DECSAI
Departamento de Ciencias de la Computación e I.A.
Universidad de Granada

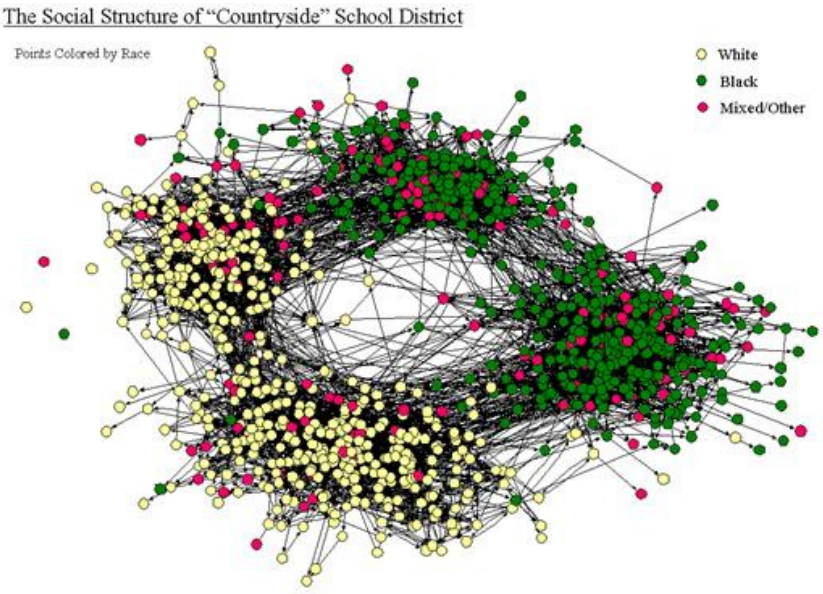
Detección de comunidades

© Fernando Berzal, berzal@acm.org

Detección de comunidades

- El problema de la detección de comunidades
- Métodos jerárquicos
- Métodos modulares
- Métodos particionales
- Métodos espectrales
- Evaluación de resultados
- Detección de comunidades con solapamiento
  - Clique Percolation Method [CPM]
  - BigCLAM
- Aplicaciones

Detección de comunidades

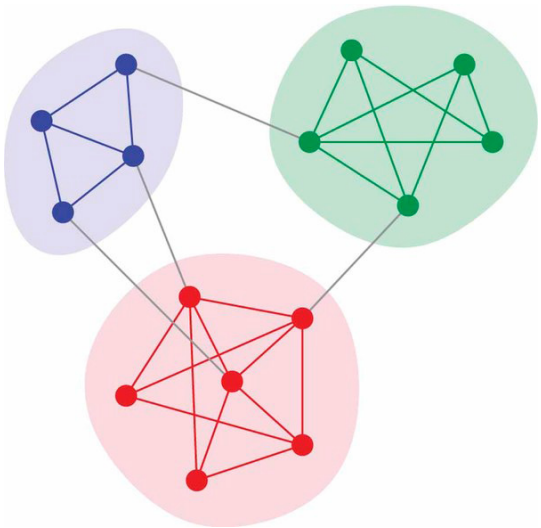


Red social FOAF [Friend of a Friend]

Detección de comunidades

El problema

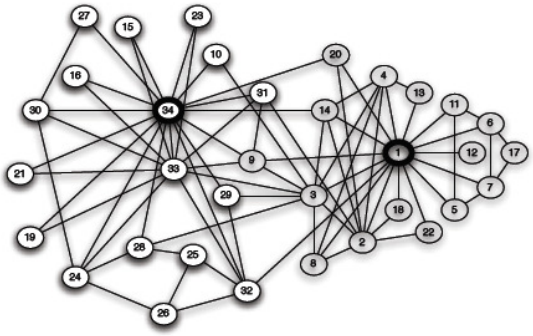
Agrupamiento [clustering] en redes



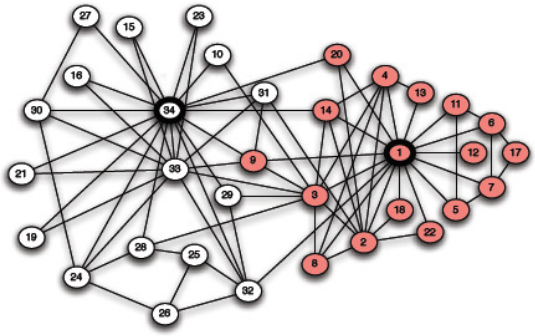
Detección de comunidades

Ejemplo

Club de kárate



(a) *Karate club network*



(b) *After a split into two clubs*

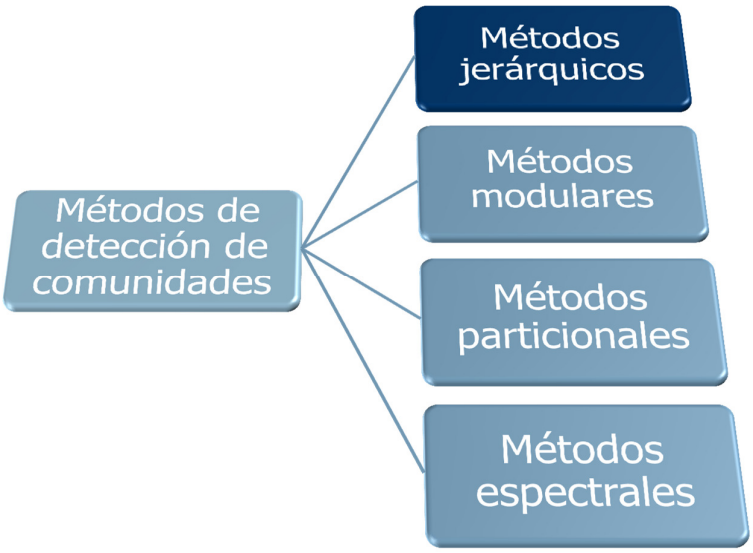
W. W. Zachary:
An information flow model for conflict and fission in small groups,
Journal of Anthropological Research 33:452-473 (1977)

