Curso

2

MACHINE LEARNING & BIG DATA

Unsupervised Learning: Clustering

# Algoritmos No-Supervisados

JOSÉ NELSON ZEPEDA DOÑO

# Cluster de Estudio: Advanced Analytics

Este material es el resumen de muchos autores que por medio de sus libros y documentos nos ofrecen fuentes riquísimas de conocimiento sobre los temas de Big Data y Machine Learning.
Algunas citas, figuras y tablas pueden ser encontradas de forma textual tal como lo indica el autor en su material original.

Nelson Zepeda
MIP • V 1.0
San Salvador El Salvador
Phone 503 79074137 • @nelsonzepeda733

# Tabla de Contenido


Algoritmos No supervisados ..... 1
Clustering ..... 4
- Transformación de los Datos ..... 5
- Transformación Lineal ..... 6
- Transformación con mínimos y máximos ..... 6
- Transformación con la Mediana y MAD ..... 7
- Transformación Logarítmica ..... 7

Fundamentos de Clustering ..... 7
- Algoritmo K-Means ..... 8
- Variantes del algoritmo K-means ..... 11
- Tipos de Distancias ..... 12
- Métricas de Validación Internas ..... 13
- Sugerencias prácticas de pre y post-procesamiento ..... 14
- Métodos Jerárquicos ..... 14
- Determinando el Número Adecuado de Clusters ..... 16

Bibliografía ..... 18

Capítulo 1

# Algoritmos No supervisados

*Machine Learning es una rama de la Inteligencia Artificial cuyo objetivo es desarrollar técnicas/procesos para que las computadoras aprendan, este aprendizaje puede ser supervisado y no supervisado.*

Aprendizaje no supervisado es un método de Aprendizaje Automático donde un modelo es ajustado a las observaciones. Se distingue del Aprendizaje supervisado por el hecho de que no hay un conocimiento a priori[1].

El aprendizaje no supervisado es muy importante cuando se dispone de muestras sin etiquetas de clase, cuando el costo de etiquetarlas por un experto es alto o cuando los patrones pueden variar con el tiempo, por lo que es necesario primero procesar los datos para luego clasificar.

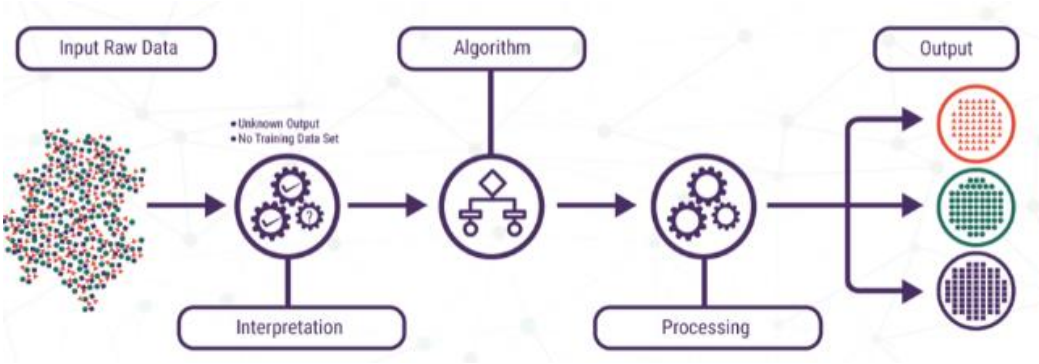


Figura 1-1 Aprendizaje No supervisado[2]

[1] https://es.wikipedia.org/wiki/Aprendizaje_no_supervisado

[2] https://datafloq.com/read/machine-learning-explained-understanding-learning/4478

La principal ventaja que presenta la clasificación no supervisada es que se puede obtener un conjunto de entrenamiento empleando muestras no etiquetadas valiéndose de algoritmos de agrupamiento, sin embargo, es necesario mencionar que el aprendizaje no supervisado es más subjetivo que el supervisando ya que su objetivo no se enfoca en una predicción o una respuesta.

Existen 2 tareas principales que pueden ser realizadas por los algoritmos no supervisados:

- Clustering: Los datos se agrupan en función de su similaridad.
- Reducción de Dimensionalidad: La data se comprime manteniendo su usabilidad y estructura.

Por sus características, el agrupamiento es una herramienta muy utilizada en distintos contextos como la Recuperación de Información y la Minería de Textos, el procesamiento de secuencias descriptoras de genes y proteínas, el seguimiento y detección de sucesos en un flujo continuo de noticias, la segmentación de imágenes, la compresión de datos, el procesamiento de bases de datos espaciales, la clasificación de zonas geográficas, la comprensión de imágenes de satélites, la visualización de datos, la prospección geológica, la organización de documentos en bibliotecas, y en muchas otras aplicaciones como la estructuración de grandes volúmenes de datos.[3].

