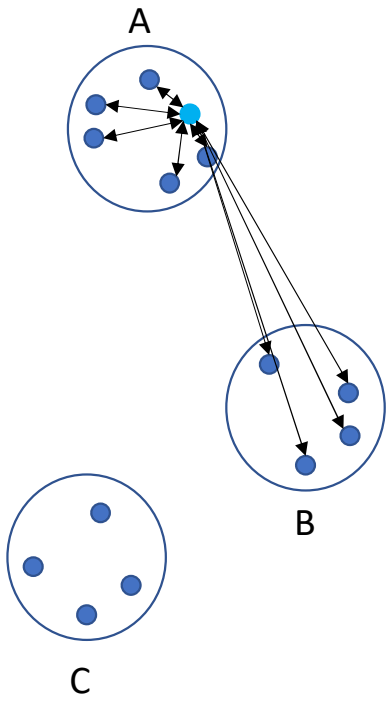
$$a(p) = cohesión(p) = \frac{\sum_{p' \in C_i,\ p' \neq p} distancia(p, p')}{m_i - 1}$$

Cohesión de la observación p con su grupo C_i. Se calcula como el promedio de las distancias a las otras observaciones p' de su grupo C_i. Note que m_i debe ser mayor que 1.

$$b(p) = separación(p) = \underset{C_j: 1 \leq j \leq k, j \neq i}{\mathbf{min}} \left(\frac{\sum_{p' \in C_j} distancia(p, p')}{m_j} \right)$$

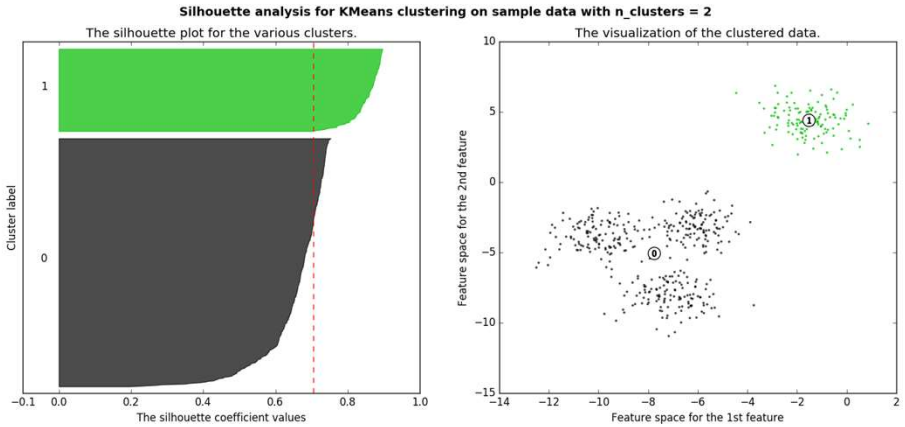
Separación de la observación p a otras observaciones p' en los demás grupos. Se calcula como la distancia promedio a los puntos del siguiente grupo más cercano. Aquí m_j representa el número de observaciones asociadas al grupo *j*.

