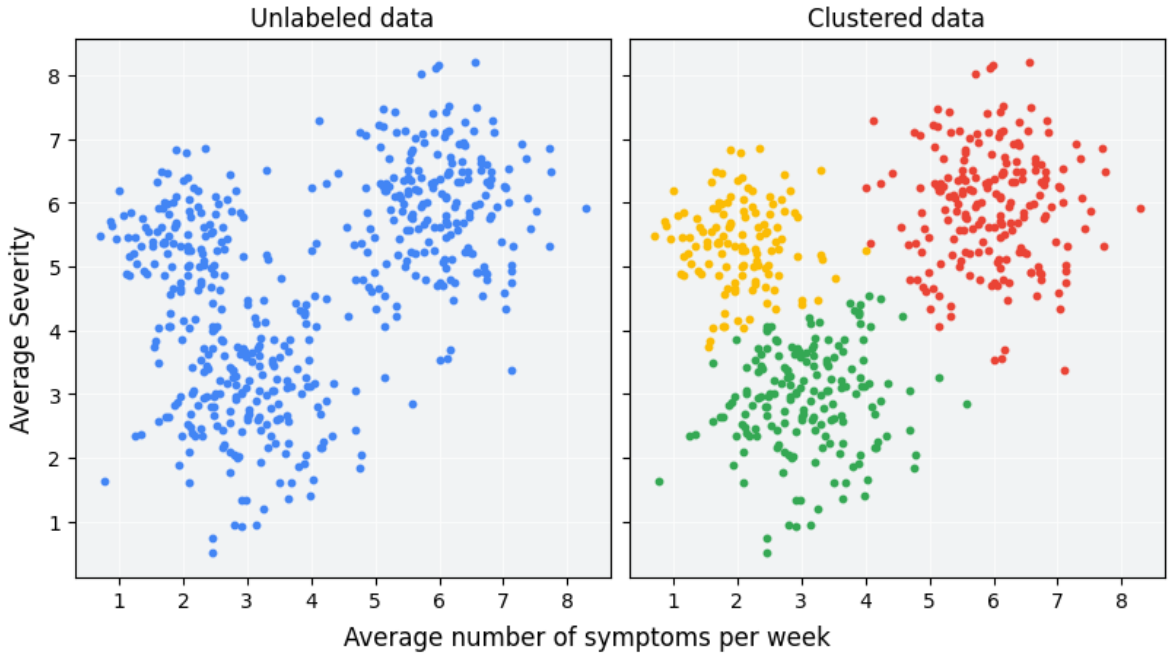
Trabajo Práctico 2: Detección de comunidades con métodos espectrales

Álgebra Lineal Computacional

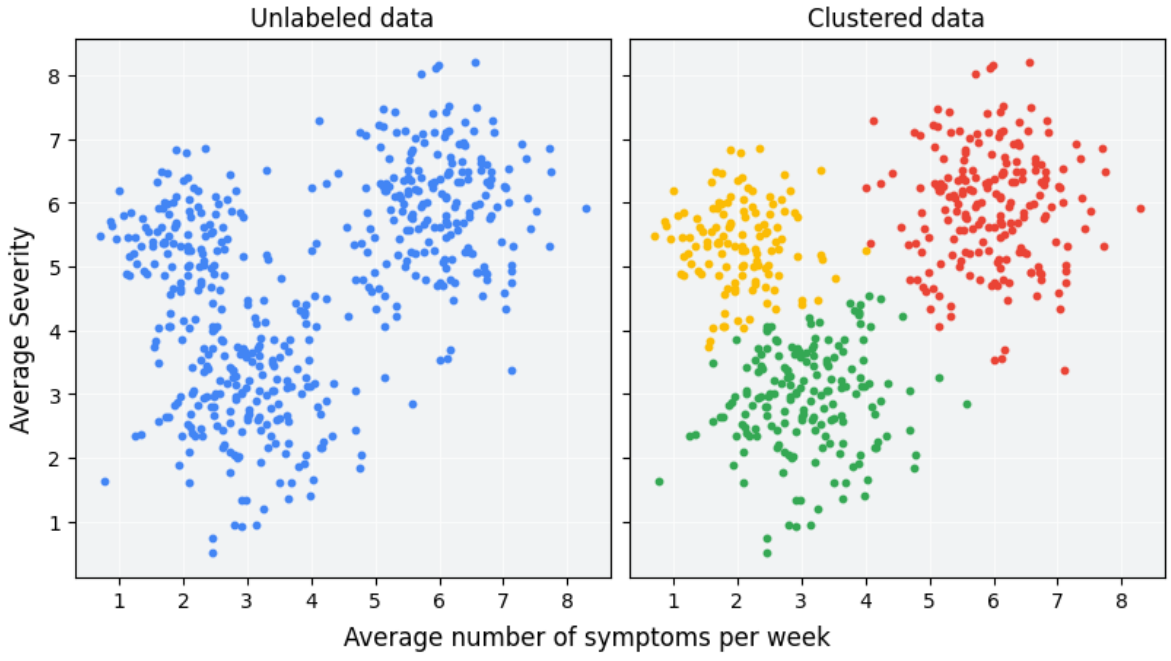
Computación + Ciencias de Datos
Facultad de Ciencias Exactas y Naturales
Universidad de Buenos Aires

