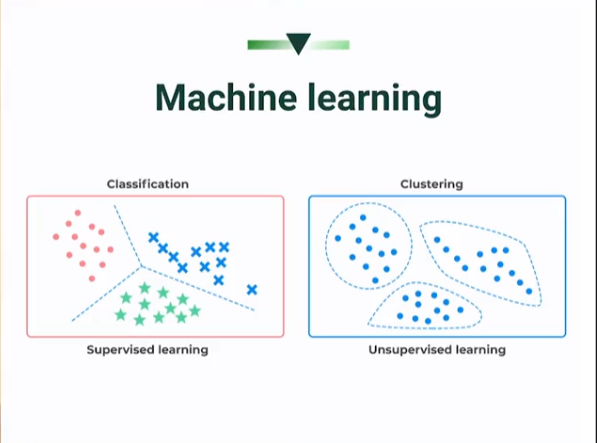
**¿QUÉ ES EL CLUSTERING?**

Diagrama

Descripción generada automáticamente

El Clustering hace parte del aprendizaje no supervizado.

La diferencia entre el aprendizaje supervisado y el no supervisado es que en el NO supervisado, como el caso del Clustering, no hay una variable objetivo, es decir, no hay una clasificación ni un valor continuo. (Precios de casas, línea de tendencia del dólar, si aprobé el examen o no, spam o no, maligno o benigno… etc). El Clustering consiste en encontrar las similitudes y características de un datapoint a otro o clustering.



**¿CUÁNDO USAR CLUSTERING?**

* Mayor contexto de mi dataset, segmentarlo por similitudes para mayor contexto de qué debo analizar de ese cluster específico
* Detección de outliers
* Clasificar/Agrupar (sin variable objetivo)
* Tareas manuales de etiquetas

**¿QUÉ PUEDO LOGRAR?**

* Clasificación del tráfico de una página
* Segmentación de clientes por su perfil, por ejemplo, por la base de datos de consumo. Inclusive, grupo bronce, plata y oro.
* Clasificación de contenido
* Identificar comportamientos fraudulentos con los outliers
* Ciencia en los deportes
* ¡Mucho, mucho más…!



**HAY MUCHOS MODELOS DE CLUSTERING, LOS MÁS IMPORTANTES SON:**

**KMEANS** Su función principal está ene l hecho de que a priori se le dice al modelo cuántos centroides debería buscar (*n\_clusters*) y con ello el algoritmo va a intentar solucionar de la mejor manera lo que se le está pidiendo.

Desventajas:

* Tiene problemas detectando outliers y ruido, esto conlleva a que se clusterice de mala forma.
* Funciona mejor siempre que la data tenga forma globular pero no está pensado para que encuentre datos con formas aleatorias

Estos problemas se resuelven utilizando clustering con base en la densidad, no en centroides. Estos algoritmos se diferencian del kmeans por el hecho de que los separa por zonas altamente densas en puntos de otras de menor densidad. Estos son los DBSCAN (density-based spatial clustring for applications with noise) y su successor jerárquico (Hierarchical) HDBSCAN.

**HIERARCHICAL CLUSTERING**

**DBSCAN Y HDBSCAN**

* **DBSCAN:** Está basado en densidad y típicamente requiere densidad uniforme dentro de un cluster y caídas de densidades entre los clusters. Tiene dos parámetros bases clásicos que necesitan ser elegidos cuidadosamente para optimizarlo:

1. ***eps***: el radio que define el vecino alrededor de cada punto.
2. ***min\_samples*:** el mínimo número de puntos de datos requerido en un vecindario para que un punto de los datos pueda ser considerado el punto central.

* **HDBSCAN:** Clasifica los datos en tres categorías

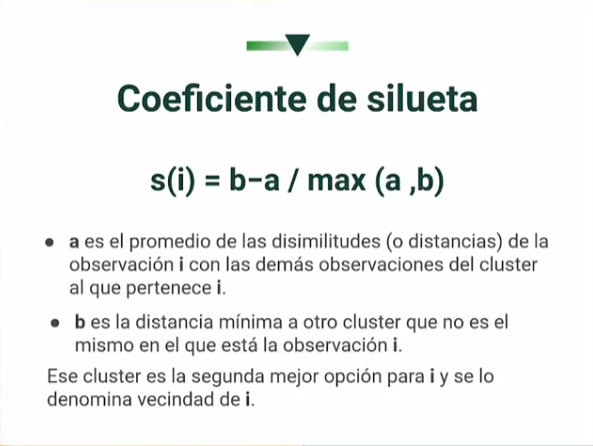
1. Core point: un punto que tiene como mínimo min\_samples puntos dentro de un vecindario con radio eps.
2. Border point: un punto que tiene menos que min\_samples, pero al menos un punto central en su vecindario
3. Noise: un punto que no es ni punto central ni un punto de borde y que tiene menos que min\_samples puntos en su vecindario
4. **PARÁMETRO:**