Detección de comunidades

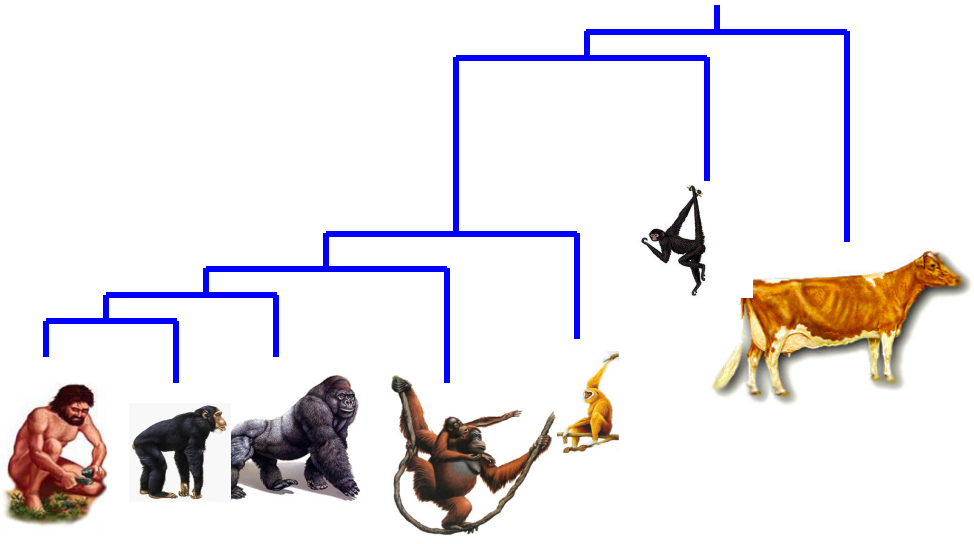
Heurísticas

- Enlaces mutuos & vecinos compartidos
- Frecuencia de enlaces dentro de una comunidad (cliques & k-cores)
- Cercanía [closeness] de los miembros de una comunidad (n-cliques)
- Frecuencia relativa de los enlaces comunidad (enlaces entre miembros de una comunidad frente a enlaces con "no-miembros")

Detección de comunidades



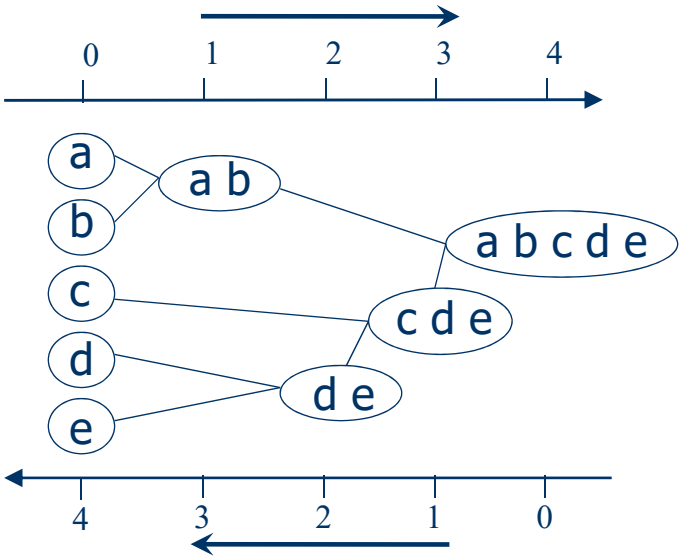
Métodos jerárquicos



DENDROGRAMA: La similitud entre dos objetos viene dada por la "altura" del nodo común más cercano.

Métodos jerárquicos

Métodos aglomerativos
(AGNES: AGglomerative NESting)



Métodos divisivos
(DIANA: Divisive ANAlysis)

Métodos jerárquicos

Métodos jerárquicos aglomerativos

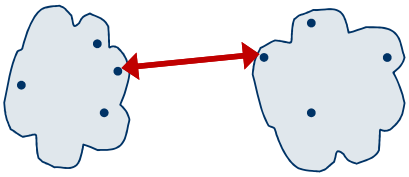
Calcular la matriz de similitud/distancias
Inicialización: Cada caso, un cluster
Repetir
    Combinar los dos clusters más cercanos
    Actualizar la matriz de similitud/distancias
hasta que sólo quede un cluster

- Estrategia de control irrevocable (greedy):
  Cada vez que se unen dos clusters,
  no se reconsidera otra posible unión.

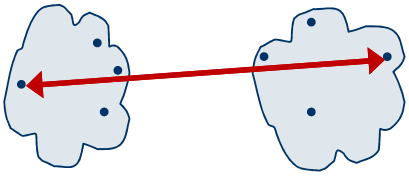
Métodos jerárquicos

¿Cómo se mide la distancia entre clusters?

