Clustering espectral

Ximena Gutierrez-Vasques

Víctor Mijangos

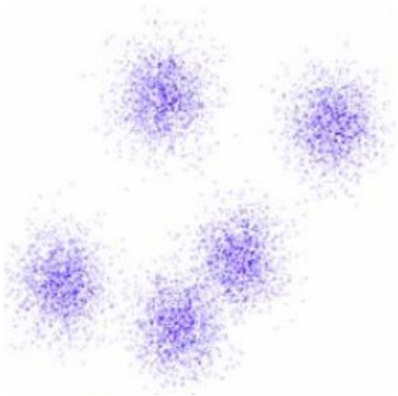
Comunidad Elotl

RIIAA MEX

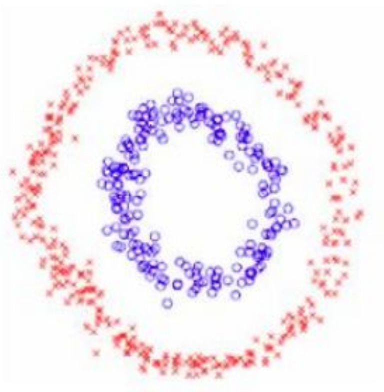
Clustering espectral

- Algoritmos para **agrupar** instancias, usando **eigenvectores** de matrices que se construyen a partir de los datos
- Las instancias se representan utilizando una **topología de grafos**
- Empíricamente muestra un buen desempeño

Clustering espectral



Compacticidad



Conectividad

Diferentes criterios para clustering

Clustering espectral

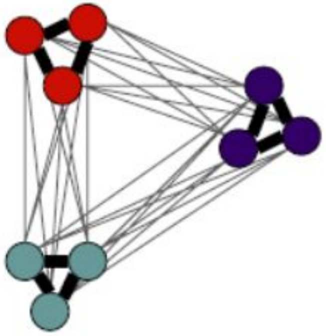
- Dado un conjunto de instancias de entrada y alguna **medida de similitud** que las asocie, representamos a estos datos usando un **grafo de similitud:**

$$G = (\underbrace{V}_{\text{instancias}}, \underbrace{E}_{\text{similitud}})$$

- La idea es que dos vértices están conectados, si la similitud entre esas dos instancias es positiva (o mayor que un cierto valor). El arco tiene como peso esta similitud

Clustering espectral

- El problema se reformula usando el grafo de similitud:
- Queremos encontrar una **partición del grafo,** de tal manera que los arcos entre diferentes grupos tengan un **peso muy bajo** (puntos de diferentes clusters son disimilares) y los arcos entre puntos de un mismo cluster tengan un **peso alto** (similares entre ellos)



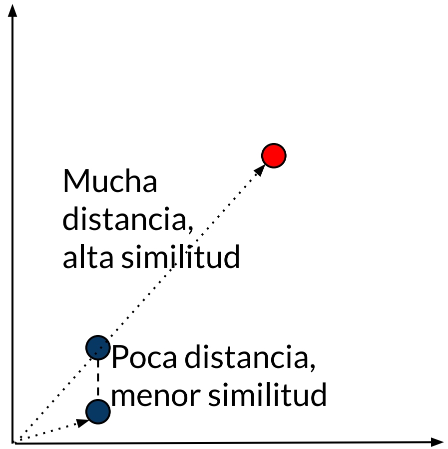
Clustering espectral

Diferentes formas de crear el grafo de similitud:

k-nearest neighbor graph: (conectamos dos nodos con un arco no dirigido, si esos nodos están dentro los primeros k vecinos)

fully connected graph: (conectamos todos los puntos con similitud positiva y les asignamos un peso a los arcos dependiendo de la similitud)

Clustering espectral



- Las medidas de distancia (espacio vectorial) y similitud (pesos del grafo) son distintas:
- Mientras dos vectores cercanos tienden a tener una distancia 0; esto implica que la similitud entre ellos debe aumentar.
- Para pasar de una distancia entre vectores a una similitud entre nodos de un grafo, se utiliza una función llamada **graph kernel.**

Clustering espectral

- Los graph kernels se basan en una métrica; generalmente, la **distancia euclidiana**.
- Se pueden utilizar diferentes funciones que toman una distancia y la llevan a una similitud. Algunas son:

$$K_{Eu}(u,v) = \frac{\sigma^2}{||u-v||^2}$$

$$K_{EuA}(u,v) = (1 + \frac{||u-v||^2}{\sigma^2})^{-1}$$

$$K_{Ga}(u,v) = e^{-\frac{||u-v||^2}{2\sigma^2}}$$

a) Euclidiano inverso

b) Euclidiano ajustado

c) Gaussiano

Clustering espectral

PASOS

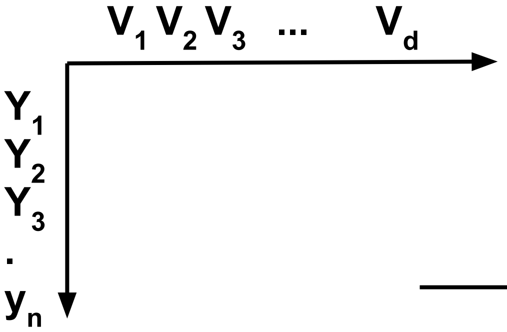
1. Construimos un grafo de similitud G = (V, E), elegimos un valor de *k* clusters
2. Expresamos a este grafo mediante una matriz de adyacencia o de pesos $W \in R^{(nxn)}$
3. Calculamos también una matriz *D* de grado del grafo $D \in R^{(nxn)}$
4. Calculamos el Laplaciano $L=D-W$