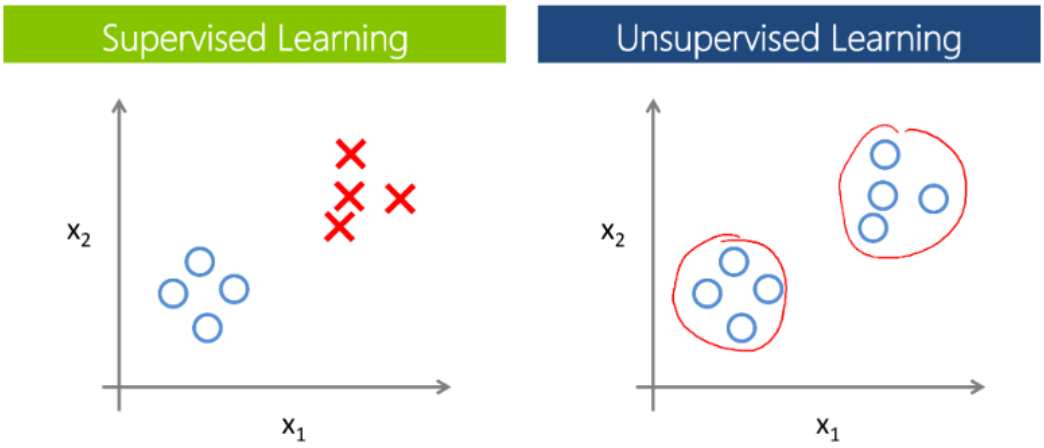


Figura 1-2 Supervisados vs No supervisados

A continuación se enumeran algunas aplicaciones de aprendizaje no supervisado:

1. Clustering: Permite dividir los datos en diferentes grupos en función de su similaridad.

[3] https://www.ecured.cu/Algoritmos_de_clasificaci%C3%B3n_no_supervisada

2. Deteccion de Anomalias: Permite descubrir observaciones o datos inusuales, es muy útil en la búsqueda de actividad fraudulenta.

3. Asociación: Identifica objetos u eventos que frecuentemente ocurren juntos, el análisis de canasta es el ejemplo principal.

4. Variables Latentes: Técnicas que se aplican durante la fase de pre-procesamiento tales como la reducción de variables en un dataset

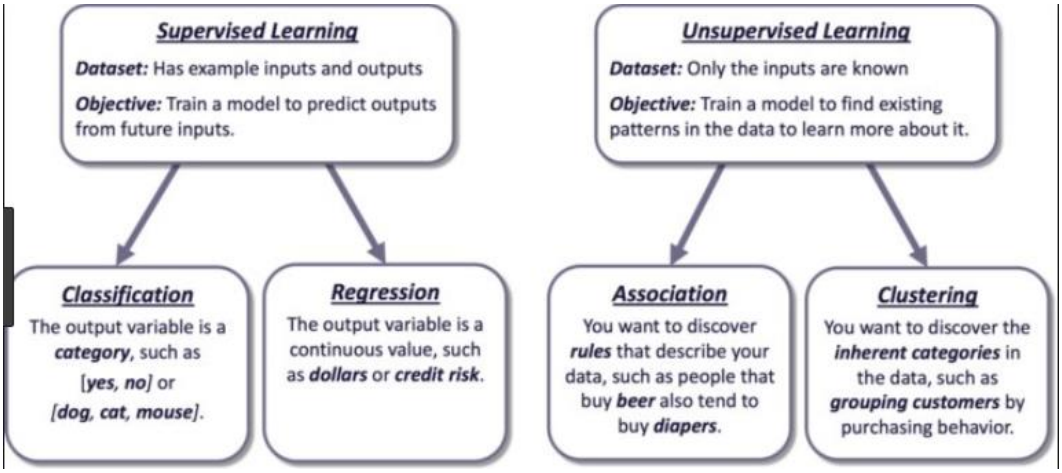


Figura 1-3 Supervisados vs No supervisados

Capítulo 2

# Clustering

*La identificación de grupos, permite descubrir y explicar algunos patrones ocultos en la data.*

El análisis de clusters, se considera una técnica de aprendizaje no supervisado ya que su objetivo es encontrar las relaciones entre las diferentes variables de estudio teniendo en cuenta que las relaciones descubiertas no están en función de ninguna variable target.

Los algoritmos de clusterización buscan cumplir con 3 requerimientos primordiales:

1. **Flexibilidad**: Se debe poder incluir atributos numéricos y categóricos.
2. **Robustez**: Estabilidad en los clusters ante cualquier ruido.
3. **Eficiencia**: Tiempos adecuados de procesamiento.

Históricamente, los algoritmos de clustering se han utilizado para generar segmentos que puedan ser interpretados, detección de outliers, agrupaciones durante la fase de pre-procesamiento, etc.

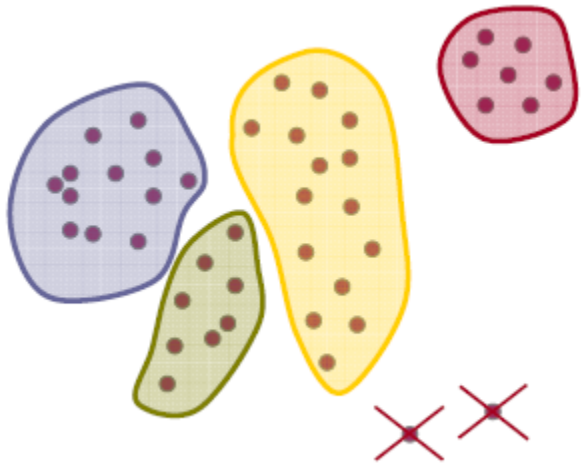

Figura 2-1 Detección de Outliers.