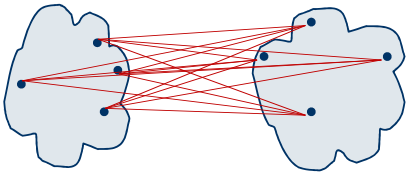
- Mínimo [single-link]



- Máximo (diámetro) [complete-link]



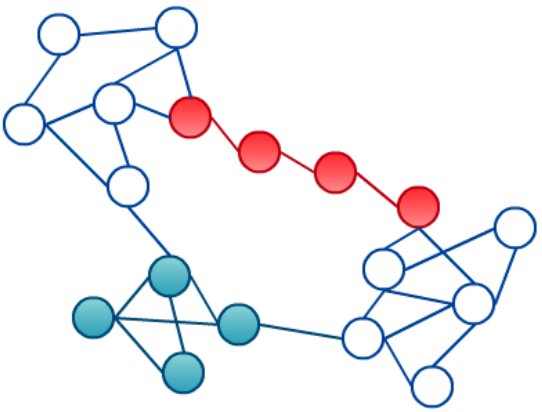
- Promedio [averaga-link]



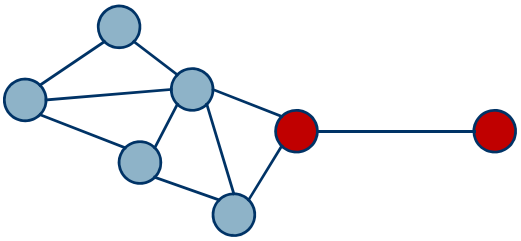
Métodos jerárquicos

Problemas

- Simple-link: Encadenamiento



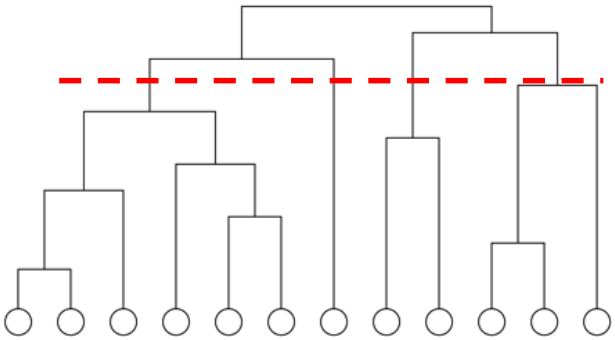
- Complete-link: Existencia de outliers



Métodos jerárquicos

Método de Newman & Girvan

Algoritmo jerárquico divisivo

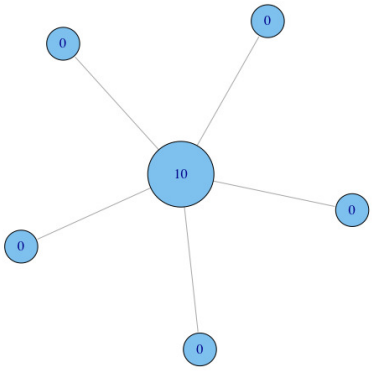


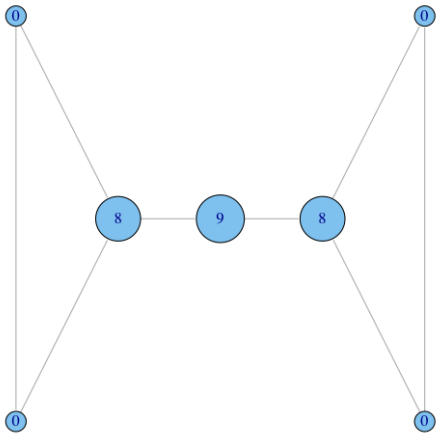
Michelle Girvan & Mark E.J. Newman:
"Community structure in social and biological networks"
PNAS **99**(12):7821–7826, 2002
doi:10.1073/pnas.122653799

Métodos jerárquicos

Método de Newman & Girvan

Betweenness [intermediación]



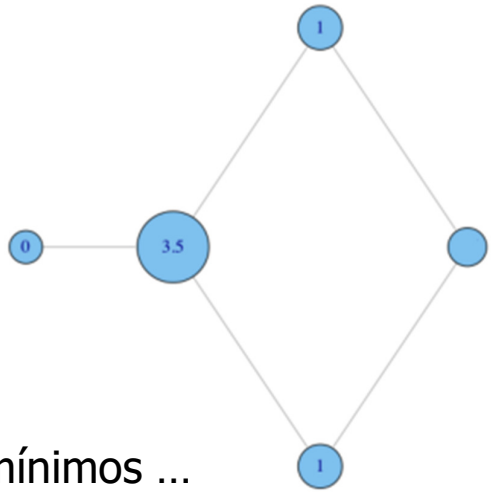


IDEA:
Número de caminos mínimos que pasan por cada nodo como medida de la importancia de ese nodo.

Métodos jerárquicos

Método de Newman & Girvan

Betweenness [intermediación]



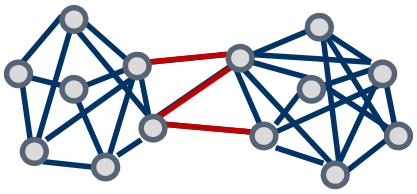
Asignación parcial de crédito
cuando existen varios caminos mínimos …

La misma idea se puede extender para evaluar la importancia de los enlaces en función del número de caminos mínimos de los que forman parte.

