TALENTO TECH

TIC

Lección 3: Clustering

Universidad Tecnológica de Pereira

faceIT

Tiempo de ejecución: 2 horas

PLANTEAMIENTO DE LA SESIÓN	MATERIALES
El clustering, o agrupamiento, es una técnica de aprendizaje no supervisado utilizada en análisis de datos y minería de datos para identificar patrones y estructuras subyacentes en un conjunto de datos, agrupando objetos similares en clusters o grupos. Esta técnica es útil para la exploración de datos, segmentación de mercado, comprensión de la estructura de datos y reducción de la dimensionalidad, entre otras aplicaciones, permitiendo descubrir información útil y reveladora sin la necesidad de etiquetas previamente definidas.	Dataset iris: https://archive.ics.uci.edu/dataset/53/iris

Un clúster es un conjunto de elementos que son similares entre ellos y distintos de elementos de otros clústeres. En la figura 1 se muestra un ejemplo en donde existen datos que pertenecen a tres categorías diferentes, dependiendo de sus coordenadas (encerrados en círculos de color verde). En la figura se puede apreciar que cada clúster tiene elementos lo suficientemente similares para ser considerados parte de un mismo grupo, y que entre los datos de diferentes clústeres hay diferencias que permiten separar los clústeres. En el ejemplo los clústeres son claramente separables por una curva. Sin embargo, dependiendo de los datos con los que se trabaje, en datos reales tal condición no es claramente apreciable.

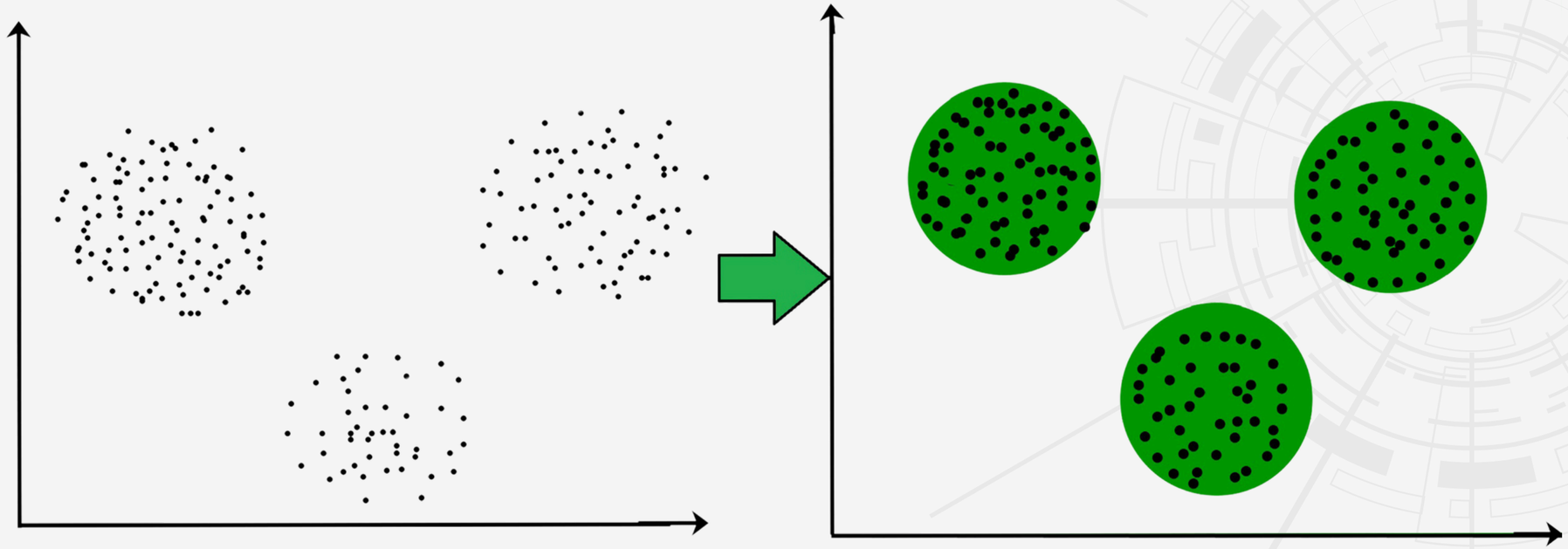

Figura 1: Clústeres.

Los clústeres no necesariamente deben tener forma esférica como se ve en la figura 2:

