Aprendizaje automatizado

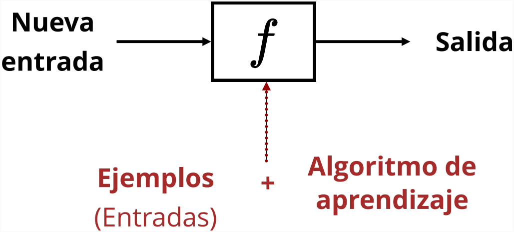
Agrupamiento

Gibran Fuentes Pineda

Abril 2023

Aprendizaje sin supervisión

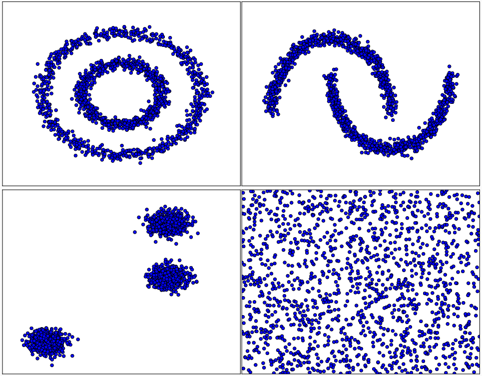
- Ejemplos sólo contienen entradas sin salidas deseadas
- Algunas tareas: el agrupamiento y el descubrimiento de patrones



- Busca encontrar la estructura escondida de los datos sin necesitar etiquetas

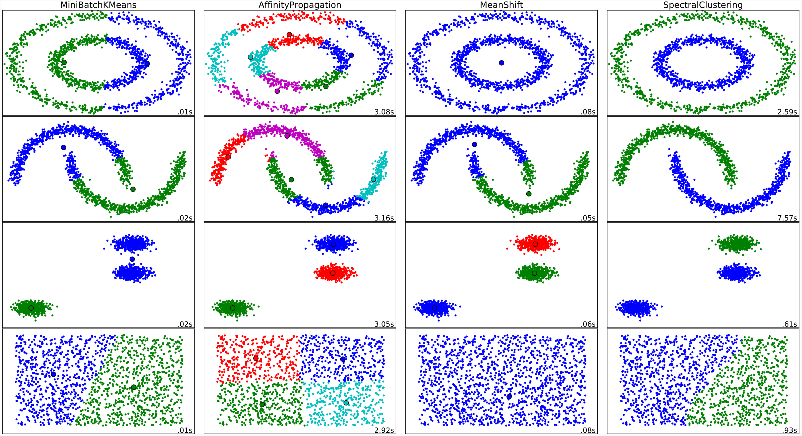
Agrupamiento

- Objetivo: agrupar ejemplos en base a su proximidad
- Criterios: por conectividad y por compacidad



Ejemplo de `http://scikit-learn.org`

Diferentes algoritmos de agrupamiento



Ejemplo de `http://scikit-learn.org`

Agrupamiento por K-medias

- Divide ejemplos en K grupos, asignando cada ejemplo al grupo con el centroide más cercano

- Busca los K centroides que minimicen

$$E[\boldsymbol{\mu}_1, \dots, \boldsymbol{\mu}_K] = \sum_{i=1}^{n} \sum_{k=1}^{K} r_{ik} \|\mathbf{x}_i - \boldsymbol{\mu}_k\|^2$$

  donde $r_{ik} = 1$ si $\boldsymbol{\mu}_k$ es el centroide más cercano a $\mathbf{x}_i$ y $r_{ik} = 0$ en caso contrario

Algoritmo de K-medias

1. Elige K ejemplos aleatoriamente como centroides iniciales

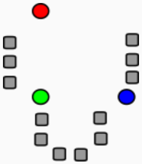


Imagen tomada de Wikipedia (K-means clustering)

1. Elige K ejemplos aleatoriamente como centroides iniciales
2. Asigna cada ejemplo al centroide más próximo

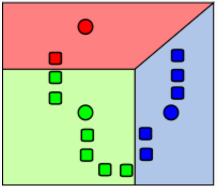


Imagen tomada de Wikipedia (K-means clustering)

Algoritmo de K-medias

1. Elige K ejemplos aleatoriamente como centroides iniciales
2. Asigna cada ejemplo al centroide más próximo
3. Re-calcula los centroides a partir de las asignaciones

$$\boldsymbol{\mu}_k = \frac{1}{n_k}\sum_{i=1}^{n} r_{ik}\mathbf{x}_i, n_k = \sum_{i=1}^{n} r_{ik}, k = 1, \ldots, K$$