Métodos jerárquicos

Método de Newman & Girvan

Clústering jerárquico utilizando "edge betweenness"



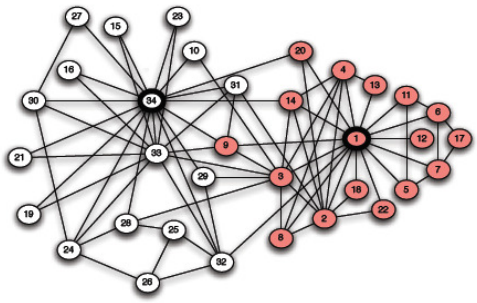
compute the betweenness of all edges
while (betweenness of any edge > threshold)
    remove edge with highest betweenness
    recalculate betweenness

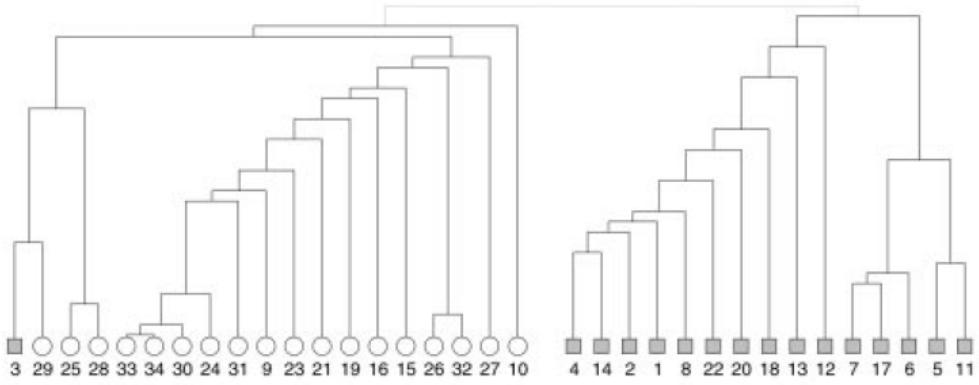
Ineficiente debido a la necesidad de recalcular el "edge betweenness" de todos los enlaces en cada iteración.

Métodos jerárquicos

Método de Newman & Girvan

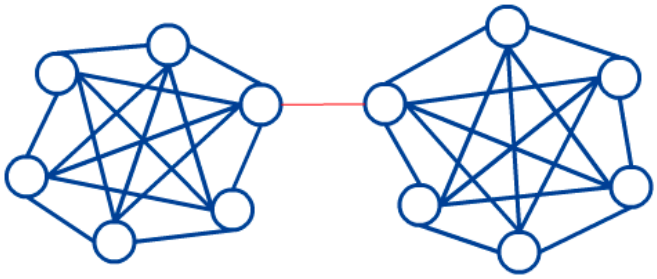
Red del club de kárate





Métodos jerárquicos

Método de Radicchi



IDEA:

Una comunidad contiene nodos muy interconectados entre sí, con muchos ciclos; sin embargo, los enlaces que conectan unas comunidades con otras se ven involucrados en menos ciclos.

Métodos jerárquicos

Método de Radicchi

Coeficiente de agrupamiento

nbr(n)	Vecinos del nodo n en la red.
k	Número de vecinos de u, i.e. |nbr(n)|.
max(n)	Número máximo de enlaces entre los vecinos de n, e.g. k*(k-1)/2.

Coeficiente de clustering para el nodo n:

CC(n) = (#enlaces entre vecinos de n) / max(n)

Métodos jerárquicos

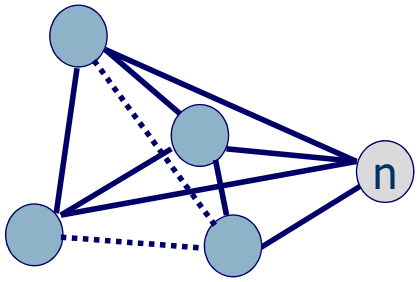
Método de Radicchi

Coeficiente de agrupamiento

k = 4

m = 6

CC(n) = 4/6 = 0.66



0 <= CC(n) <= 1

Similitud del conjunto de vecinos de n a un clique.

Métodos jerárquicos

Método de Radicchi

Coeficiente de agrupamiento de los enlaces

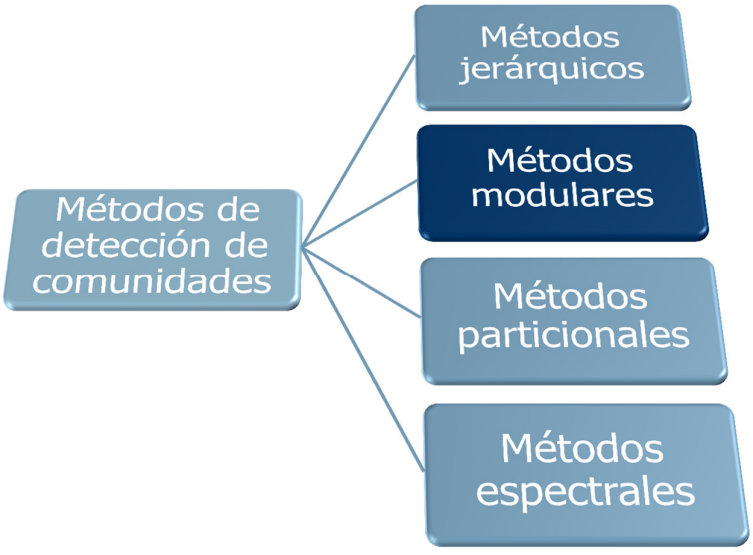
$$C_{ij} = \frac{z_{ij} + 1}{\min(k_i - 1, k_j - 1)}$$

k_i Grado del nodo i

z_{ij} Número de triángulos en los que participan los nodos i y j

Más eficiente que el método de Newman & Girvan.