Típicamente, el proceso de clustering involucra 5 fases:

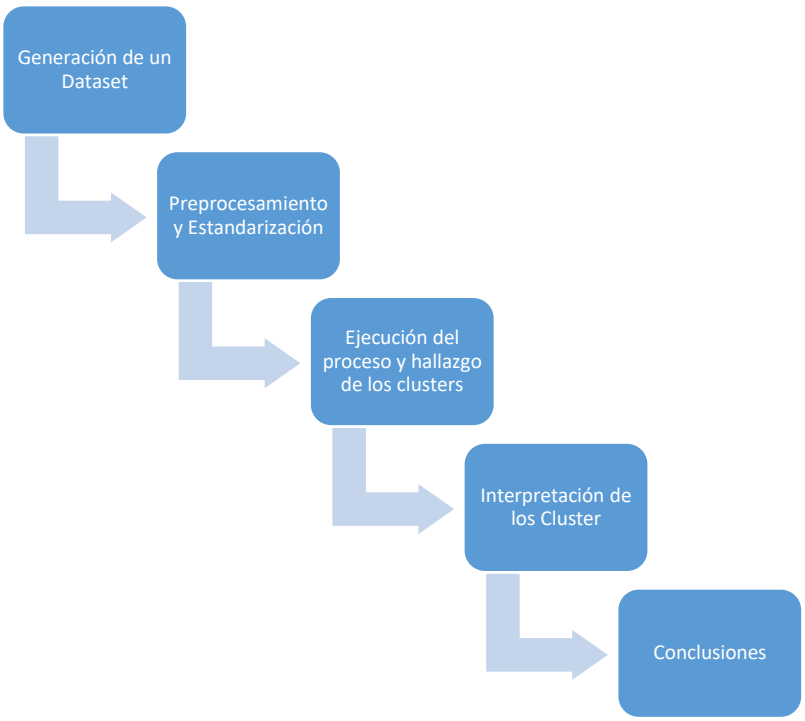


Figura 2-2 Proceso de Clustering

**Transformación de los Datos**

Actualmente, es normal que los analistas dediquen buena parte de su tiempo a comprender y jugar con los datos. Cuando la data es buena, la construcción de modelos será una tarea relativamente sencilla y también debemos tener en mente que cualquier mejora en los datos, traerá un impacto positivo sobre los modelos.

Usualmente para construir un modelo, se generan diferentes supuestos. En el caso de los modelos de clustering es muy importante que las variables numéricas estén expresadas en una escala similar.

Si las unidades entre las distintas variables son muy diferentes, el resultado obtenido no será el apropiado.

Por ejemplo, si estamos haciendo un análisis relacionado con personas y como atributos tenemos, el ingreso dólares y las edades de los sujetos, observaremos que la variable de ingreso tiende a ser mucho mayor y por ende ofusca la edad, y solo por poner un ejemplo, pueda ser que a nuestra organización le sea de mayor interés el grupo de personas jóvenes menores de 20 años, que aquellas que tienen salarios superiores a los diez mil dólares.

Si no aplicamos ninguna transformación, cuando el algoritmo de clustering compare las distancias, el modelo puede subestimar la importancia de la variable edad.

Para mitigar este problema de las diferentes escalas y naturaleza de las variables en la data, se pueden aplicar diferentes métodos de estandarización.

*La estandarización es el proceso de ajustar los datos a un rango específico.*

El siguiente grafico muestra visualmente lo que sucede cuando aplicamos una transformación y el rango oscilara entre 0 y 1

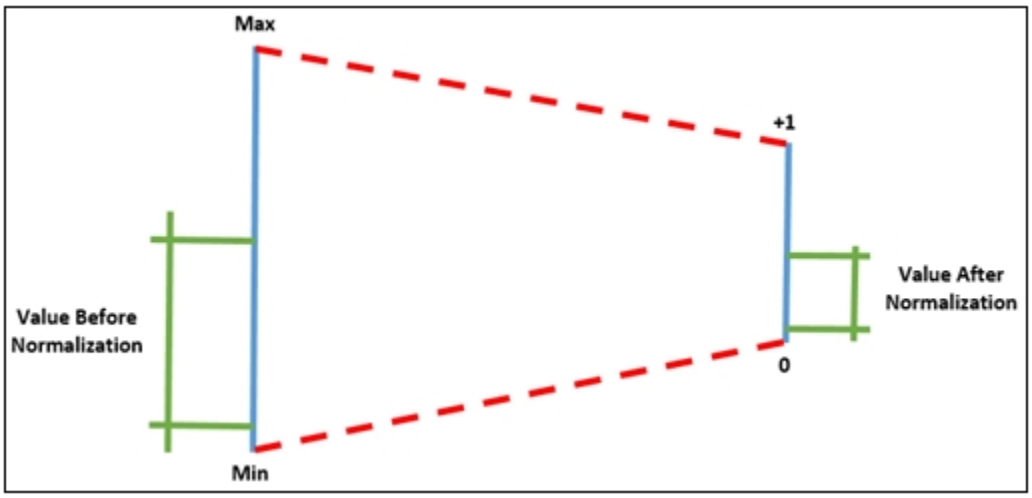


Figura 2-3 Transformación de una variable.

Vale la pena remarcar que la nueva data obtenida después del proceso de transformación ya no estará en con unidades de medidas, sin embargo, mantendrá las relaciones y proporcionalidad entre los diferentes atributos.

**Transformación Lineal**

El método más utilizado se conoce como el proceso de transformación lineal o recentralización (Recenter en Ingles), dicho proceso está basado en la aplicación de la transformación Z en donde se debe extraer primero la media y luego la desviación de cada variable.