1er Cuatrimestre 2025

Segmentos y comunidades

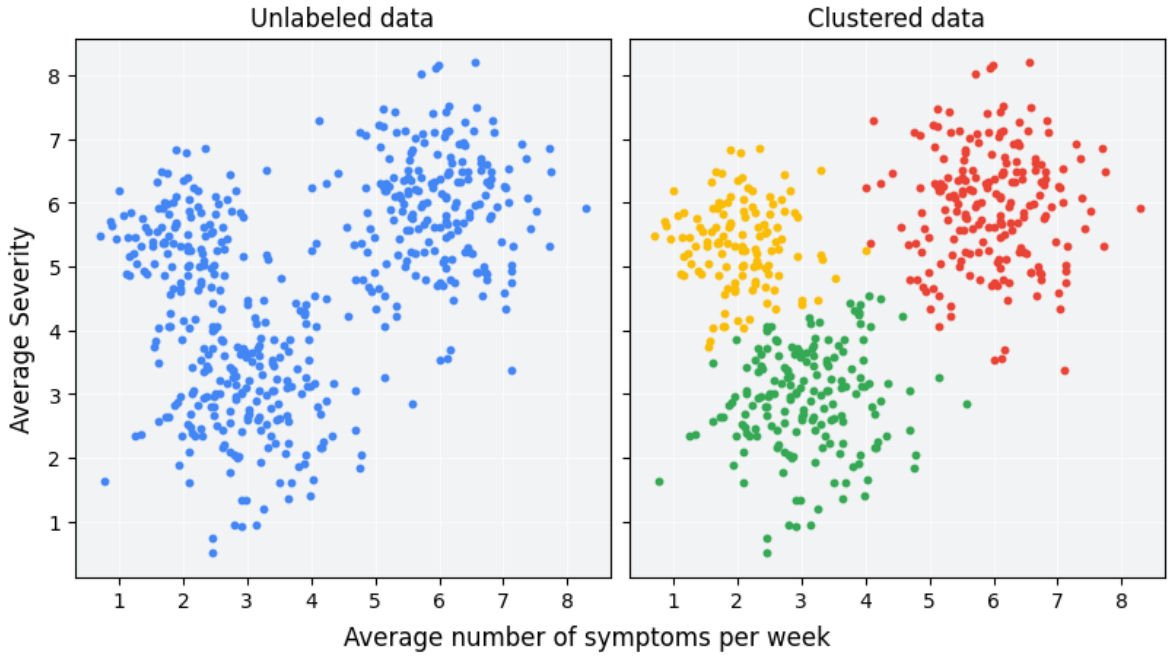


Segmentos y comunidades



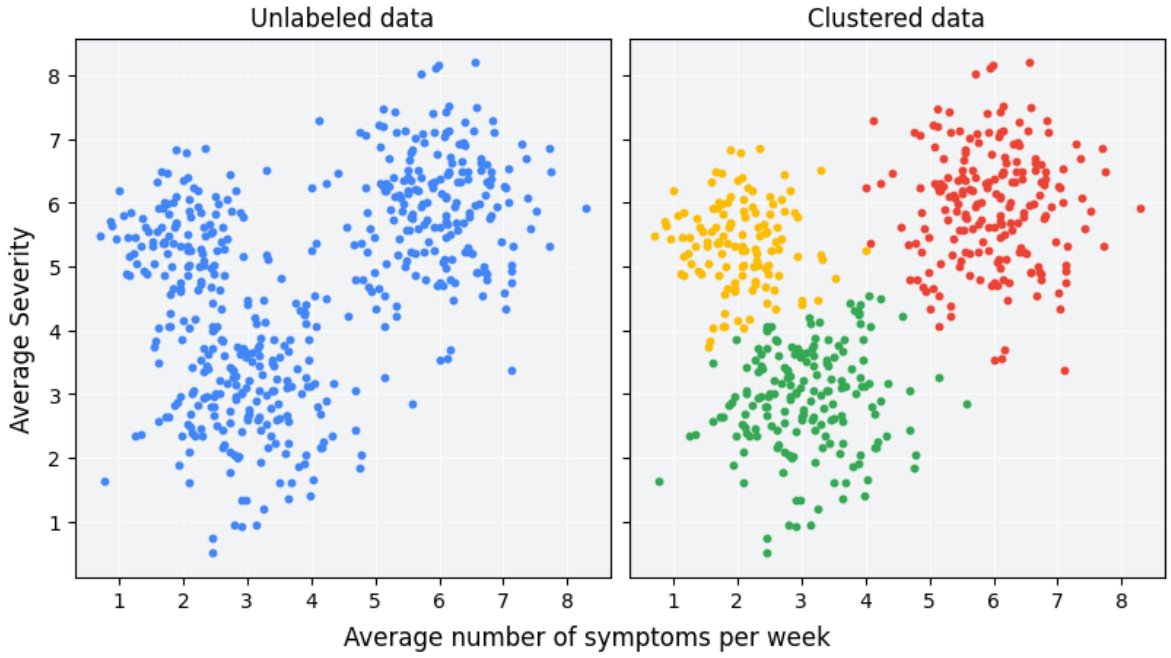
- Las comunidades son segmentos de un dataset cómo cualquier otro.

Segmentos y comunidades



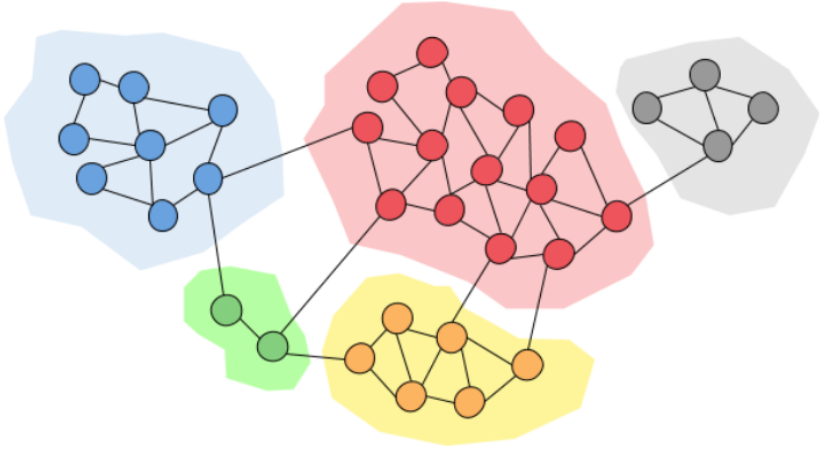
- Las comunidades son segmentos de un dataset cómo cualquier otro.
- Los segmentos son una subdivisión "natural" de los datos, y nos pueden ayudar a encontrar jerarquías o identificar comportamientos típicos.