Detección de comunidades



Métodos modulares

ORIGEN

Medida de modularidad Q

IDEA

Uso del término de "modularidad" como cualquier medida numérica que resulte adecuada para determinar y encontrar comunidades.

La detección de comunidades se convierte en un problema de optimización numérica…

Métodos modulares

Modularidad Q

- Métrica que compara los enlaces internos de una comunidad frente a los enlaces que connectan la comunidad con el resto de la red.

$$Q = \frac{1}{2m} \sum_{vw} \left[A_{vw} - \frac{k_v k_w}{2m} \right] \delta(c_v, c_w)$$

Vértices en la misma comunidad

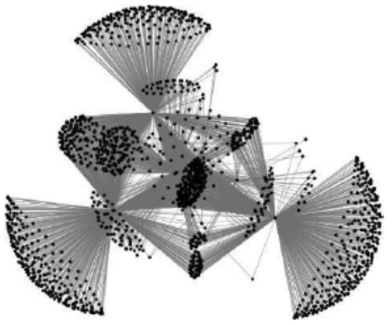
Matriz de adyacencia

Probabilidad de un enlace entre dos vértices (propocional a sus grados)

NOTA: En una red completamente aleatoria, Q=0

Métodos modulares

Algoritmo greedy



start with all vertices as isolates
do
  join clusters with the greatest increase in modularity (ΔQ)
while ($\Delta Q > 0$)

Aaron Clauset, Mark E. J. Newman, Cristopher Moore:
"Finding community structure in very large networks"
Physical Review E 70(6):066111, 2004
doi:10.1103/physreve.70.066111

Métodos modulares

Algoritmo Fast Greedy

FASE 1: Inicialización

Formar pequeños grupos con algún método particional sencillo (tipo K-Means), p.ej. K=n/2

FASE 2: Algoritmo greedy aleatorio

Mientras queden enlaces que mejoren Q:

Seleccionar (de forma aleatoria) un enlace que mejore la modularidad de la red y añadirlo.

Métodos modulares

Algoritmo MultiStep Greedy

IDEA

Se selecciona el enlace que más incrementa la modularidad (ΔQ).

Si no están en contacto, se pueden añadir varios enlaces en la misma iteración del algoritmo greedy.

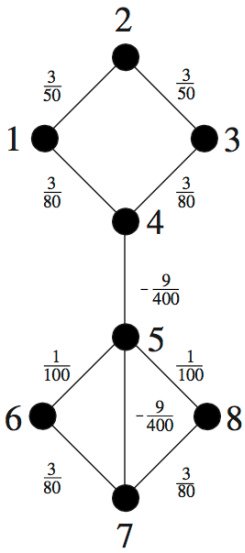
P. Schuetz & A. Caflisch: "Efficient modularity optimization by multistep greedy algorithm and vertex mover refinement" Physical Review E, 77(4):046112, 2008

Métodos modulares

Algoritmo MultiStep Greedy

ESTRUCTURA DE DATOS QMatrix

Mejoras de modularidad asociadas a cada arista (ΔQ_{ij}).



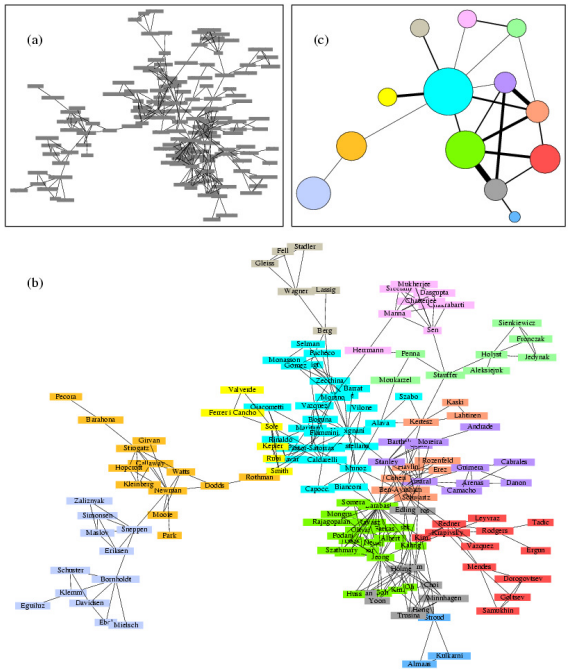
ΔQ_{ij}	i	j
3/50	1	2
3/50	2	3
3/80	1	4
3/80	3	4
3/80	6	7
3/80	7	8
1/100	5	6
1/100	5	8
~~-9/400~~	~~4~~	~~5~~
~~-9/400~~	~~5~~	~~7~~

