KIBERNUM
IT Academy

Módulo 6
Clase 3-1

Aprendizaje de Máquina No Supervisado

Análisis de Clustering

Contenido

- Describir los conceptos básicos de Clustering
- Identificar los algoritmos de clústering más utilizados

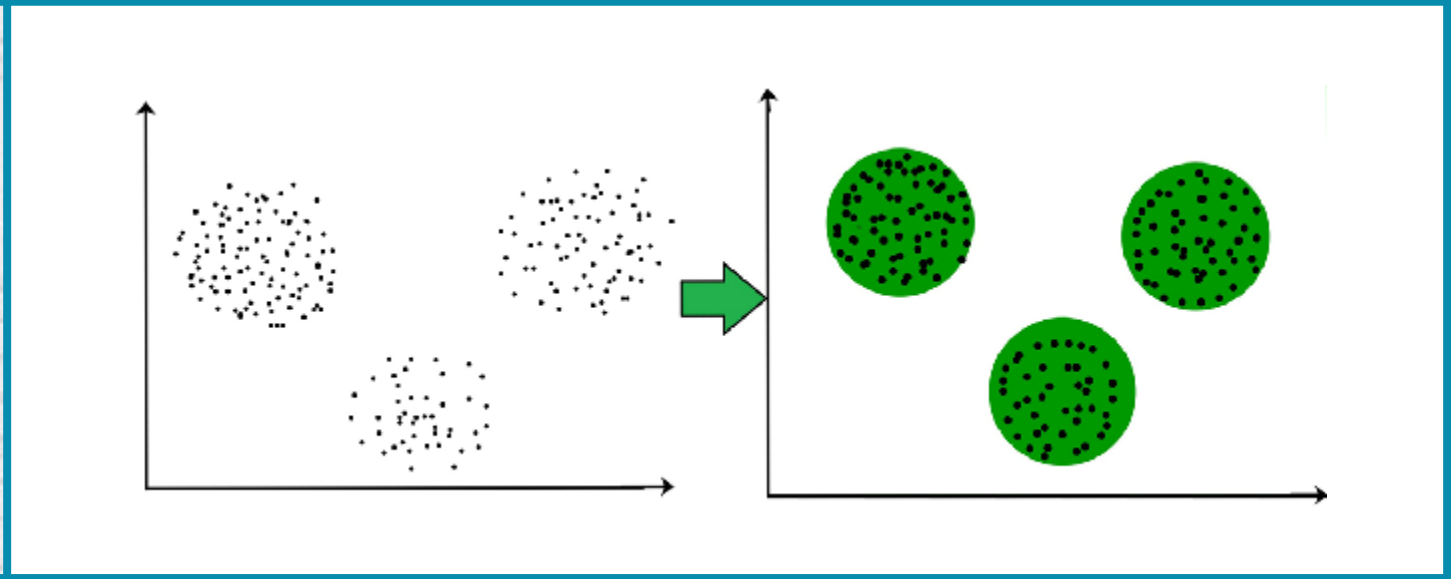
KIBERNUM

Análisis de Clustering

¿Qué es Clustering?

Clustering, también conocido como agrupamiento, es una técnica de aprendizaje no supervisado en la que se utilizan algoritmos para identificar grupos o clústeres de objetos o datos similares. El objetivo principal del clustering es agrupar objetos similares juntos y separar objetos diferentes en grupos distintos, sin tener una clasificación previa de los datos.

El algoritmo de clustering, funciona al analizar las características de los datos y buscando patrones y similitudes en los valores. Estos patrones se utilizan para agrupar los objetos o datos en clusters o grupos.