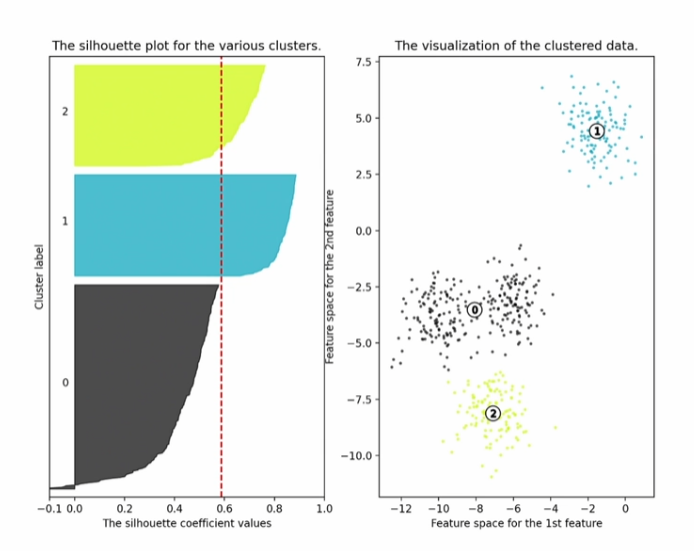
***min\_cluster\_size:*** el mínimo número de puntos para que se forme un cluster

**¿CÓMO EVALUAR EL MODELO DE CLUSTERING?**

No sabemos el número de clusters, ni la variable objetivo. Entonces, ¿cómo se hace?

Usando, por ejemplo, ***el Coeficiente de Silueta***.





El rendimiento del cluster se mide en qué tanto están los clusters parecidos a 1, siendo el mejor valor. Si es negativo, es un mal valor

**K-MEANS**

1. Indicar la cantidad de centroides que se van a buscar dentro de los datos
2. Ubica los centroides aleatoriamente
3. Cada punto se asigna al centroide más cercano
4. Recalcula los centroides con el promedio
5. Repite el paso 3 y 4 hasta que no se muevan más los centroides

**¿Cuándo usar K-Means?**

Ventajas: Alto performance, simple, resultados interpretables, garantiza la convergencia (hasta cuando ya no se mueva más el centroide) y es adaptable a nuevos data points.

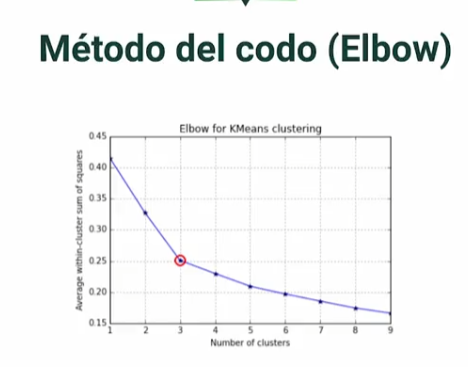
Desventajas: Replicabilidad, es muy difícil replicar los mismos resultados siempre. Le afectan los outliers, mejor performance cuando son esféricos. Es necesario a priori tener una idea de cuántos centroides hay. Es muy afectado por la alta dimensionalidad.

* Conozco la cantidad de clusters
* Resultados rápidos
* Interpretación simple
* Datos de formas esféricas
* Cuando los quiero escalar metiendo mucha más data

**¿CÓMO ENCONTRAR EL VALOR ADECUADO DE K EN EL ALGORITMO DE K-MEANS?**

1. Método del elbow (Del codo): Consiste en raficar la inercia, o la suma de los errores cuadráticos en el K-Means con diferentes K y seleccionar en el que esté el codo, es decir, cuando tiene un mínimo valor en donde se estabiliza, pero con menos clusters, no infinitamente.

WCSS (Within-Cluster Sum of Square): La suma de la diferencia elevada al cuadrado de cada punto contra su centroide en el cluster.