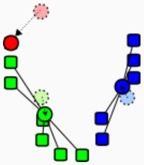


Imagen tomada de Wikipedia (K-means clustering)

Algoritmo de K-Means

1. Elige K ejemplos aleatoriamente como centroides iniciales
2. Asigna cada ejemplo al centroide más próximo
3. Re-calcula los centroides a partir de las asignaciones
4. Repite hasta cumplir criterio de convergencia (por ej. que E no disminuya)

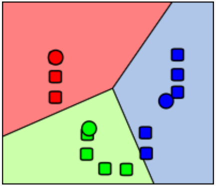


Imagen tomada de Wikipedia (K-means clustering)

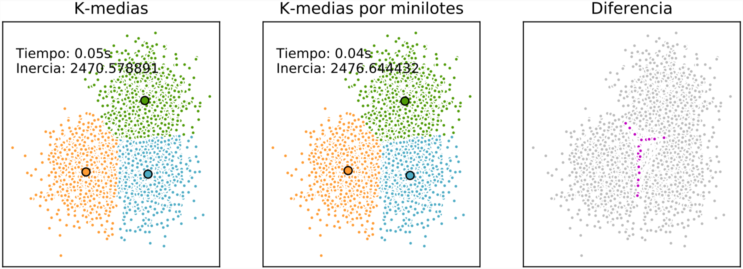
Agrupamiento de imágenes de dígitos con K-medias

- El algoritmo de K-medias genera una partición del espacio representado por el diagrama de Voronoi en el que cada punto está asociado al centroide más próximo.

Ejemplo de `http://scikit-learn.org`

K-medias por minilotes[1] *vs* K-medias

- Actualiza centroides y asignaciones usando un ejemplo o un subconjunto pequeño de ejemplos a la vez



Ejemplo de `http://scikit-learn.org`

[1] D. Sculley. *Web-Scale K-Means Clustering*, 2010.

Agrupamiento jerárquico

- Construye de forma gradual una jerarquía de grupos siguiendo un criterio dado
    - **Aglomerativo**: empieza considerando cada dato como un grupo y va mezclando grupos
    - **Divisivo**: empieza considerando todos los datos como un solo grupo y lo va dividiendo
- Dados dos grupos $\{\mathcal{G}_a, \mathcal{G}_b\}$, algunos criterios son
    - Mínimo o simple: $\min \{dist(\mathrm{x}^{(i)}, \mathrm{x}^{(j)}) : \mathrm{x}^{(i)} \in \mathcal{G}_a, \mathrm{x}^{(j)} \in \mathcal{G}_b\}$
    - Completo o máximo: $\max \{dist(\mathrm{x}^{(i)}, \mathrm{x}^{(j)}) : \mathrm{x}^{(i)} \in \mathcal{G}_a, \mathrm{x}^{(j)} \in \mathcal{G}_b\}$
    - Promedio
$$\frac{1}{\mid \mathcal{G}_a \mid\mid \mathcal{G}_b \mid} \sum_{\mathrm{x}^{(i)} \in \mathcal{G}_a} \sum_{\mathrm{x}^{(j)} \in \mathcal{G}_b} dist(\mathrm{x}^{(i)}, \mathrm{x}^{(j)})$$
    - Mínima varianza: elige el par de grupos en el que la varianza intergrupal se incremente menos