*Recenter – Z score*

$$Z = \frac{X - \mu}{\sigma}$$

**Transformación con mínimos y máximos**

Este es otro método de transformación y su resultado nos devuelve un dataset cuyas variables numéricas están en el rango de 0 a 1. El cálculo se realiza mediante la resta del valor de cada observación menos el valor mínimo de esa variable dividido entre el rango de dicha variable.

*Scale [0-1]*

$$Z_i = \frac{X_i - min\,(x)}{max(x) - min\,(x)}$$

**Transformación con la Mediana y MAD**

Este método es más robusto que la transformación lineal (Z), se debe extraer la mediana de cada valor y luego hay que dividirlo entre la desviación absoluta media.

*Median / MAD*

$$Z = \frac{X - Me}{MAD}$$

En donde

$$MAD = median(|x_i - \tilde{x}|)$$

**Transformación Logarítmica**

Este método es muy utilizado cuando estamos ante escenarios cuya distribución de datos presentan un sesgo elevado ya sea a la izquierda o a la derecha.

Esta transformación puede requerir más trabajo ya que los valores que produce pueden tender al infinito.

# Fundamentos de Clustering

El clustering se basa en los conceptos de similaridad y distancia, en donde la proximidad entre los puntos es determinada por la función de la distancia.

Un aspecto importante a tener en cuenta es el hecho de que la persona que está haciendo el análisis debe indicar cuantos clusters se producirán y también deberá brindar soporte en la interpretación de los resultados.

Existen muchos métodos, pero los más populares son los algoritmos basados en clasificación jerárquica y el k-means.

A continuación se enumeran los diferentes métodos de clustering basados en su lógica de segmentación:

- Método por Partición: Los datos son divididos en un numero pre-calculado de clusters

- Método Jerárquico: Se construyen particiones basadas en una estructura de árbol.
- Métodos de Densidad: Construye los clusters tomando en cuenta la proximidad entre los diferentes puntos, es decir unifica entre vecinos.
- Método de Grilla (Grid): Construye las particiones basado en una estructura de grillas.

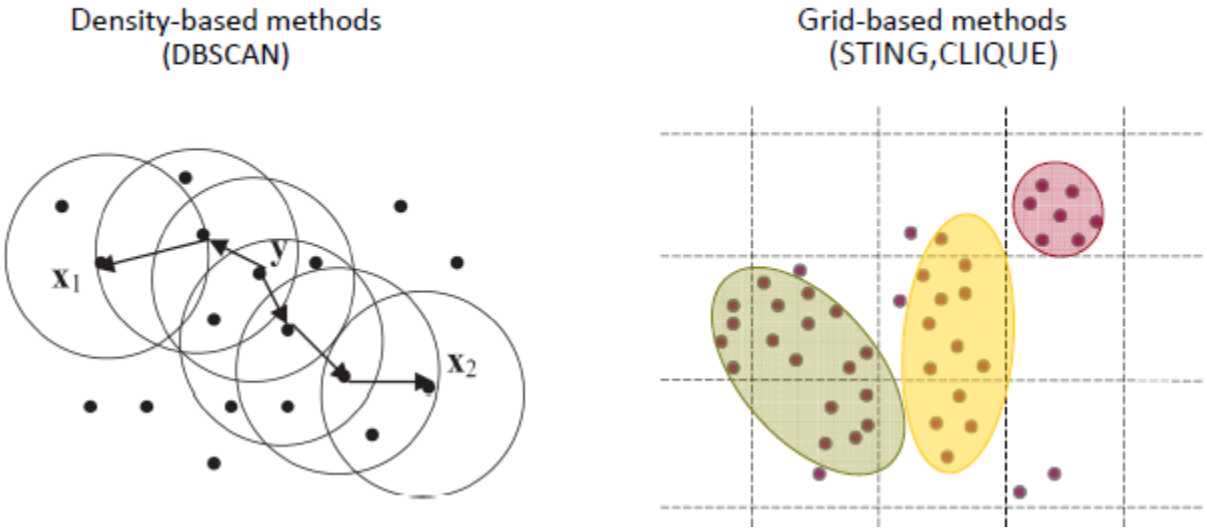


Figura 2-4 Representación gráfica para método de densidad y de grilla

**Algoritmo K-Means**

El algoritmo de las K-means (presentado por MacQueen en 1967) es uno de los algoritmos de aprendizaje no supervisado más simples para resolver el problema de la clusterización.

K-means es un método de agrupamiento, que tiene como objetivo la partición de un conjunto de n observaciones en k grupos en el que cada observación pertenece al grupo cuyo valor medio es más cercano[4].

El procedimiento aproxima por etapas sucesivas un cierto número (prefijado) de clusters haciendo uso de los centroides de los puntos que deben representar.

El algoritmo se compone de los siguientes pasos[5]:

[4] https://es.wikipedia.org/wiki/K-means

[5] http://www.cs.us.es/~fsancho/?e=43

1. Sitúa K puntos en el espacio en el que "viven" los objetos que se quieren clasificar. Estos puntos representan los centroides iniciales de los grupos.

2. Asigna cada objeto al grupo que tiene el centroide más cercano.

3. Tras haber asignado todos los objetos, recalcula las posiciones de los K centroides.

4. Repite los pasos 2 y 3 hasta que los centroides se mantengan estables. Esto produce una clasificación de los objetos en grupos que permite dar una métrica entre ellos.