ALGORITMO

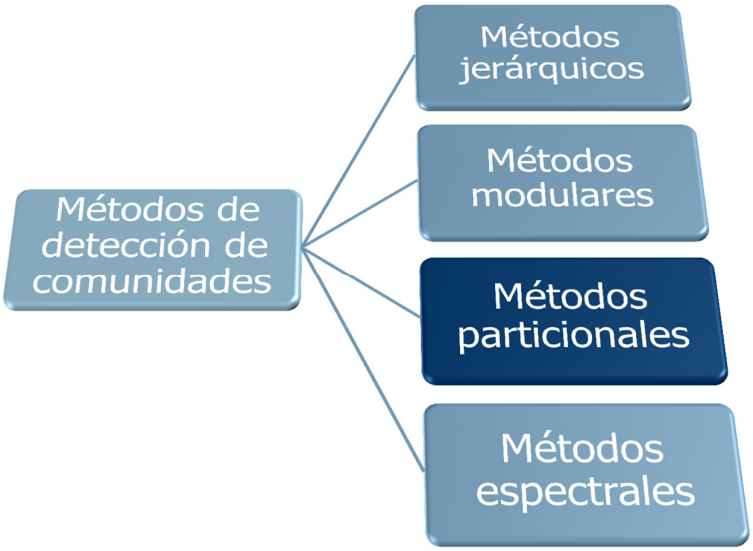
start with all vertices as isolates
do
  compute QMatrix
  sort QMatrix (descending ΔQ, ascending link)
  add non-touching links with greatest increase in modularity (ΔQ_{ij})
while ($\Delta Q_{ij} > 0$ for some link to be added)

Métodos modulares

Aplicación: Visualización de grandes redes (Gephi)



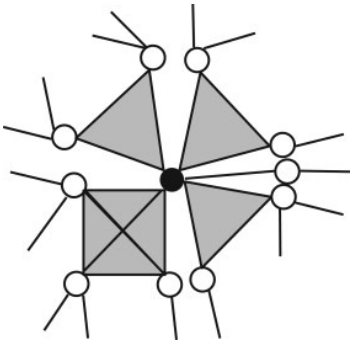
Detección de comunidades



Métodos particionales

Cliques & k-cores

- Cliques (subgrafos completos)
  - La ausencia de un simple enlace descalifica al clique completo
  - Los cliques se solapan.



- K-cores
  (cada nodo, conectado al menos con otros k nodos)

Métodos particionales

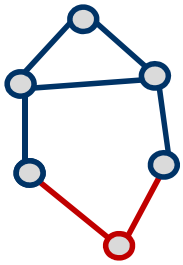
n-cliques

Cualquier pareja de nodos a distancia máxima n

IDEA: Flujo de información a través de intermediarios.

Problemas:

- Diámetro > n
- n-cliques desconectados



2 – clique
diámetro = 3

Camino fuera del 2-clique

Solución: **n-clubs** (subgrafos máximos de diámetro n)

Métodos particionales

Particionamiento sobre un espacio métrico

Técnicas clásicas de clustering basadas en agrupar un conjunto de puntos de un espacio métrico

- **Minimum k-Clustering** (intenta minimizar el diámetro de los clusters)
- **Min-Sum k-Clustering** (intenta maximizar la cohesión dentro de los clusters, i.e. la distancia media entre cada par de nodos dentro de cada clúster).
- **K-Center** (intenta minimizar la distancia máxima del centroide a los demás puntos del clúster).
- **K-Means** (intenta minimizar la distancia media del centroide a los demás puntos del clúster)

Métodos particionales

Particionamiento sobre un espacio métrico

K-Means

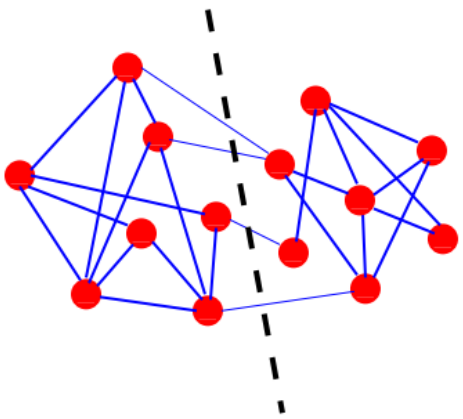
IDEA

- Se transforma la red en un conjunto de puntos de un espacio métrico (p.ej. usando el algoritmo de visualización de redes de Fruchterman-Reingold).
- Se aplica el algoritmo de las k medias.

Métodos particionales

Particionamiento de grafos

Se divide el grafo
en k componentes conexas
intentando minimizar
una **función de corte**.

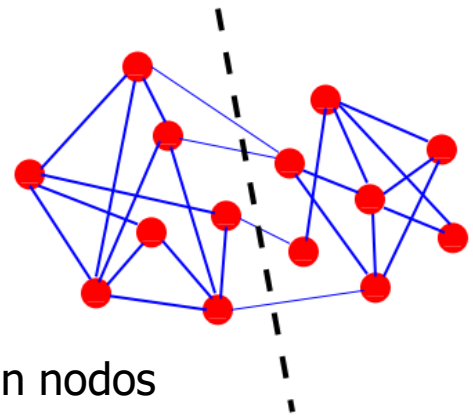


p.ej. Corte mínimo

$$Cut(C_1, C_2, \dots, C_k) = \frac{1}{2}\sum_{i=1}^{k} W(C_i, \overline{C_i}) \qquad W(C_r, C_t) = \sum_{i \in C_r, j \in C_t} a_{ij}$$