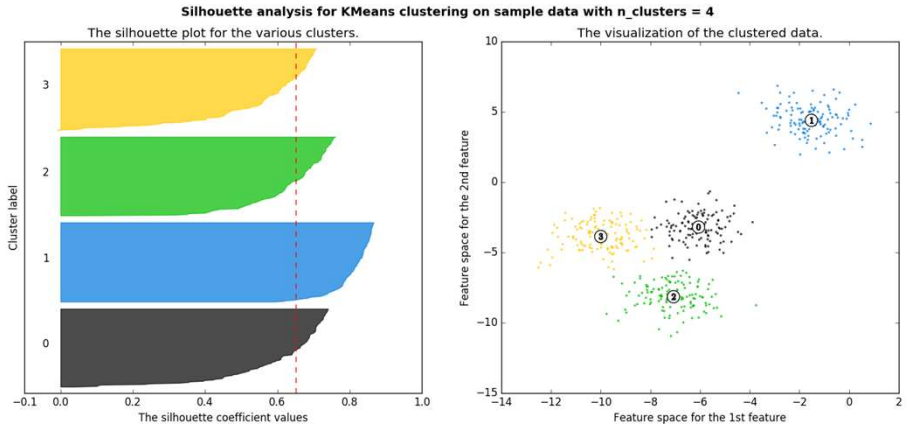
Selección de K: método de la silueta

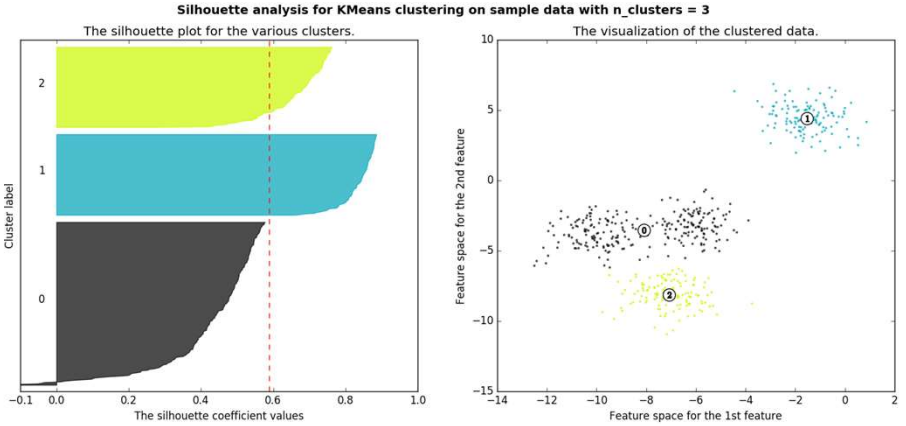


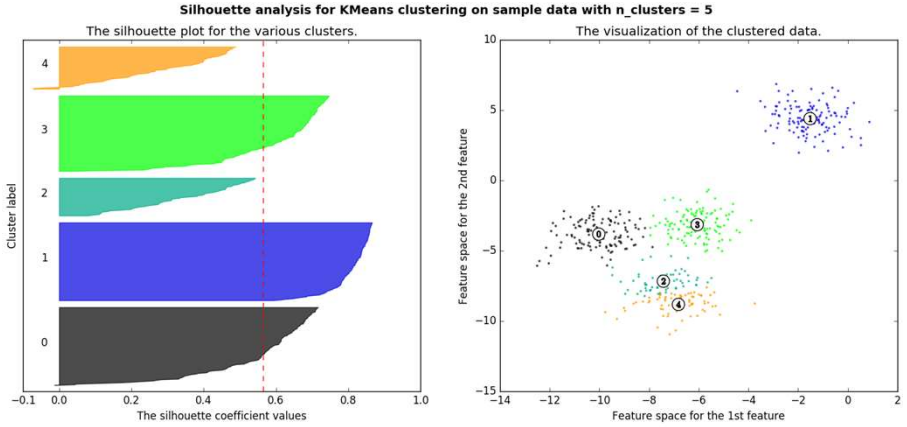
- Analiza las observaciones y los grupos, buscando posibles problemas de asignación dados por el valor del *K*:
  - ✓ El rango del valor de la silueta está entre -1 y 1.
  - ✓ Un valor de la silueta de 0 implica que la asignación de una observación a su grupo es indiferente.
  - ✓ Se espera que las observaciones del mismo grupo estén más cerca que las de otros grupos.
  - ✓ Para que la silueta sea positiva se requiere que la separación sea mayor que la cohesión, i.e.: $b(p) > a(p)$.
- Cómo interpretar los valores de la silueta para un grupo *i* dado?
  - ✓ 0.7 a 1.0: el grupo *i* es robusto.
  - ✓ 0.5 a 0.7: el grupo *i* es razonablemente robusto.
  - ✓ 0.25 a 0.5: el grupo *i* puede ser artificial y quizás no esté capturando la estructura.
  - ✓ <0.25: el grupo *i* debería descartarse, pues no está capturando la estructura.
- Se puede obtener una medida general de desempeño del agrupamiento calculando el promedio de los valores de la silueta de todas las observaciones.
- **Con el método de la silueta se busca maximizar el valor de silueta promedio.**