Dendogramas

- Diagrama jerárquico que muestra los agrupamientos en distintos niveles

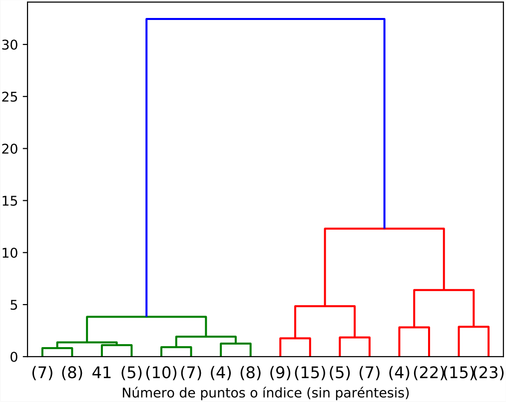


Imagen generada por ejemplo de scikit-learn

Agrupamiento espectral

- Se calcula la matriz laplaciana L a partir de la matriz de adyacencia o afinidad A de la siguiente manera
  - **Sin normalizar**
  $$\mathsf{L} = \mathsf{D} - \mathsf{A}$$
  - **Normalizada simétrica**
  $$\mathsf{L}_{\text{sim}} = \mathsf{D}^{-1/2}\mathsf{L}\mathsf{D}^{-1/2} = \mathsf{I} - \mathsf{D}^{-1/2}\mathsf{A}\mathsf{D}^{-1/2}$$
  - **Caminata aleatoria**
  $$\mathsf{L}_{\text{ca}} = \mathsf{D}^{-1}\mathsf{L} = \mathsf{I} - \mathsf{D}^{-1}\mathsf{A}$$
- Se realiza el agrupamiento usando K-medias sobre los puntos representados por los K eigenvectores con mayores eigenvalores de la matriz laplaciana L

Corte de grafos

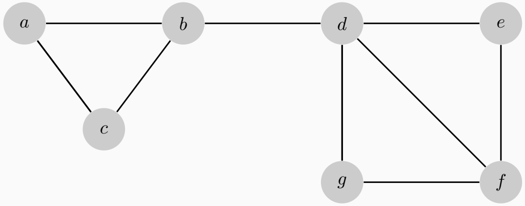
- Consideremos una red como grafo no dirigido $\mathcal{G}(\mathcal{V}, \mathcal{E})$

$$Corte(\mathsf{A}) = \sum_{i \in \mathsf{A}, j \notin A} w_{i,j}$$

donde A es un conjunto de vértices del grafo y $w_{i,j}$ denota el peso de la arista entre los vértices i y j.

Partición del grafo

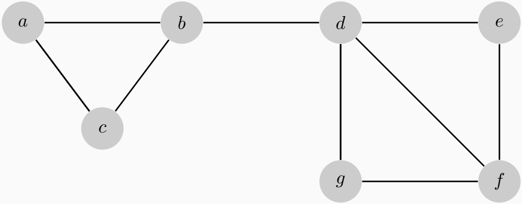
- Objetivo: dividir los vértices en dos grupos disjuntos tal que
  - Se maximice el número de conexiones entre vértices del mismo grupo
  - Se minimice el número de conexiones entre vértices de grupos distintos



Representación de grafos: matriz de adyacencia

- Matriz de adyacencia A

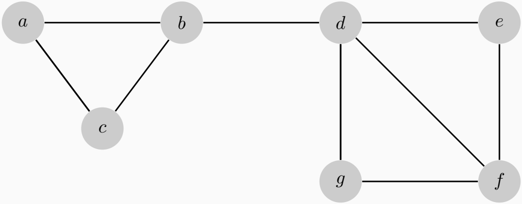
$$\mathsf{A} = \begin{bmatrix} 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 \end{bmatrix}$$



Representación de grafos: matriz de grado

- Matriz de grado G

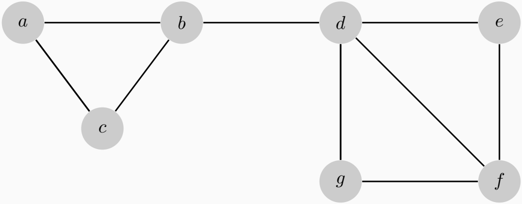
$$\mathsf{G} = \begin{bmatrix} 2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 3 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 4 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 3 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 2 \end{bmatrix}$$



Representación de grafos: matriz laplaciana

- Laplaciana $\mathsf{L} = \mathsf{G} - \mathsf{A}$

$$\mathsf{L} = \begin{bmatrix} 2 & -1 & -1 & 0 & 0 & 0 & 0 \\ -1 & 3 & -1 & -1 & 0 & 0 & 0 \\ -1 & -1 & 2 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 4 & -1 & -1 & -1 \\ 0 & 0 & 0 & -1 & 2 & -1 & 0 \\ 0 & 0 & 0 & -1 & -1 & 3 & -1 \\ 0 & 0 & 0 & -1 & 0 & -1 & 2 \end{bmatrix}$$



Eigendescomposición de grafos

- La idea es analizar el *spectrum* de la matriz laplaciana del grafo
    - Conjunto de eigenvectores $\{\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(K)}\}$ del grafo con sus correspondientes eigenvalores asociados $\Lambda = \{\lambda_1, \ldots, \lambda_K\}$, donde $\lambda_1 \leq \lambda_2, \cdots, \leq \lambda_K$