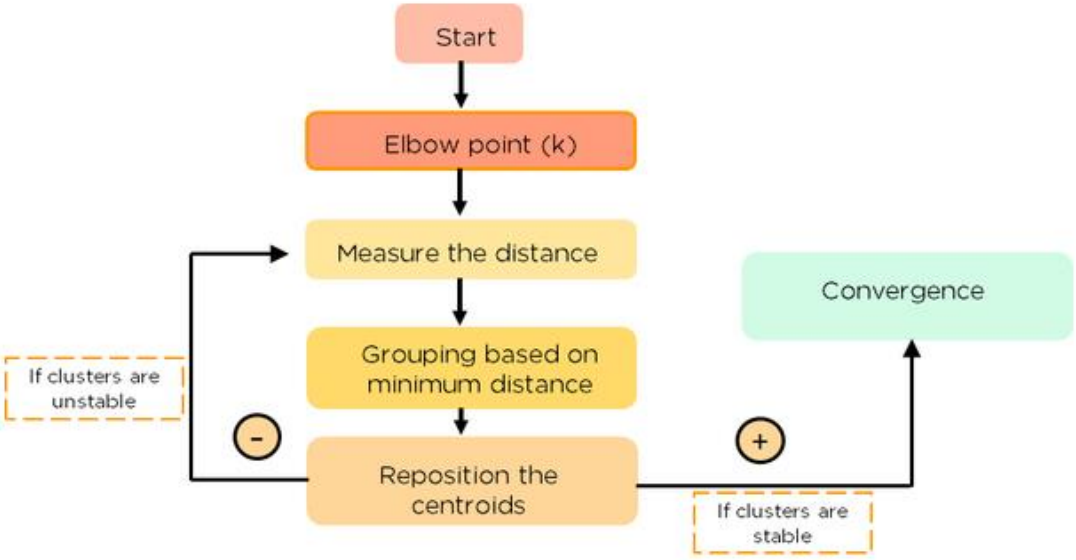


Figura 2-5 Representación de las Iteraciones en K-Means

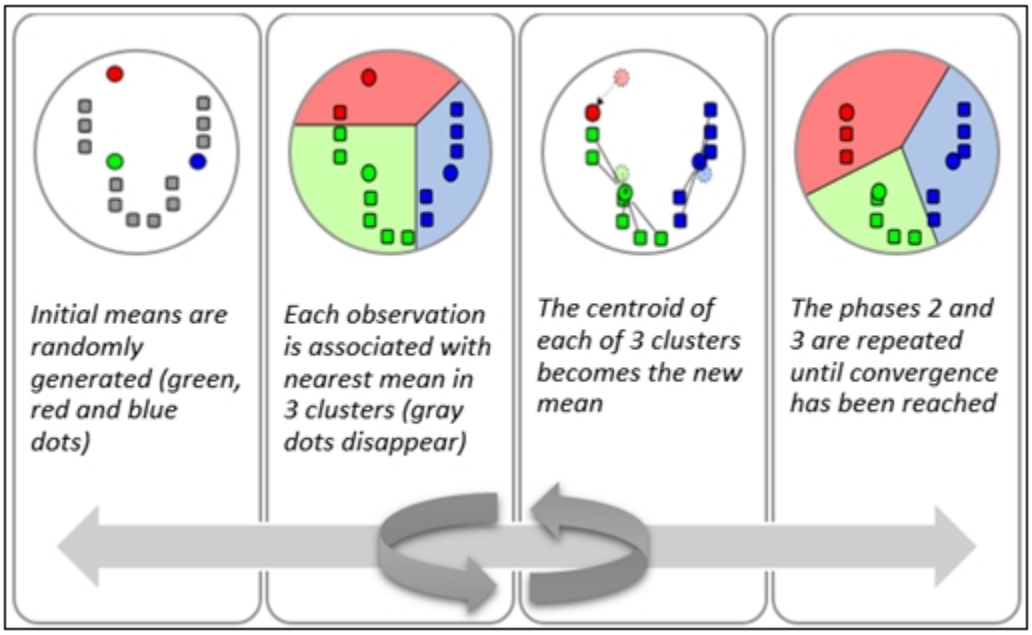


Figura 2-6 Representación de las Iteraciones en K-Means

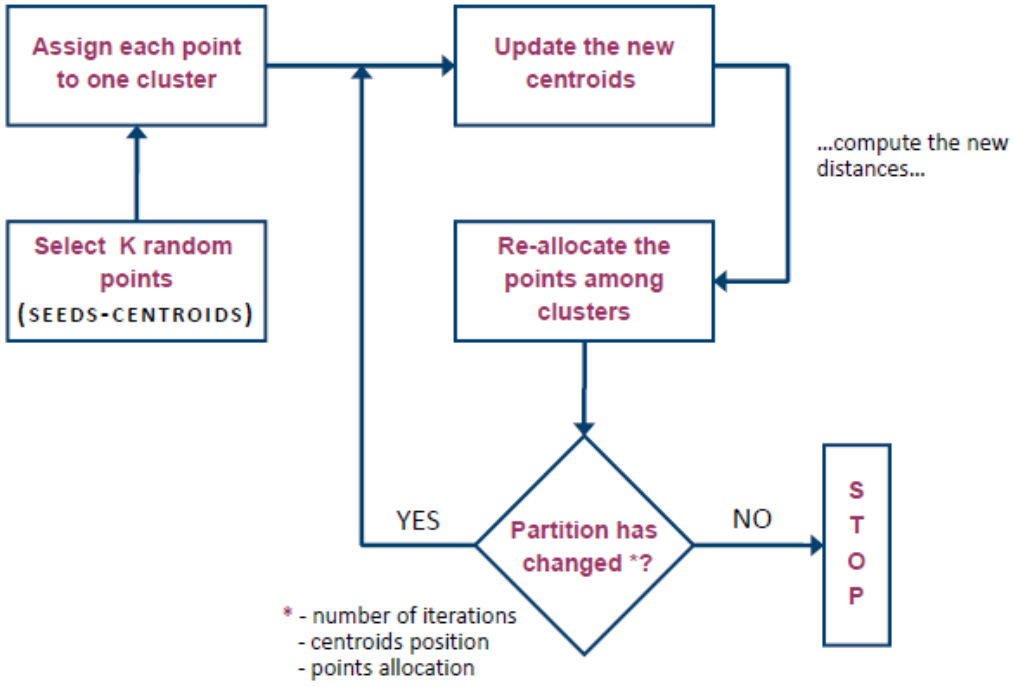


Figura 2-7 Algoritmo K-Means

Antes de proseguir, es necesario mencionar que el algoritmo k-means presenta las siguientes debilidades:

- El agrupamiento final depende de los centroides iniciales.
- La convergencia en el óptimo global no está garantizada, y para problemas con muchos ejemplares, requiere de un gran número de iteraciones para converger

El objetivo del algoritmo es encontrar la distancia mínima entre los centroides y las diferentes observaciones. Matemáticamente se describe mediante la siguiente ecuación:

$$\underset{S}{argmin} \sum_{i=1}^{k} \sum_{x \epsilon S_i} \| x - \mu_i \|^2$$

Where $\mu_i$ is the mean of points in $S_i$.

Ahora bien, dado que conocemos que los centroides iniciales son muy relevantes para este algoritmo, se cuentan con diferentes opciones de inicialización:

- Forgy
- Partición Aleatoria (Random Partition)

**Variantes del algoritmo K-means**

Las variantes del k-means son ramificaciones que particionan un conjunto de datos en clusters, a continuación se describen cada una:

- K –medianas: Este algoritmo funciona de forma similar al k-means y es también sensible a la selección de centroides iniciales, continua sustituyendo el valor de promedios por el vector de medianas del grupo de datos y utiliza una distancia manhattan como una medida de disimilitud.
- K-medoids: Fue introducido por Kaufman y Rousseeuw en 1987. Este algoritmo está basado en un conjunto de datos localizados muy en el centro de cada clusters, los puntos restantes del grupo son agrupados con el medoids más cercano. Iterativamente este algoritmo realiza intercambios entre los datos representativos y los que no lo son, hasta que minimice una diferencia entre lo k –medoids y los vectores que forman los clustering.