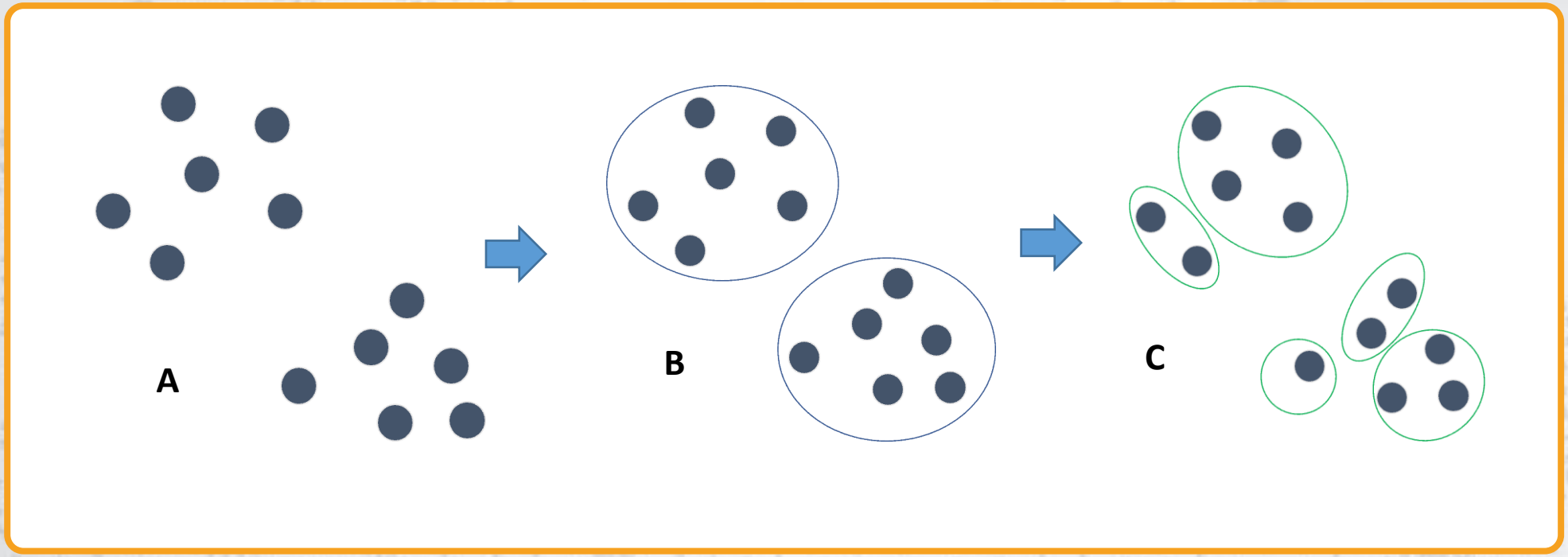
Corresponde a técnicas de Machine Learning no-supervisado en donde a partir de datos no etiquetados, se le asigna una etiqueta.

¿Qué es Clustering?

El clustering, al ser un aprendizaje no supervisado, **no tiene respuestas correctas**. Esto hace que la evaluación de los grupos (cantidad y significado) sea un tanto subjetiva.

¿Qué es Clustering?

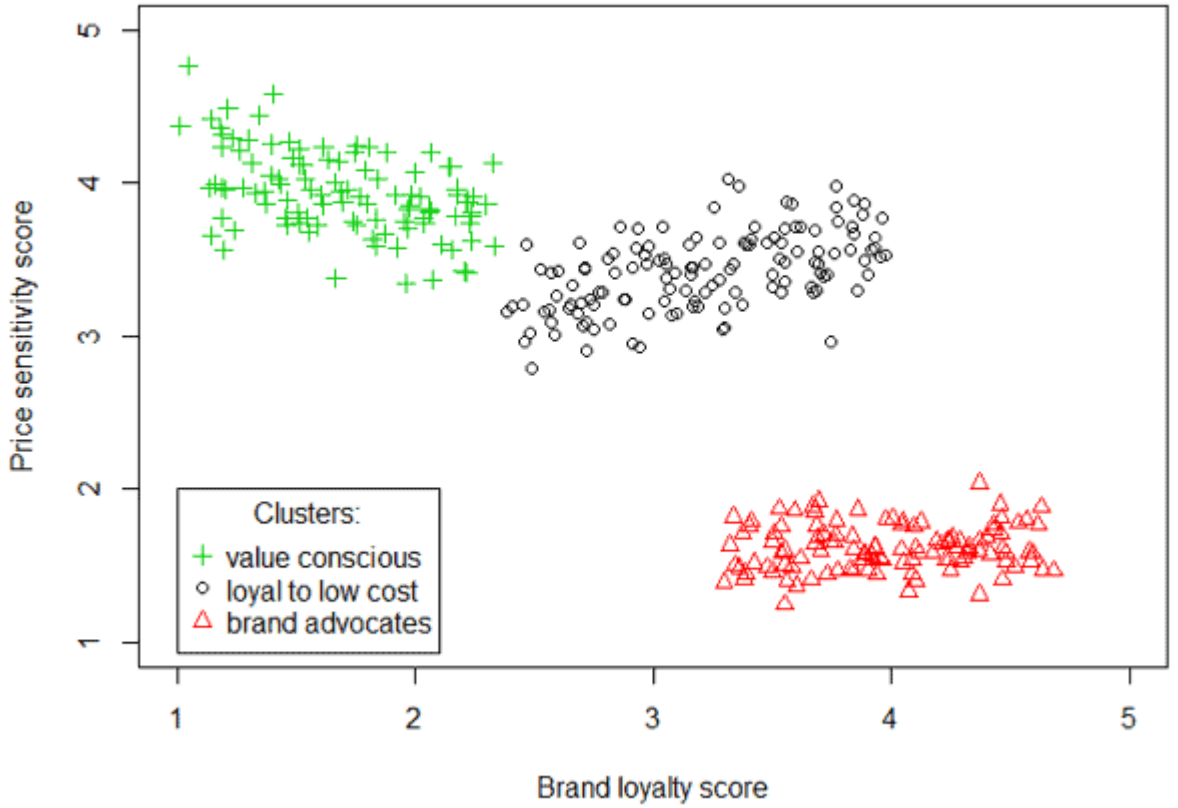
En el siguiente ejemplo, alguna persona podría decir que hay dos clusters, sin embargo, alguien también podría argumentar la existencia de 5 clusters. No hay una respuesta correcta o incorrecta, **va a depender de la interpretación que se haga**.