- Dada la matriz laplaciana $\mathbf{L}$ de un grafo $\mathcal{G}$, su primer eigenvalor λ_1 y correspondiente eigenvector son
    - $\lambda_1 = 0$ y $\mathbf{x} = [1, 1, \ldots, 1]$, por lo que $\mathbf{L} \cdot \mathbf{x} = 0$

Encontrando λ_2 (1)

- Para cualquier matriz de simétrica $\mathbf{M}$

$$\lambda_2 = \min_{\mathbf{x}} \frac{\mathbf{x}^\top \mathbf{M}\mathbf{x}}{\mathbf{x}^\top \mathbf{x}}$$

- Para una matriz laplaciana $\mathbf{L}$

$$\begin{aligned}
\mathbf{x}^\top \mathbf{L}\mathbf{x} &= \sum_{i=1,j=1}^{n} L_{i,j} \cdot x_i \cdot x_j = \sum_{i=1,j=1}^{n} (D_{i,j} - A_{i,j}) \cdot x_i \cdot x_j \\
&= \sum_{i==j} D_{i,j} \cdot x_i^2 - \sum_{(i,j)\in \mathbf{E}} 2 \cdot x_i \cdot x_j = \sum_{(i,j)\in \mathbf{E}} \left[x_i^2 + x_j^2 - 2 \cdot x_i \cdot x_j\right] \\
&= \sum_{(i,j)\in \mathbf{E}} \left[x_i - x_j\right]^2
\end{aligned}$$

Encontrando λ_2 (2)

- Restricciones
  - $\mathbf{x}$ es un vector unitario, lo que implica que

$$\sum_{i=1}^{n} x_i^2 = 1$$

  - $\mathbf{x}$ es ortogonal a $[1, 1, \ldots, 1]$ (primer eigenvector); en consecuencia

$$\sum_{i=1}^{n} x_i \cdot 1 = \sum_{i=1}^{n} x_i = 0$$

- Por lo tanto, el problema de minimización busca valores x_i con pocas aristas que crucen por cero

$$\lambda_2 = \min_{x_i} \frac{\sum_{(i,j)\in \mathbf{E}} \left[x_i - x_j\right]^2}{\sum_{i=1}^{n} x_i^2}$$

Corte óptimo

- Problema: cortar el grafo $\mathcal{G}$ en dos subgrafos $\mathcal{A}$ y $\mathcal{B}$
    - Se representa por un vector $\mathbf{x}$, donde un elemento tiene un valor 1 si pertenece a $\mathcal{A}$ y -1 si pertenece a $\mathcal{B}$
- Se busca un vector no trivial

$$\mathbf{x} = \underset{\mathbf{x} \in \{-1,+1\}^n}{\arg\min} \sum_{(i,j) \in \mathbf{E}} \left[x_i - x_j\right]^2$$

- No es posible resolver de forma exacta, por lo que se relaja el problema a

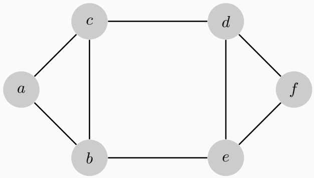
$$\mathbf{x} = \underset{\mathbf{x} \in \mathbb{R}^n}{\arg\min} \overbrace{\sum_{(i,j) \in \mathbf{E}} \left[x_i - x_j\right]^2}^{f(\mathbf{x})}$$

- La solución está dada por el eigenvector asociado al segundo eigenvalor de la matriz laplaciana $\mathbf{L}$

Partición espectral

1. Construir la matriz laplaciana $\mathbf{L}$.
2. Calcular los eigenvalores y eigenvectores de la matriz laplaciana
3. Mapear los vértices a los elementos del eigenvector $\mathbf{e}$ asociado al segundo eigenvalor
4. Ordenar los elementos del vector $\mathbf{e}$
5. Agrupar vértices dividiendo el vector ordenado en dos
    - Por ejemplo usando el 0 o la mediana.