Métodos particionales

Particionamiento de grafos
Algoritmo de Kernighan-Lin

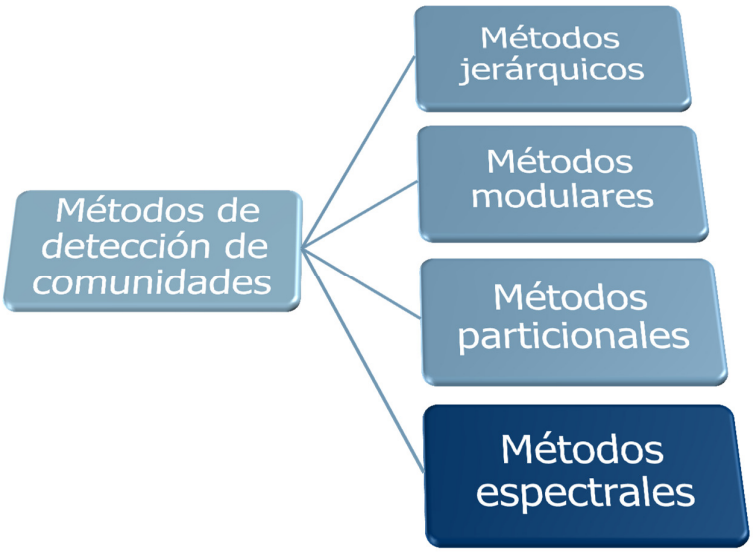


- Bisección mínima (k=2)
- Algoritmo greedy heurístico:
  Iterativamente, se intercambian nodos para minimizar el corte.
- Selección de parejas de nodos de acuerdo a una función de coste asociado al intercambio.

$$g(i,j) = D_i + D_j - 2w_{ij} \qquad I_i = \sum_{j \in A} w_{ij} \qquad E_i = \sum_{j \in B} w_{ij}$$

$$D_i = E_i - I_i$$

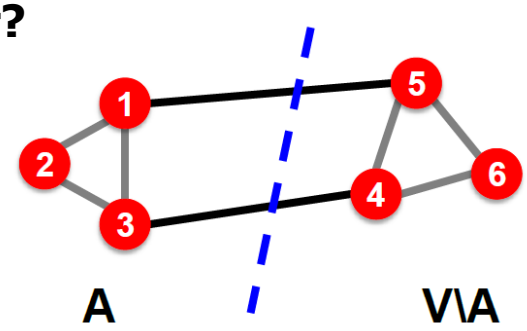
Detección de comunidades



Métodos espectrales

¿Qué hace bueno a un cluster?

- Se maximiza el número de conexiones dentro del cluster.
- Se minimiza el número de conexiones con otros clusters.



$$cut(A) = 2$$

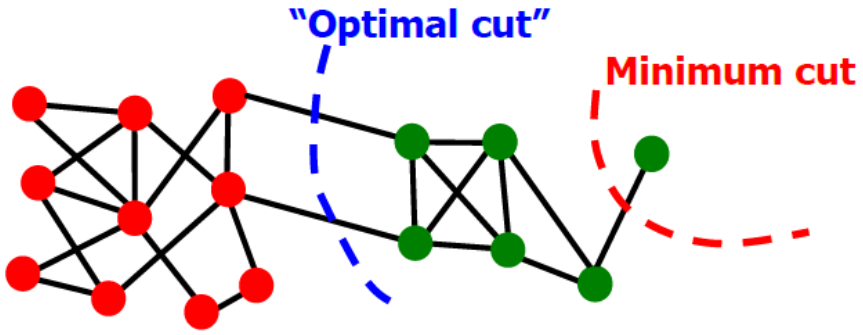
IDEA

Expresar la calidad del cluster como una función del "corte" que separa al cluster del resto de la red.

Métodos espectrales

PROBLEMA

El corte sólo tiene en cuenta conexiones entre clusters.



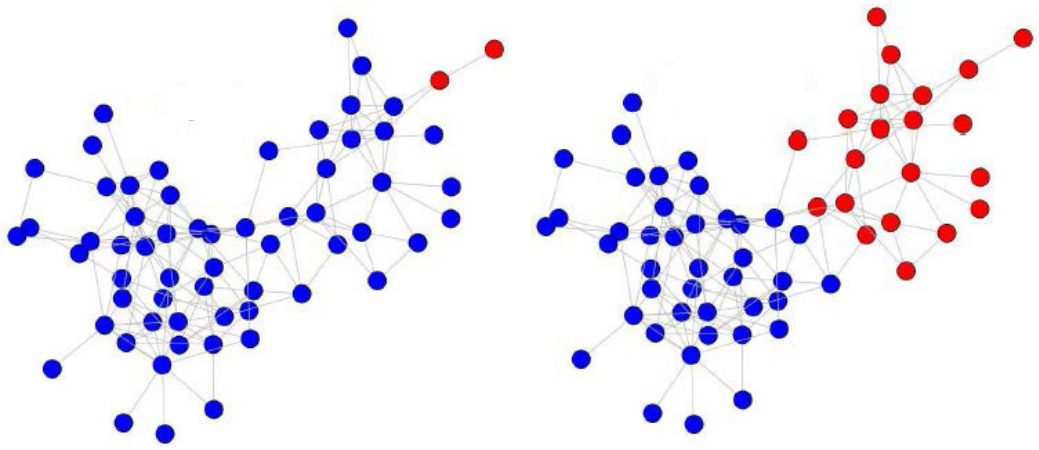
SOLUCIÓN

La **conductancia** (conectividad del grupo con el resto de la red, con respecto a la densidad del grupo) ofrece particiones más balanceadas…