Ejemplos de Clustering

Segmentación de Clientes

- Preocupados por el precio (verde): no son leales a la marca y son muy sensitivos al precio.
- Leales a precios bajos (negro): son leales a la marca pero sólo si es barato.
- Defensores de la marca (rojo): son leales a la marca sin importar demasiado el precio.



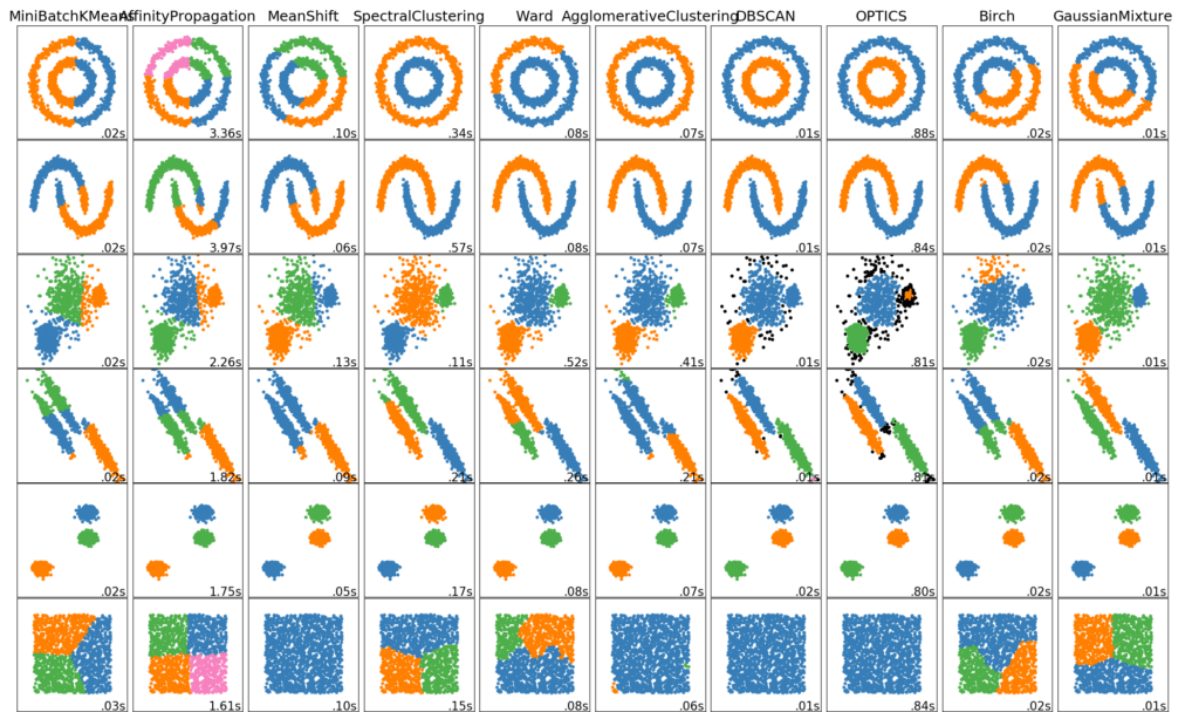
KIBERNUM

Algoritmos de Clustering

Algunos de los algoritmos de clustering más populares son k-means, agrupamiento jerárquico, agrupamiento espectral, agrupamiento basado en densidad, entre otros. Cada uno de estos algoritmos tiene sus propias fortalezas y debilidades, por lo que la elección del algoritmo adecuado dependerá del tipo de datos y de los objetivos del análisis.