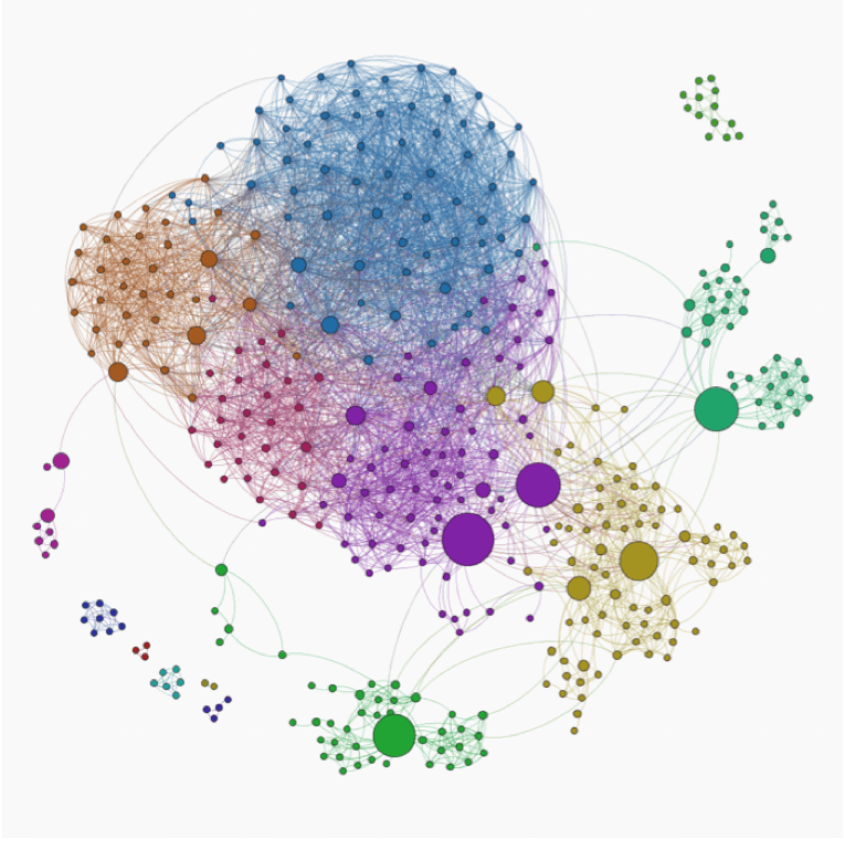
Segmentos y comunidades



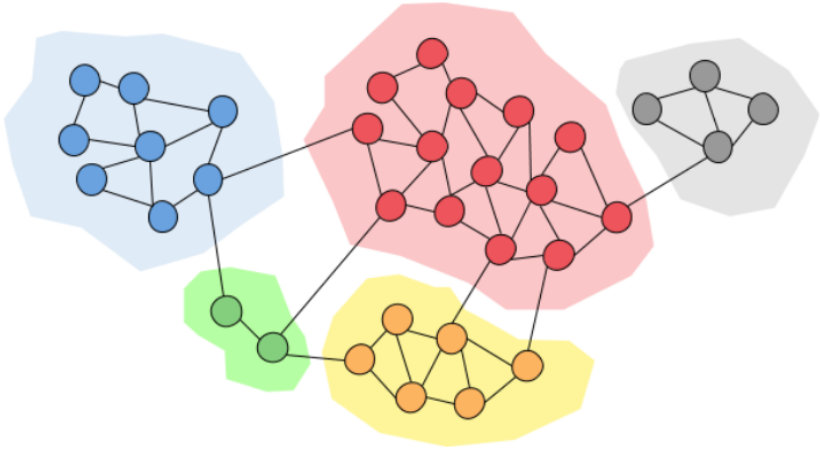
- Las comunidades son segmentos de un dataset cómo cualquier otro.
- Los segmentos son una subdivisión "natural" de los datos, y nos pueden ayudar a encontrar jerarquías o identificar comportamientos típicos.
- Dependiendo del problema de interés, hay **muchas** heurísticas y métodos formales para encontrar estos grupos.