Selección de K: método de la silueta









Fuente: https://scikit-learn.org/stable/auto_examples/cluster/plot_kmeans_silhouette_analysis.html

Selección de K: método de Calinski-Harabasz (CH) o criterio de relación de varianza (Variance Ratio Criterion)

Para un conjunto de datos *E* con n_E **observaciones** que se ha agrupado en ***k*** **grupos**, el índice *s* de Calinski-Harabasz se define como la relación entre la "media de dispersión entre grupos" y la "dispersión dentro del grupo", donde la dispersión se define como la suma de distancias al cuadrado.

$$s = \frac{\mathrm{tr}(B_k)}{\mathrm{tr}(W_k)} \times \frac{n_E - k}{k - 1}$$

$$W_k = \sum_{q=1}^{k} \sum_{x \in C_q} (x - c_q)(x - c_q)^T$$

Matriz de dispersión dentro de los grupos.

$$B_k = \sum_{q=1}^{k} n_q (c_q - c_E)(c_q - c_E)^T$$

Matriz de dispersión entre grupos.

Donde:

tr: es la traza de la matriz.

C_q: es el conjunto de observaciones en el grupo *q*.

c_q: es el centro del grupo *q*.

c_E: es el centro del conjunto de datos *E*.

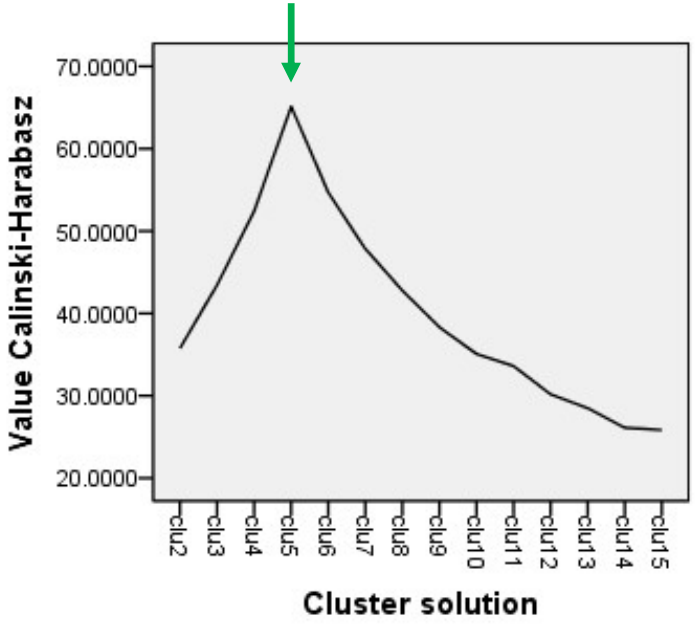
n_q: es el número de puntos en el grupo *q*.

Selección de K: método de Calinski-Harabasz (CH) o criterio de relación de varianza (Variance Ratio Criterion)

- Se busca el valor de *K* que maximice el valor del índice de Calinski-Harabasz.

  sklearn.metrics.calinski_harabasz_score

El valor del índice es más alto cuando los grupos son densos y están bien separados, lo que se relaciona con el concepto estándar de un grupo.



Lecturas Complementarias

- 9 Distance Measures in Data Science (https://towardsdatascience.com/9-distance-measures-in-data-science-918109d069fa)

- Silhouette (Clustering) https://en.wikipedia.org/wiki/Silhouette_(clustering)

- Método de Calinski-Harabasz

Caliński, T., & Harabasz, J. (1974). "A Dendrite Method for Cluster Analysis". Communications in Statistics-theory and Methods 3: 1-27. doi:10.1080/03610927408827101.