Partición espectral: otro ejemplo con 2 grupos



$$
\mathsf{L} = \begin{bmatrix} 3 & -1 & -1 & -1 & 0 & 0 \\ -1 & 2 & -1 & -0 & 0 & 0 \\ -1 & -1 & 3 & 0 & 0 & -1 \\ -1 & 0 & 0 & 3 & -1 & -1 \\ 0 & 0 & 0 & -1 & 2 & -1 \\ 0 & 0 & -1 & -1 & -1 & 3 \end{bmatrix}
$$

λ	0	1	2	3	4	5
x	1	1	-5	-1	-1	-1
	1	2	4	-2	1	0
	1	1	1	3	-1	1
	1	-1	-5	-1	1	1
	1	-2	4	-2	-1	0
	1	-1	1	3	1	-1

Partición espectral: ejemplo con 2 grupos

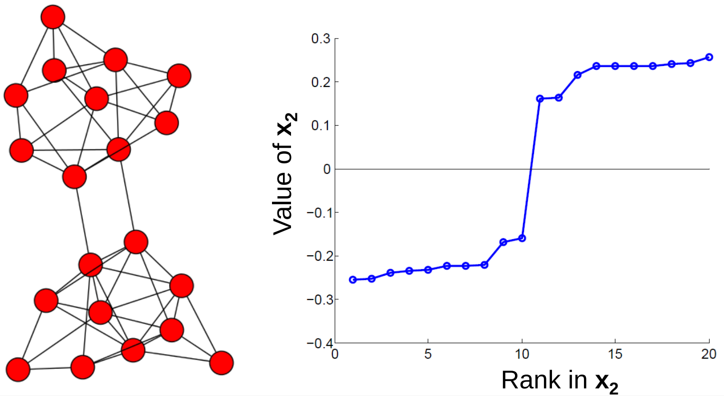


Figura tomada de diapositivas por J. Leskovec, A. Rajaraman, J. Ullman: Mining of Massive Datasets, `http://www.mmds.org`

Partición espectral: ejemplo con 4 grupos (1)

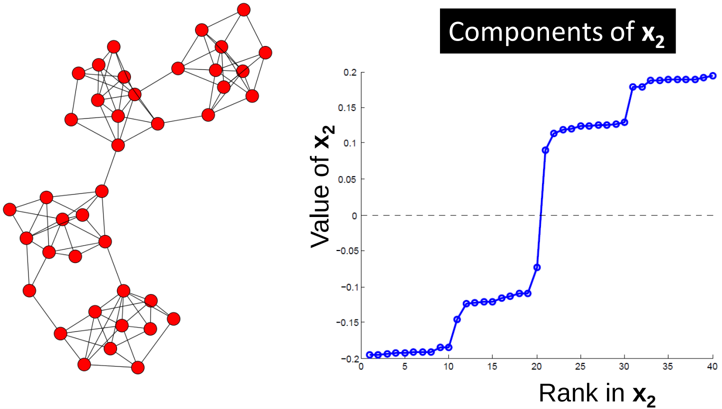


Figura tomada de diapositivas por J. Leskovec, A. Rajaraman, J. Ullman: Mining of Massive Datasets, `http://www.mmds.org`

Partición espectral: ejemplo con 4 grupos (2)

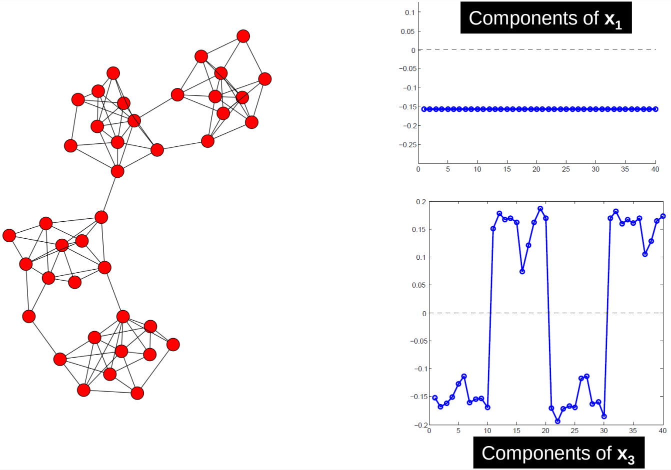


Figura tomada de diapositivas

por J. Leskovec, A. Rajaraman, J. Ullman: Mining of Massive Datasets, `http://www.mmds.org`

Partición de grafos en K grupos

- Partición recursiva
  - Divide el grafo en dos grupos, posteriormente se aplica el algoritmo en cada grupo generando a su vez dos subgrupos cada uno y así sucesivamente.
- Múltiples eigenvectores
  - Se representa cada vértice usando los elementos de múltiples eigenvectores (asociados a los eigenvalores $\lambda_2, \lambda_3, \ldots, \lambda_K$).
  - Se agrupan las representaciones usando algún algoritmo de agrupamiento (por ej. K-medias).

frame