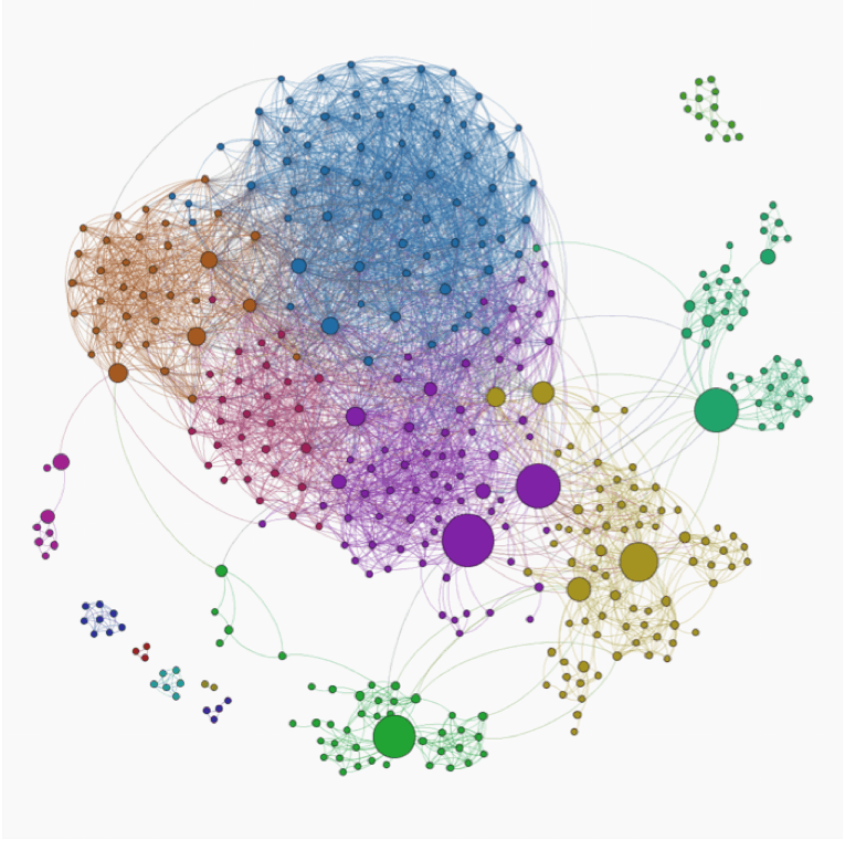
¿Qué hay de nuevo, grafo?





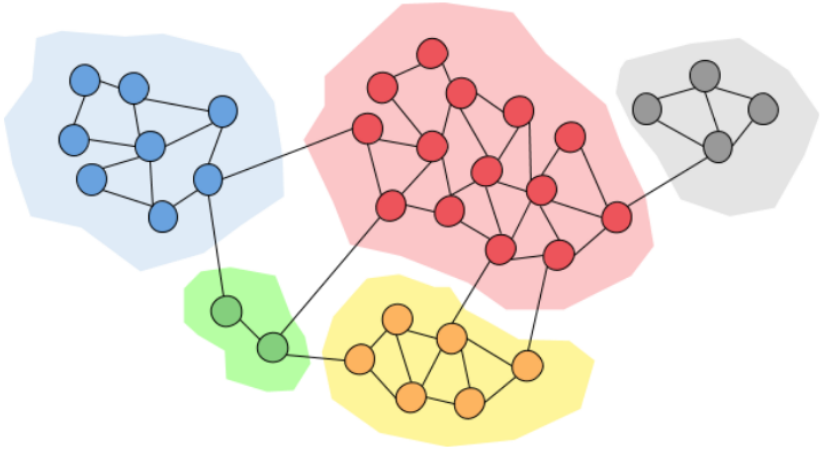
¿Qué hay de nuevo, grafo?

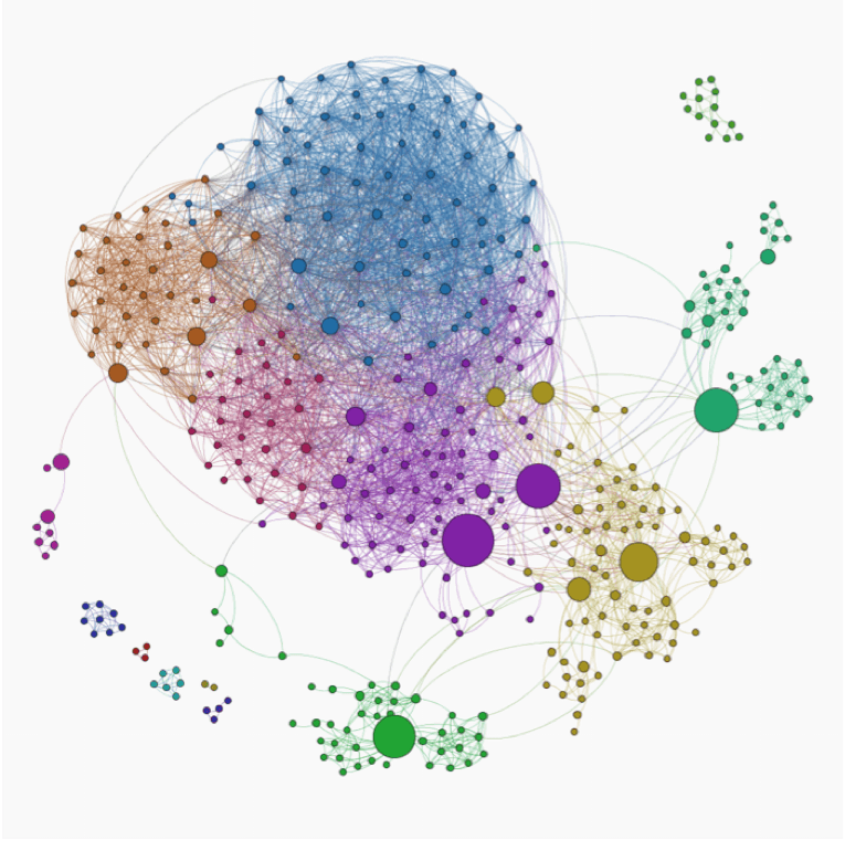




- Si pensamos en el algoritmo de *k-means*, la clasificación se basa en la distancia.