- Fuzzy c-means: Es un algoritmo que fue desarrollado para solucionar los datos que pueden pertenecer parcialmente a más de un clusters. FCM realiza una partición suave del conjunto de datos, un tipo de partición suave especial es aquella en la que la suma de los grados de pertenencia de un punto específico en todos los clusters es igual a 1

**Tipos de Distancias**

A pesar de que la más conocida es la distancia Euclideana, existen un total de 8 tipos de distancias:

| Distancia | Definición |
|---|---|
| Distancia Euclideana | Distancia proveniente de la raíz cuadrada entre 2 vectores. |
| Distancia Máxima | Distancia máxima entre 2 componentes de X y Y |
| Distancia Manhattan | Distancia absoluta entre 2 vectores |
| Distancia Canberra | Distancia Manhattan ponderada |
| Distancia Binaria | Los vectores son tratados como bits, si un elemento tiene valor se representa con un 1, de lo contrario son 0 |
| Distancia Pearson | Distancia de tipo Euclidea, conocida como Pearson No Centrada *sum(x_i y_i) / sqrt [sum(x_i^2) sum(y_i^2)]* |
| Distancia por Correlación | Tambien conocida como Pearson Centrada. 1 - corr(x,y) |
| Distancia Spearman | Calcula la distancia basada en un ranking |

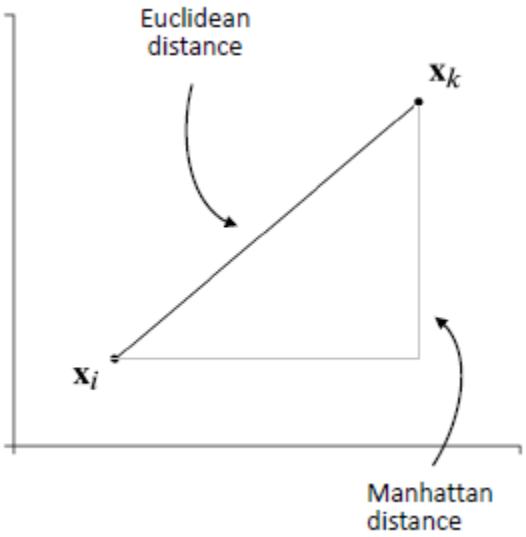


Figura 2-7 Distancias

**Métricas de Validación Internas**

Estas métricas miden el clustering únicamente basadas en información de los datos. Evaluan que tan buena es la estructura del clustering sin necesidad de información ajena al propio algoritmo y su resultado. Estas métricas de validación interna suelen ser utilizadas para escoger el numero óptimo de clusters, así como también el mejor algoritmo.