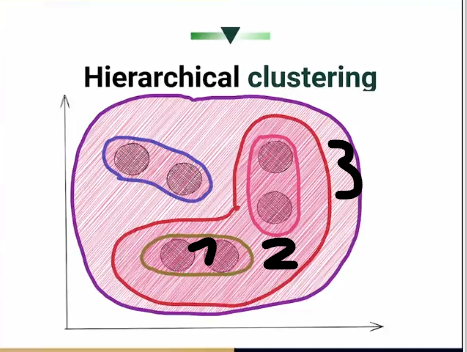


1. Coeficiente de silueta

Donde a es el promedio de disimilitudes entre los data points a donde pertenece i, b es la distancia mínima entre los que no corresponden al cluster donde está i. Maximizar a 1.{

**HIERARCHICAL CLUSTERING**

Va a mirar las distancias entre todos los data points cuando encuentre que hay una distancia menor que las demás va a hacer un cluster. Después revisa las distancias entre los clusters y toma la menor distancia. Lo agrupa otra vez.

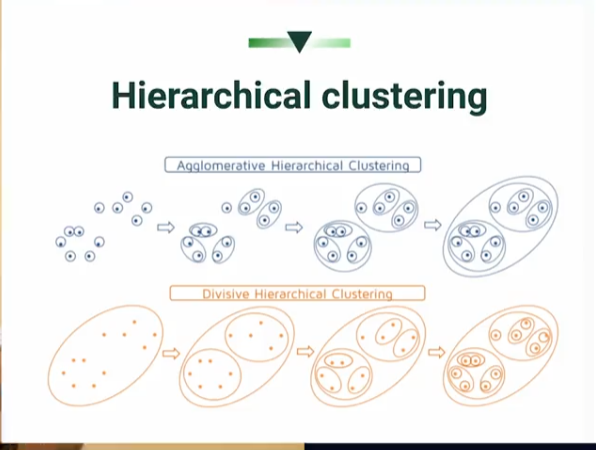


Va a llegar un momento en el que toma un único cluster gigante. Esto se ve en forma de Dendrograma.

Diagrama

Descripción generada automáticamente

Esto, si bien es cierto, es una de las dos posibilidades de clustering jerárquico. Hay una variación en la cual en vez de agrupar se hace una división. Es decir, el proceso inverso. Se empieza desde afuera y se va llegando al límite



**¿CÓMO SE SABE CUÁL ES EL CLUSTER CON EL QUE TIENE QUE ALINEARSE? PORQUE SE DEFINE UNA PROPIEDAD EN EL ALGORITMO QUE ES EL**

**LINKAGE**

**Diagrama

Descripción generada automáticamente**

* **Linkeo simple:** La distancia de los data point más cercanos y esa hace match
* **Linkeo complejo:** La distancia más amplia entre los cluster y esa hace match
* **Linkeo promedio:** La distancia promedio entre todos los data point y esa hace match
* **WARD:** Hace operaciones, evalúa las varianzas y siempre va a hacer match con la menor varianza de todas

**¿CUÁNDO HACER CLUSTERING JERÁRQUICO?**

**Ventajas**

* No necesito el número de clusters
* Simple
* Resultados fácilmente interpretables
* Única ejecución
* Ayuda visual del dendrograma

**Desventajas**

* Tarda en dataset largos
* No tiene un objetivo matemático
* Le afectan outliers drásticamente
* Mayor necesidad de computo

**¿CUÁNDO ENTONCES?**

* **Comprender los resultados de manera visual**
* **Tengo un dataset pequeño**
* **Desconozco la cantidad de clusters por completo**
* **Resultados rápidos**

**DBSCAN Y HDBSCAN**

El DBSCAN, también llamado DENSITY BASED SPATIAL CLUSTERING OF APLICATION WITH NOISE, es un método de clusterización muy importante que se basa en densidades. Su principio es que: para la creación e un cluster se tiene que tener una alta aglomeración de datapoints y puntos, con alta densidad. Se logra haciendo un radio alrededor de cada círculo y éste radio debe contener n número de vecinos cercanos para que tenga suficiente densidad.

Requiere:

Eps: Epsilon del radio

MinPts = Mínimo de puntos para ser un punto core



En ésta imagen se puede ver que KMeans falla al clusterizar ciertos puntos, porque lo que intenta es hacer centroides y desde allí clusterizar, pero no es así. Entonces, es muy útil usar el DBSCAN

**¿CUÁNDO USAR DBSCAN?**

**Ventajas**

* No requiere especificar un número de clusters
* Es capaz de detectar outliers o ruido
* Puede encontrar cluster en distintas formas que no sean esféricas

**Desventajas**

* Los hiper-parámetos son muy determinantes, algunas combinaciones no funcionan igual para todo los grupos con distintas densidades
* Los puntos frontrerizos a los que se puede acceder desde más de un cluster pueden formar parte de cualquier cluster

**¿CUÁNDO USARLO?**

* Desconozco la cantidad de clusters
* No uso formas esféricas
* Densidades similares entre clusters

**PCA Y EL PROBLEMA DE LA DIMENSIONALIDAD**