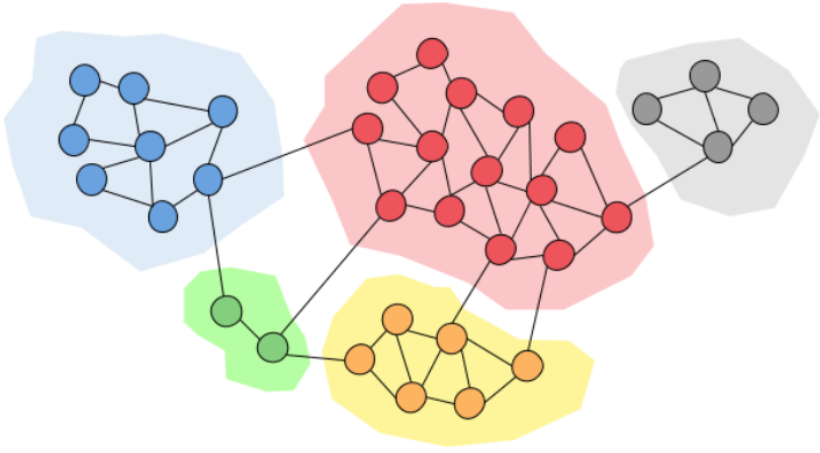
¿Qué hay de nuevo, grafo?

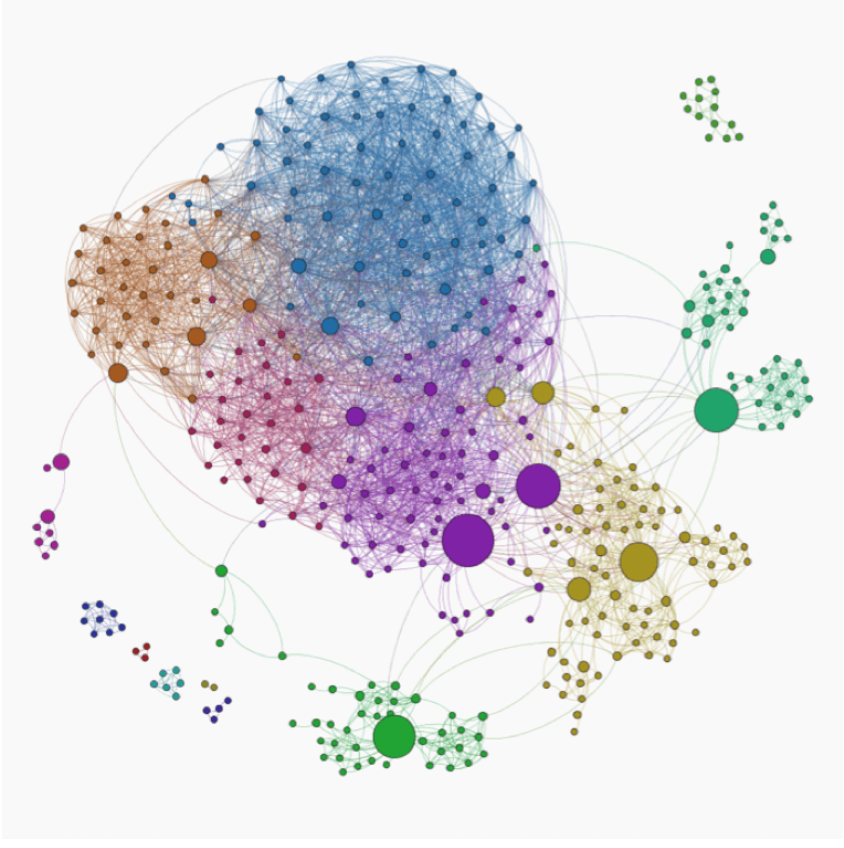




- Si pensamos en el algoritmo de *k-means*, la clasificación se basa en la distancia.
- En un grafo la distancia no necesariamente es una buena medida de similaridad.

¿Qué hay de nuevo, grafo?





- Si pensamos en el algoritmo de *k-means*, la clasificación se basa en la distancia.
- En un grafo la distancia no necesariamente es una buena medida de similaridad.
- La existencia de multiples caminos puede indicar una correlación fuerte, aunque la distancia sea la misma.

Dos posibles respuestas

En lugar de buscar minimizar distancias a un centro, muchas propuestas se han enfocado en buscar encontrar grupos que estén más conectados entre sí que con otros grupos.

Dos posibles respuestas

En lugar de buscar minimizar distancias a un centro, muchas propuestas se han enfocado en buscar encontrar grupos que estén más conectados entre sí que con otros grupos.
En este TP vamos a usar dos de ellas:

- El laplaciano L, que vamos a usar para buscar cortes mínimos de una red.
- La modularidad Q que vamos a usar para encontrar grupos que estén más conectados que lo que esperaríamos por azar.

Dos posibles respuestas

En lugar de buscar minimizar distancias a un centro, muchas propuestas se han enfocado en buscar encontrar grupos que estén más conectados entre sí que con otros grupos.
En este TP vamos a usar dos de ellas:

- El laplaciano L, que vamos a usar para buscar cortes mínimos de una red.
- La modularidad Q que vamos a usar para encontrar grupos que estén más conectados que lo que esperaríamos por azar.
- Hay otras opciones, como buscar regiones donde los caminantes al azar queden *atrapados* por tiempos largos una vez que entran.

Bisección de redes

Los métodos que aplicaremos en este TP se basan en *biseccionar* recursivamente la red. Es decir que **siempre vamos a estar pensando en partir una red en dos grupos**.

Bisección de redes

Los métodos que aplicaremos en este TP se basan en *biseccionar* recursivamente la red. Es decir que **siempre vamos a estar pensando en partir una red en dos grupos**.

Si tenemos N vertices en la red, el vector $\mathbf{s} \in \mathbb{R}^N$ indica pertenencia a un grupo, con $\mathbf{s}_i = 1$ si pertenecen al grupo 1 y $\mathbf{s}_i = -1$ si pertenece al grupo 2.