Las métricas de validación interna se basan en 2 criterios:

- **Cohesión:** El miembro de cada clúster debe ser lo más cercano posible a los otros miembros del mismo clúster, la afinidad entre las observaciones se puede medir mediante el coeficiente de Jaccard o bien el coeficiente de afinidad.

- **Separación:** Los clústeres deben estar ampliamente separados entre ellos. Existen varios enfoques para medir esta distancia entre clúster:

    - Distancia entre el miembro más cercano,

    - Distancia entre los miembros más distantes

    - Distancia entre los centroides.

Las métricas se listan a continuación:

- **Sum of Squared Within (SSW)**: Medida interna especialmente usada para evaluar la Cohesión de los clústeres que el algoritmo de agrupamiento generó.

$$SSE = \sum_{i=1}^{k} \sum_{x \epsilon C_i} dist^2(m_i, x)$$

- **Sum of Squared Between (SSB)**: Es una medida de separación utilizada para evaluar la distancia interclúster (Separación)

$$SSB = \sum_{j=1}^{k} n_j \, dist^2 \, (c_j - \overline{x})$$

Otros índices internos, que no están basados en la cohesión y separación son:

- **Indice de Davies-Bouldin (DB):** Valores pequeños para el índice DB indica clústeres compactos, y cuyos centros estas bien separados los unos de los otros. Consecuentemente el número de clústeres que minimiza el índice DB se toma como el óptimo.

$$DB = \frac{1}{k} \sum_{i=1, i \neq j}^{k} \max \left( \frac{\sigma_i + \sigma_j}{d(c_i, c_j)} \right)$$

- **Coeficiente de Silhouette:** Contrasta la distancia promedio de elementos en el mismo cluster con la distancia promedio de elementos en otros clusters. Los elementos con alto valor se consideran bien agrupados, mientras que objetos con medidas bajas se consideran outliers.

$$s(x) = \frac{b(x) - a(x)}{\max\{a(x), b(x)\}}$$

#### Sugerencias prácticas de pre y post-procesamiento

Pre-procesamiento:

- Estandarizar los datos
- Descartar los outliers

Post-procesamiento:

- Descartar los clusters pequeños que pueden representar outliers
- Re-clusterizar los clusters que son de gran tamaño
- Unificar los clusters que son cercanos

#### Métodos Jerárquicos

Los métodos jerárquicos tienen las siguientes características:

- Estan basados en una estructura de árbol (dendograma)

- Utiliza las distancias entre los diferentes puntos para unificar o particionar
- No se necesita el número de clusters como entrada del proceso

Los algoritmos a utilizar pueden ser de tipo aglomerativos o divisivos:

- **Aglomerativos**: Inicialmente cada observación representa un cluster, por cada iteración se van agrupando los 2 cluster más cercanos y el algoritmo se detiene cuando cada observación esta categorizada dentro de un cluster.
- **Divisivos**: Inicialmente todas las observaciones están dentro de un solo cluster, por cada iteración se generan 2 clusters en función de su distancia máxima, el algoritmo se detiene cuando cada observación representa un cluster único.