Method name	Parameters	Scalability	Usecase	Geometry (metric used)
K-Means	number of clusters	Very large `n_samples`, medium `n_clusters` with MiniBatch code	General-purpose, even cluster size, flat geometry, not too many clusters	Distances between points
Affinity propagation	damping, sample preference	Not scalable with n_samples	Many clusters, uneven cluster size, non-flat geometry	Graph distance (e.g. nearest-neighbor graph)
Mean-shift	bandwidth	Not scalable with `n_samples`	Many clusters, uneven cluster size, non-flat geometry	Distances between points
Spectral clustering	number of clusters	Medium `n_samples`, small `n_clusters`	Few clusters, even cluster size, non-flat geometry	Graph distance (e.g. nearest-neighbor graph)
Ward hierarchical clustering	number of clusters or distance threshold	Large `n_samples` and `n_clusters`	Many clusters, possibly connectivity constraints	Distances between points
Agglomerative clustering	number of clusters or distance threshold, linkage type, distance	Large `n_samples` and `n_clusters`	Many clusters, possibly connectivity constraints, non Euclidean distances	Any pairwise distance
DBSCAN	neighborhood size	Very large `n_samples`, medium `n_clusters`	Non-flat geometry, uneven cluster sizes	Distances between nearest points
OPTICS	minimum cluster membership	Very large `n_samples`, large `n_clusters`	Non-flat geometry, uneven cluster sizes, variable cluster density	Distances between points
Gaussian mixtures	many	Not scalable	Flat geometry, good for density estimation	Mahalanobis distances to centers
Birch	branching factor, threshold, optional global clusterer.	Large `n_clusters` and `n_samples`	Large dataset, outlier removal, data reduction.	Euclidean distance between points

Algoritmos de Clustering



La comparación de los algoritmos de clustering está hecha en función a los parámetros que necesitan, su escalabilidad, caso de uso y geometría (métrica usada). Algunas preguntas útiles pueden ser:

- ¿Tengo una idea del número de grupos que quiero encontrar? … ¿o prefiero que el algoritmo lo encuentre?
- ¿Tengo muchísimos datos? En este caso, deberemos tener en cuenta la escalabilidad del algoritmo.

Aplicaciones del Clustering

Las técnicas de clusterización tienen una amplia variedad de aplicaciones en diferentes campos, entre las que se incluyen:

1. **Marketing y publicidad**: La clusterización se utiliza para segmentar a los consumidores en diferentes grupos basados en sus preferencias, hábitos de compra, comportamientos y otras variables. Esto ayuda a las empresas a personalizar su publicidad y marketing para llegar a los consumidores de manera más efectiva.

2. **Biología**: En biología, la clusterización se utiliza para agrupar células en diferentes tipos o grupos basados en su expresión genética. Esto ayuda a los científicos a entender mejor la biología celular y a identificar posibles tratamientos para enfermedades.

Aplicaciones del Clustering

1. **Análisis de redes sociales**: La clusterización se utiliza para agrupar a los usuarios de las redes sociales en diferentes comunidades o grupos basados en sus intereses y relaciones sociales. Esto ayuda a las empresas a entender mejor a su audiencia y a diseñar estrategias de marketing más efectivas.

2. **Finanzas**: La clusterización se utiliza para agrupar diferentes activos financieros en diferentes clases de activos, como acciones, bonos y materias primas. Esto ayuda a los inversores a diversificar sus carteras y a minimizar el riesgo.

3. **Imagen y visión por computadora**: La clusterización se utiliza para segmentar y clasificar imágenes en diferentes grupos basados en su contenido, como el color, la textura y la forma. Esto ayuda a las máquinas a entender mejor las imágenes y a realizar tareas como la detección de objetos y la clasificación de imágenes.

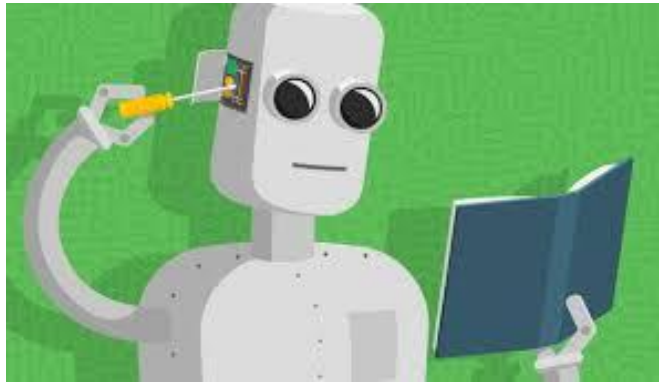
Tarea de Clustering

Ejemplo: “Segmentar clientes en subconjuntos símiles”

Experiencia
Datos de variables sociodemográficas, comportamiento de compra, comportamiento de pago.

Tarea
Generar grupos homogéneos de clientes.

Performance
Similitud de los clientes en cada grupo.



“Se dice que un computador aprende de la *experiencia E*, con respecto a una *tarea T* y una medida de *performance P*, si su performance en T, medido por P, mejora con la experiencia E.”

(Tom Mitchell, 1998)

KIBERNUM

Dudas y consultas

Gracias