Bisección de redes

Los métodos que aplicaremos en este TP se basan en *biseccionar* recursivamente la red. Es decir que **siempre vamos a estar pensando en partir una red en dos grupos**.

Si tenemos N vertices en la red, el vector $\mathbf{s} \in \mathbb{R}^N$ indica pertenencia a un grupo, con $\mathbf{s}_i = 1$ si pertenecen al grupo 1 y $\mathbf{s}_i = -1$ si pertenece al grupo 2.

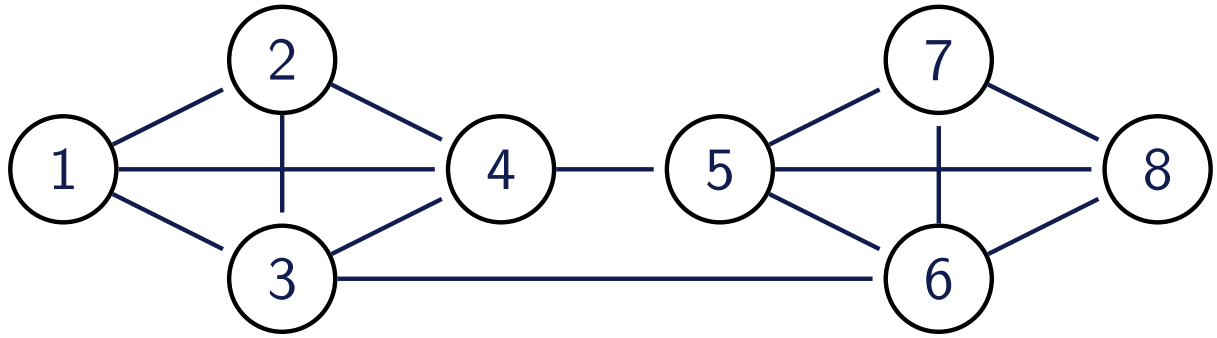
Por ejemplo $\mathbf{s} = (1, 1, 1, 1)^t$ indica que hay cuatro vertices y todos están en la misma comunidad. $\mathbf{s} = (1, 1, -1, -1)^t$ indica que los primeros dos están en la comunidad 1 y los últimos dos en la comunidad 2.

Corte mínimo y laplaciano

Una estrategia es buscar el *corte mínimo* de la red: la mínima cantidad de conexiones que separan dos grupos. Se puede mostrar que esto es igual a encontrar el mínimo de:

$$\Lambda = \frac{1}{4}\mathbf{s}^t L \mathbf{s}$$
$$L = K - A$$

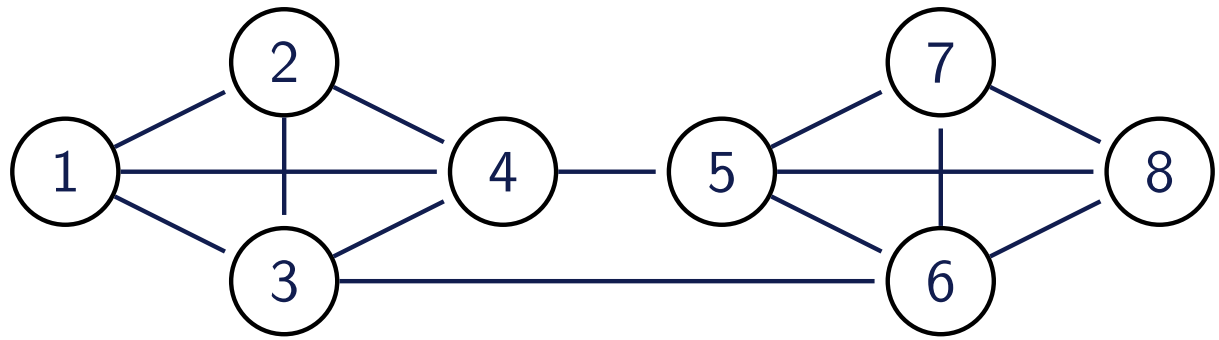
con A la matriz de adyacencia y K la matriz de grado que definimos en el TP1.



Modularidad y modelos nulos

Otra estrategia es buscar grupos que estén más conectados entre sí que con otros. *Noten que esto invierte el foco del problema.*
La modularidad Q se calcula como

$$Q = \frac{1}{2E}\mathbf{s}^t R \mathbf{s}$$
$$R = A - P$$
$$P = \frac{1}{2E} diag(K) diag(K)^t$$



Resolución por métodos espectrales

- En ambos casos queremos encontrar un $\mathbf{s}$ tal que $\mathbf{s}^t M \mathbf{s}$ sea extremo (máximo o mínimo), con $\mathbf{s}$ compuesto de 1s y -1s. Además en ambos ejemplos la matriz M es simétrica si A es simétrica (no dirigida).

Resolución por métodos espectrales

- En ambos casos queremos encontrar un $\mathbf{s}$ tal que $\mathbf{s}^t M \mathbf{s}$ sea extremo (máximo o mínimo), con $\mathbf{s}$ compuesto de 1s y -1s. Además en ambos ejemplos la matriz M es simétrica si A es simétrica (no dirigida).
- Como M es simétrica, habrá una base de autovectores ortogonales asociada, y entonces $\mathbf{s} = \sum_{i=1}^{N} a_i \mathbf{v}_i$. Más aún,

$$\mathbf{s}^t M \mathbf{s} = \sum_{i=1}^{N} a_i^2 \lambda_i$$

.