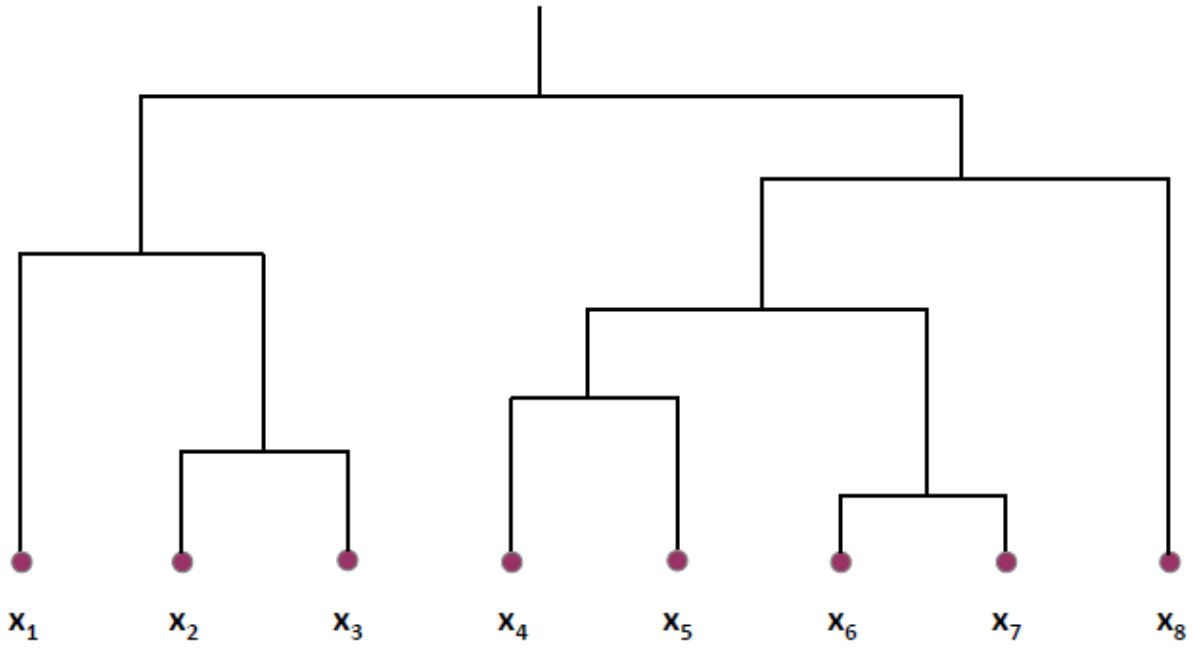


Figura 2-8 Método Jerárquico

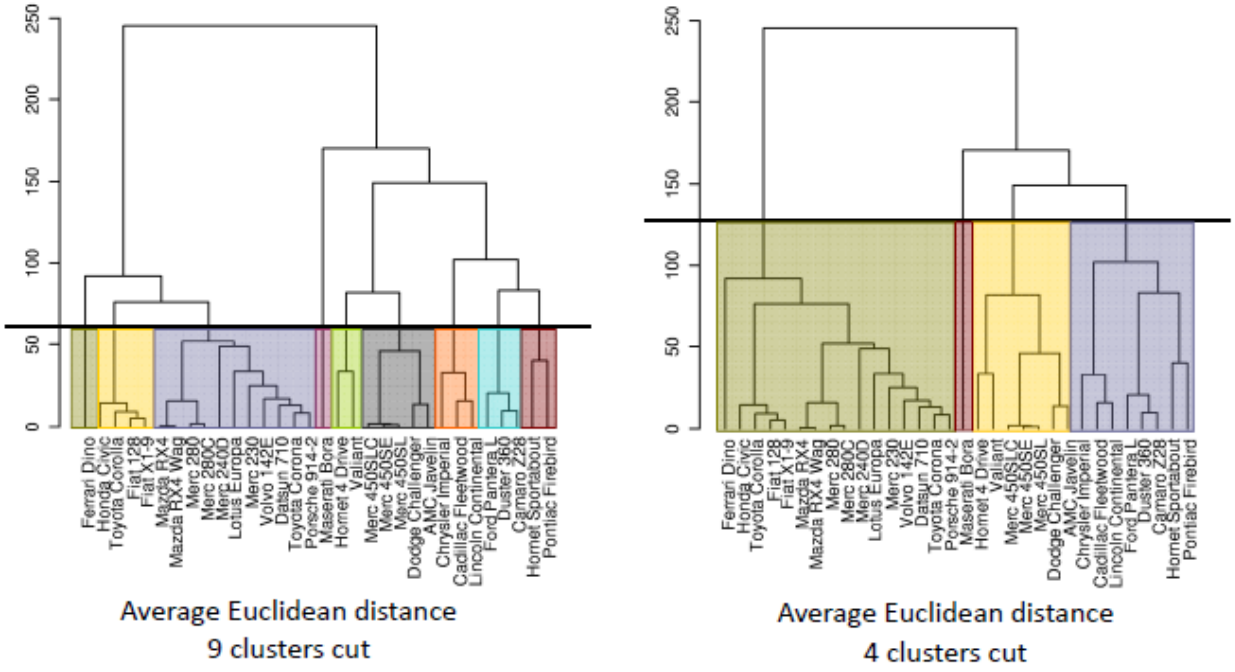


Figura 2-9 Método Jerárquico Aglomerativo

**Determinando el Número Adecuado de Clusters**

Tal como se mencionó en los párrafos iniciales, los algoritmos de clustering por agrupamiento (k-means) requiere como parámetro de entrada la cantidad de clusters en los que se necesita particionar la data. Las metodologías para estimar dicho número se describen a continuación:

- Regla del Pulgar: Corresponde a la raíz cuadrada del total de observaciones dividido por 2.

$$K \approx \left(\frac{m}{2}\right)^{1/2}$$

- Método del Codo (Elbow Method): El porcentaje de varianza que se puede explicar esta en función del número de clusters.

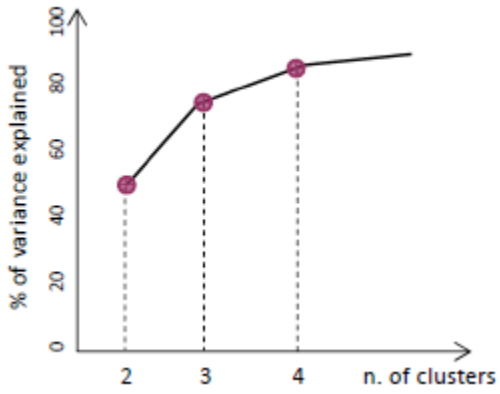


Figura 2-10 Método del Codo

- Método de Silhouette: Silhouette indica que tan similar es una observación con respecto a las demás observaciones del mismo clusters con respecto a otros clusters.

$$\text{silh}\left(\mathbf{x}_i\right) = \frac{v_i - u_i}{\max\left(u_i, v_i\right)}$$

- within [-1,1]
- the closer to 1 the better
- average silhouette

# Bibliografía


- Unsupervised Learning with R
  By Erik Rodríguez Pacheco, 2015

- Data Clustering
  By Charu C. Aggarwal; Chandan K. Reddy, 2016

- A Framework for Analysis of Data Quality Research
  by Richard Y. Wang,1995

- An Introduction to Data Cleaning with R
  by Edwin de Jonge & Mark van der Loo,2013