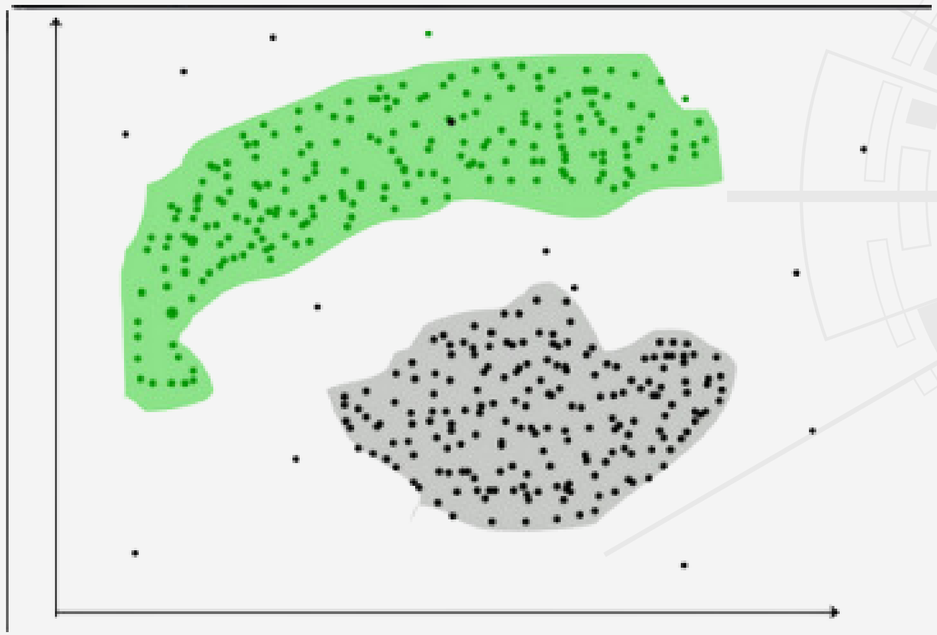

Figura 2: Clústeres no esféricos

La importancia del crear clústeres (clustering) es que permite detectar los grupos intrínsecos que pueden existir entre datos no etiquetados. Estos algoritmos permiten resolver tareas como:
Etiquetar datos, saber cuántos grupos diferentes existen en un conjunto de datos, reducir las dimensiones de los datos y saber cuales datos son los más representativos de un grupo.

Existen varios métodos que permiten agrupar datos como son:

Métodos basados en densidad: Estos métodos consideran que los clústeres son regiones densas que poseen similaridades y diferencias de regiones menos densas en el espacio. Estos métodos tienen buena precisión y la habilidad de unir clústeres. Por ejemplo el algoritmo:

DBSCAN (Density based spatial clustering of applications with noise)
OPTICS (ordering points to identify clustering structure), entre otros.

Métodos basados en jerarquías: Estos métodos agrupan datos en forma de una estructura de datos de tipo árbol, la cual cuenta con una jerarquía de nodos. Los clústeres nuevos están formados empleando clústeres formados previamente. Existen jerarquías aglomerativas
(bottom-up) y divisivas (top-down).
Por ejemplo, el algoritmo CURE y BIRCH.

Métodos de particionado: Estos métodos particionan los objetos en k clústeres. Este método optimiza un criterio que permite establecer si un dato pertenece o no a determinado clúster, el criterio es generalmente una medida de distancia, como es el caso de los algoritmos de k-means y CLARANS.

Métodos basados en rejillas: En estos métodos, los espacios de datos se separan en rejillas y se miden las distancias de los objetos a cada una de las rejillas. La ventaja de estos métodos es que tienen un número predefinido de clusters desde el inicio, con lo que las operaciones son rápidas. Como por ejemplo los algoritmos STING, wave cluster y CLIQUE.

Como ejemplo veremos el funcionamiento del algoritmo k-means que es uno de los más populares, tanto en ciencia de datos como en minería de datos.

En el algoritmo de K-Means lo que se quiere es agrupar un conjunto de datos en k grupos o clústeres de tal forma que los datos cercanos a cada clúster sean lo más cercanos posibles (similares, con una medida de distancia pequeña) entre sí.

Para iniciar se asignan aleatoriamente N/k puntos a cada uno de los k clústeres.

Cada uno de los k grupos definidos estará representado por un punto en el espacio llamado centroide. Este punto es la media aritmética de los puntos aleatoriamente asignados al grupo. Posteriormente se calculan las distancias de los datos a los clústeres y se reagrupan los datos, es decir, en lugar de ser una selección aleatoria, solo se asignarán a un cluster los N/k datos más cercanos al cluster. Se repite el cálculo del centroide (la media entre los datos que pertenecen al cluster) y se recalculan las distancias.

El proceso de recalcular el nuevo centroide y reasignar los datos a cada clúster se repite hasta que se cumpla alguna de dos condiciones. La primera, que se alcance el número máximo de iteraciones establecido o la segunda, hasta que los clústeres ya no cambien, es decir, que los centroides permanezcan constantes al repetir el proceso.

Este algoritmo tiene una dificultad y es ajustar el valor k que representa la cantidad de clústeres a originar. Una forma de resolverlo es mediante el método del codo, que consiste en encontrar un valor óptimo para k, graficando los valores de k junto con la suma de errores cuadráticos (distancia euclidiana del dato al centroide del clúster) para cada valor de k. A medida que se aumenta k en este proceso, la suma de los errores cuadráticos disminuye lentamente. Existe un punto en el que la suma de error cuadrático cambia drásticamente de pendiente, formando una especie de codo. En ese punto se considera el óptimo valor de k.

Ejercicio en clase opcional: Implementar en Python el algoritmo de k-means con un criterio de parada únicamente dado por el número de iteraciones.

Ejercicio en clase: Utilizando la librería scikit-learn, realizar clústering para cada uno de las variables del dataset iris. Revisar si los clústeres coinciden con las especies de las flores. (link del dataset iris: https://archive.ics.uci.edu/dataset/53/iris)