Resolución por métodos espectrales

- En ambos casos queremos encontrar un $\mathbf{s}$ tal que $\mathbf{s}^t M \mathbf{s}$ sea extremo (máximo o mínimo), con $\mathbf{s}$ compuesto de 1s y -1s. Además en ambos ejemplos la matriz M es simétrica si A es simétrica (no dirigida).
- Como M es simétrica, habrá una base de autovectores ortogonales asociada, y entonces $\mathbf{s} = \sum_{i=1}^{N} a_i \mathbf{v}_i$. Más aún,

$$\mathbf{s}^t M \mathbf{s} = \sum_{i=1}^{N} a_i^2 \lambda_i$$

.

- La propuesta es entonces elegir los a_i de forma que $\mathbf{s}$ sea lo más parecido a $\mathbf{v}_1$ (o $\mathbf{v}_N$) posible.

$$(\mathbf{s})_i = signo((\mathbf{v}_1)_i)$$

Como estrategia iterativa

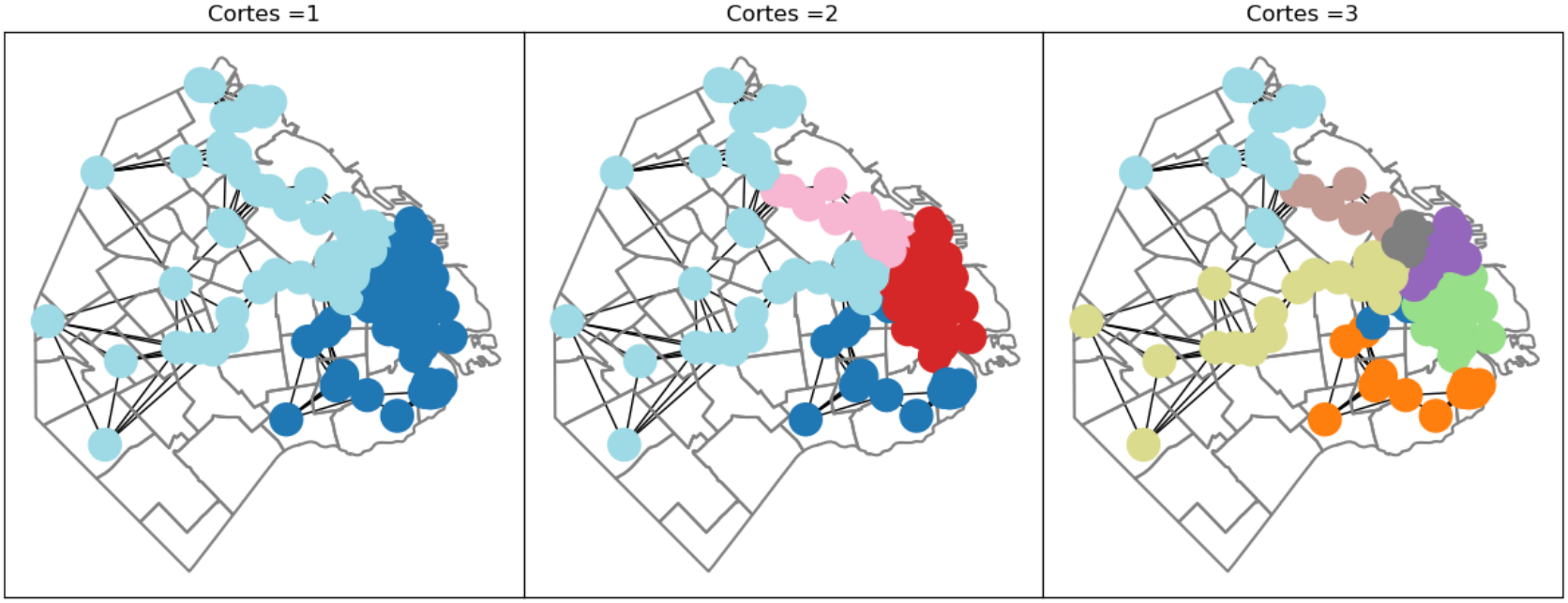
- Una vez que definimos la heuristica para partir la red en dos grupos, podemos repetir el algoritmo recursivamente sobre cada partición hasta encontrar una solución deseada (ej: número de comunidades, o aumento de la modularidad).

Como estrategia iterativa

- Una vez que definimos la heuristica para partir la red en dos grupos, podemos repetir el algoritmo recursivamente sobre cada partición hasta encontrar una solución deseada (ej: número de comunidades, o aumento de la modularidad).
- Siempre vamos a usar como herramienta principal al método de la potencia para calcular los autovectores que nos interesan.

Como estrategia iterativa

- Una vez que definimos la heuristica para partir la red en dos grupos, podemos repetir el algoritmo recursivamente sobre cada partición hasta encontrar una solución deseada (ej: número de comunidades, o aumento de la modularidad).
- Siempre vamos a usar como herramienta principal al método de la potencia para calcular los autovectores que nos interesan.



Detalles finales

- Con la modularidad, el objetivo es alcanzar maximizar Q posible. El criterio de parada es que la partición no la aumente.

Detalles finales

- Con la modularidad, el objetivo es alcanzar maximizar Q posible. El criterio de parada es que la partición no la aumente.
- Con el corte mínimo, siempre es posible encontrar un corte. El criterio de parada es realizar un número de cortes prefijado.

Detalles finales

- Con la modularidad, el objetivo es alcanzar maximizar Q posible. El criterio de parada es que la partición no la aumente.
- Con el corte mínimo, siempre es posible encontrar un corte. El criterio de parada es realizar un número de cortes prefijado.
- Esto lleva a realizar una optimización distinta para cada método:
    - En el caso de la modularidad, siempre buscamos maximizar, y por lo tanto usaremos el método de la potencia *básico*.

