Clustering: Mean Shift, DBSCAN y Agglomerative Hierarchical Clustering

Martín Gutiérrez

August 7, 2022

Otros métodos

En esta sesión, la idea es ver tres otros métodos de clustering. Estos métodos tienen un enfoque similar a los que vimos durante la sesión anterior: todos trabajan con un conjunto de puntos de datos e intentan agruparlos por características.

Mean Shift - Intro

De partida, Mean Shift lleva un nombre algo extraño, porque el algoritmo se basa en el uso de la moda, más que del promedio. La moda, a forma de recordatorio, es aquella región o valor que contiene la mayor densidad de puntos.

No obstante, y al igual que K-Means, se aplica una asociación de cada punto de datos a un centroide (el más cercano). Ahora bien, a diferencia de K-Means, no hay que especificar K, sino que el algoritmo mismo determinará cuántos clusters. Para ello, se itera sobre todos los datos.

Elementos necesarios para el algoritmo:

- Función para cálculo de vecinos (generalmente, distancia euclidiana dentro de un radio): $N(x)$
- Ventana probabilística o Kernel (generalmente, la gaussiana es la más empleada): $K(x, x')$

Mean Shift - La idea

La idea principal detrás del algoritmo es migrar los centroides hacia las regiones de mayor densidad de datos. Esto se hace calculando el movimiento de cada uno de los puntos hacia un centroide del cluster. Estos son generados directamente a partir de los puntos.

Una forma de verlo intuitivamente es que hay centros de gravedad (modas) dados por los datos mismos, y esos centros de gravedad "tiran" a los puntos hacia ellos. Eso se calcula conociendo quiénes son los datos que están cerca mío (vecinos) y cómo alteran mi movimiento.

Mean Shift - El algoritmo

Se itera N veces sobre cada punto x o hasta que converja:

1. Encontrar el conjunto de vecinos $N(x)$.
2. Calcular $m(x)$, el movimiento a efectuar de acuerdo a la siguiente expresión:

$$m(x) = \frac{\sum_{x_i \in N(x)} K(x,x_i) x_i}{\sum_{x_i \in N(x)} K(x,x_i)}$$

3. $x \leftarrow m(x)$

La función $K(x, x')$, para el caso de un Kernel Gaussiano, se define como:

$$K(x, x') = e^{-\frac{\|x-x'\|^2}{2\sigma^2}}$$

siendo σ es la desviación estándar de los datos.

DBSCAN - Intro

Density Based Spatial Clustering of Applications with Noise es otro algoritmo más de clustering que además de agrupar al igual que los otros, identifica outliers en el grupo de datos. Trabaja con medidas de densidad.

El algoritmo usa dos parámetros:

- La mínima distancia entre dos puntos (antes de considerarlos vecinos) (d)
- El mínimo número de puntos para una región densa (n)

Por supuesto, la elección de los parámetros es un problema...

DBSCAN - La idea

La idea detrás de DBSCAN es que se trabaja un cluster como un conjunto de datos fuertemente denso. El algoritmo itera sobre todos los puntos de los datos. Esto es como una "infección" producida por la regla de pertenecer al cluster que se podría enunciar como:

"Si hay más de n puntos dentro de un radio de d alrededor mío, entonces, es una región densa."

En caso de cumplirse esta condición, se revisan el resto de los puntos pertenecientes a la nueva región identificada para integrar a otros puntos nuevos (que cumplan la misma condición) al cluster.

Quedarán puntos fuera de todos los clusters encontrados, que se identificarán como outliers.

DBSCAN - El algoritmo

Concretamente, el algoritmo se describe de la siguiente forma:

Iterar sobre todos los puntos de datos no visitados:

1. Encontrar todos los puntos que estén a distancia menor de d del punto actual y marcarlo como visitado
2. Si la cantidad de esos puntos (incluyendo al que consulta) es mayor o igual a n
   - Se marcan todos los puntos encontrados como un nuevo cluster.
   - Se repiten los pasos 1 y 2 con cada uno de los puntos en el nuevo cluster, con la excepción de que se marcan inmediatamente todos los puntos encontrados ahora como del mismo cluster.

DBSCAN - Para que jueguen

Encontré un website que tiene un ejemplo interactivo de DBSCAN bastante bueno:

`https://www.naftaliharris.com/blog/visualizing-dbscan-clustering/`

Agglomerative Hierarchical Clustering - Intro

Este algoritmo enfoca el problema de manera fundamentalmente inversa a lo que estamos acostumbrados: construye los clusters por medio de fusiones entre clusters más pequeños. Esto es, ocupa un enfoque bottom-up.

El resultado son agrupaciones representadas en forma de árbol. Estas se denominan dendrogramas. Los clusters más grandes incluyen de forma anidada a los clusters más pequeños.

Agglomerative Hierarchical Clustering - La idea

De manera lógica, es bastante simple: agrupar a aquellos clusters más cercanos y juntarlos gradualmente hasta llegar a un solo cluster. Esto se hace usando distancias entre clusters.

Es de notar que la agrupación ocurre de a pares en cada nivel. Además, los resultados van quedan en una matriz que registra cada uno de los niveles de la ejecución.

Agglomerative Hierarchical Clustering - El algoritmo

1. Asignar un cluster a cada punto
2. Iterar mientras haya más de un cluster:
    1. Evaluar todas las distancias entre clusters de a pares
    2. Registrar las distancias entre clusters en una matriz
    3. Buscar la menor distancia entre clusters, fusionarlos como un nuevo cluster único y eliminar ese par de la matriz

Y con esto, terminamos clustering

El siguiente tema son clasificadores y predictores de Machine Learning.