Clustering espectral

5. Calculamos los eigen valores de la matriz *L*, nos quedamos con los *d* valores más pequeños.

6. Calculamos el eigenvector asociado a cada eigen valor

7. Construimos una matriz con estos eigenvectores

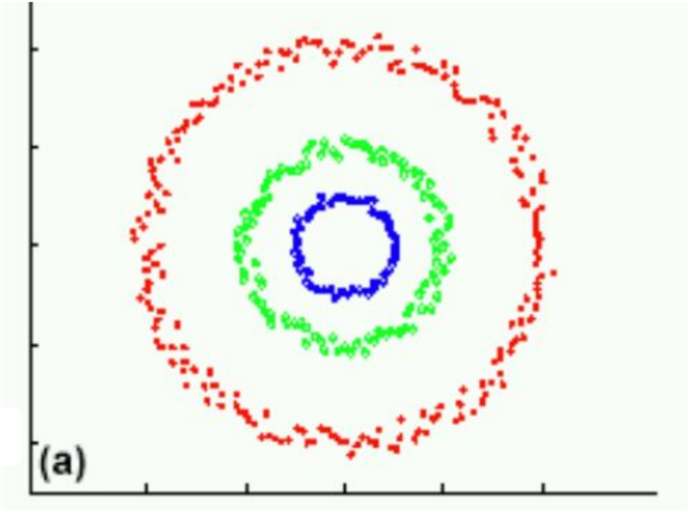


Matrix V: tamaño *n* x *d*

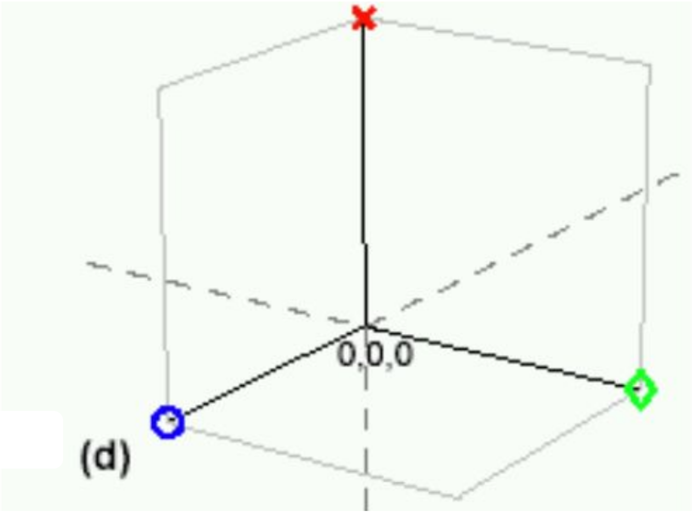
Se pasa a un espacio *d*-dimensional (idealmente convexo)

Clustering espectral

8. Finalmente, aplicamos clustering de k-Medias a los nuevos vectores $y_1, y_2, y_3, \ldots, y_n$. Para obtener K clusters



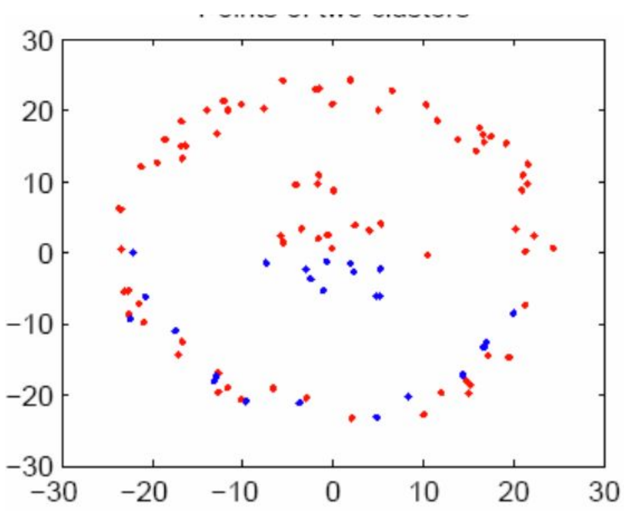
Datos originales



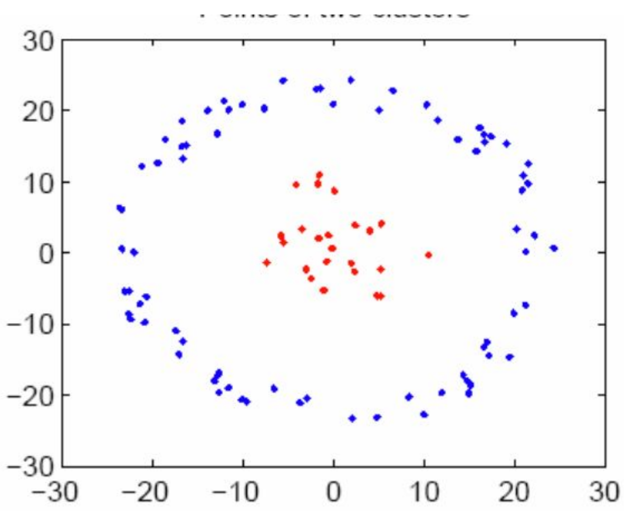
Datos proyectados

Clustering espectral

Cuando aplicamos k-medias a los eigenvectores de la matriz laplaciana, podemos encontrar clusters con fronteras **no convexas**:



k-means output



Spectral clustering output

Referencias

- https://www.cs.cmu.edu/~aarti/Class/10701/readings/Luxburg06_TR.pdf
- https://www.cs.cmu.edu/~aarti/Class/10701/slides/Lecture21_2.pdf
- https://www.researchgate.net/publication/224375502_Alternative_Similarity_Functions_for_Graph_Kernels

Práctica