Detalles finales

- Con la modularidad, el objetivo es alcanzar maximizar Q posible. El criterio de parada es que la partición no la aumente.
- Con el corte mínimo, siempre es posible encontrar un corte. El criterio de parada es realizar un número de cortes prefijado.
- Esto lleva a realizar una optimización distinta para cada método:
  - En el caso de la modularidad, siempre buscamos maximizar, y por lo tanto usaremos el método de la potencia *básico*.
  - En el caso del corte mínimo, **el laplaciano tiene** $\lambda_i \geq 0 \ \forall i$, y $\mathbf{1}$ siempre es un autovector con autovalor 0. Es decir que *buscaremos el segundo autovector más chico*.

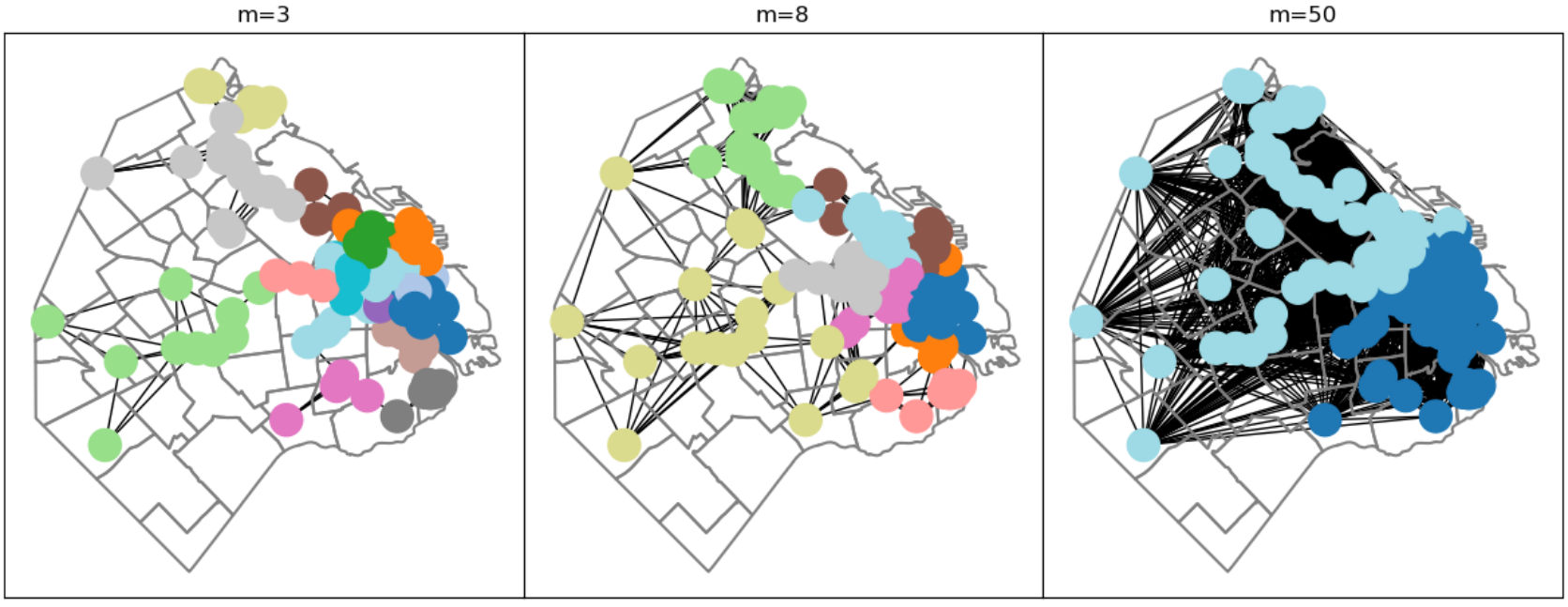
Detalles finales

- Con la modularidad, el objetivo es alcanzar maximizar Q posible. El criterio de parada es que la partición no la aumente.
- Con el corte mínimo, siempre es posible encontrar un corte. El criterio de parada es realizar un número de cortes prefijado.
- Esto lleva a realizar una optimización distinta para cada método:
  - En el caso de la modularidad, siempre buscamos maximizar, y por lo tanto usaremos el método de la potencia *básico*.
  - En el caso del corte mínimo, **el laplaciano tiene** $\lambda_i \geq 0 \ \forall i$, y $\mathbf{1}$ siempre es un autovector con autovalor 0. Es decir que *buscaremos el segundo autovector más chico*.
- Ambas matrices son simétricas y por lo tanto admiten bases de autovectores ortonormales, así como el uso de la deflación de Hotelling.

Detalles finales

- Con la modularidad, el objetivo es alcanzar maximizar Q posible. El criterio de parada es que la partición no la aumente.
- Con el corte mínimo, siempre es posible encontrar un corte. El criterio de parada es realizar un número de cortes prefijado.
- Esto lleva a realizar una optimización distinta para cada método:
  - En el caso de la modularidad, siempre buscamos maximizar, y por lo tanto usaremos el método de la potencia *básico*.
  - En el caso del corte mínimo, **el laplaciano tiene** $\lambda_i \geq 0 \ \forall i$, y $\mathbf{1}$ siempre es un autovector con autovalor 0. Es decir que *buscaremos el segundo autovector más chico*.
- Ambas matrices son simétricas y por lo tanto admiten bases de autovectores ortonormales, así como el uso de la deflación de Hotelling.
- Sobre todo cuando hay muchas particiones, los resultados pueden ser inestables: *¡revisen la reproducibilidad de los resultados y usen semillas!*

Un ejemplo de partición con modularidad



- El gráfico muestra comunas para $m = 3, 8, 50$ conexiones, con la matriz simetrizada $A' = \lceil \frac{1}{2}(A + A^t) \rceil$.
- Conforme aumenta la cantidad de conexiones, la cantidad de grupos disminuye: 15, 9 y 2.