Métodos espectrales

Conductancia

$$\phi(A) = \frac{|\{(i,j) \in E; i \in A, j \notin A\}|}{\min(vol(A),\ 2m - vol(A))}$$

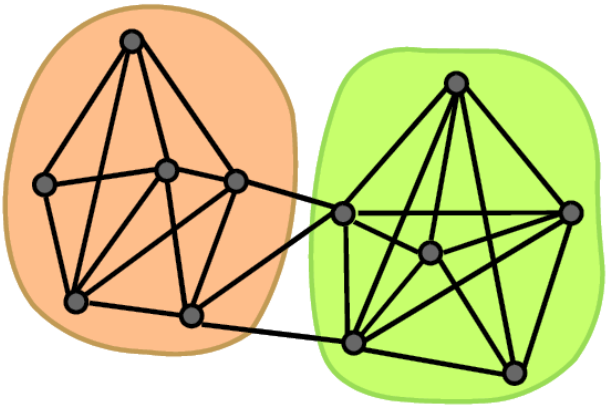


ϕ = 2/4 = 0.5

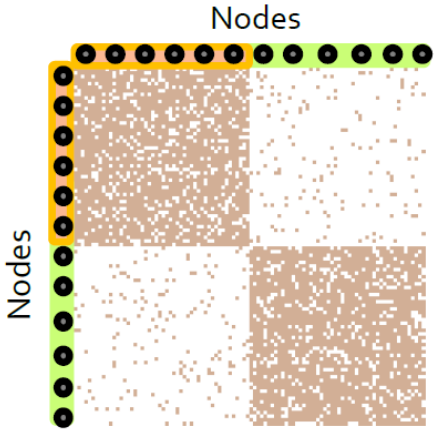
ϕ = 6/92 = 0.065

Métodos espectrales

Encontrar un corte óptimo es un problema NP-duro…



Red



Matriz de adyacencia

Métodos espectrales

A Matriz de adyacencia del grafo G

x Vector de valores asociados a cada nodo de G

Ax Para cada nodo,
suma de los valores asociados a sus vecinos.

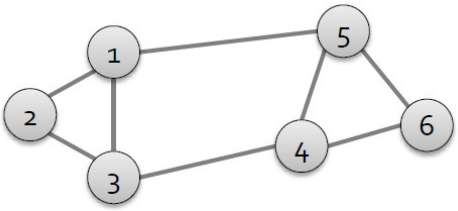
$$\begin{bmatrix} a_{11} & \dots & a_{1n} \\ \vdots & & \vdots \\ a_{n1} & \dots & a_{nn} \end{bmatrix} \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} = \lambda \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix}$$

Teoría espectral de grafos: $Ax = \lambda x$

Análisis del "espectro" de la matriz que representa G.

Métodos espectrales

Matriz de adyacencia A

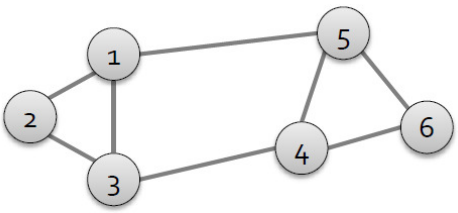


	1	2	3	4	5	6
1	0	1	1	0	1	0
2	1	0	1	0	0	0
3	1	1	0	1	0	0
4	0	0	1	0	1	1
5	1	0	0	1	0	1
6	0	0	0	1	1	0

Matriz simétrica, con eigenvectors reales y ortogonales.

Métodos espectrales

Matriz de grados D



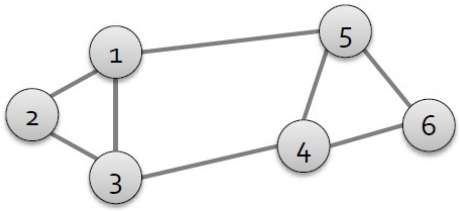
	1	2	3	4	5	6
1	3	0	0	0	0	0
2	0	2	0	0	0	0
3	0	0	3	0	0	0
4	0	0	0	3	0	0
5	0	0	0	0	3	0
6	0	0	0	0	0	2

Matriz diagonal

Métodos espectrales

Matriz laplaciana L = D - A

	1	2	3	4	5	6
1	3	-1	-1	0	-1	0
2	-1	2	-1	0	0	0
3	-1	-1	3	-1	0	0
4	0	0	-1	3	-1	-1
5	-1	0	0	-1	3	-1
6	0	0	0	-1	-1	2



- Eigenvalues: Números reales no negativos.
- Eigenvectors: Reales y ortogonales.

Métodos espectrales

En un grafo conexo…

- **Primer eigenvalue**
  Eigenvector trivial $x_1=(1,..,1)$

$$\lambda_1 = 0$$

- **Segundo eigenvalue**
  (al ser una matriz simétrica)

$$\lambda_2 = \min_x \frac{x^T M\, x}{x^T x}$$

$$x^T L\, x = \sum_{i,j=1}^{n} L_{ij}\, x_i x_j = \sum_{i,j=1}^{n} \left(D_{ij} - A_{ij}\right) x_i x_j$$

$$= \sum_i D_{ii} x_i^2 - \sum_{(i,j)\in E} 2 x_i x_j$$

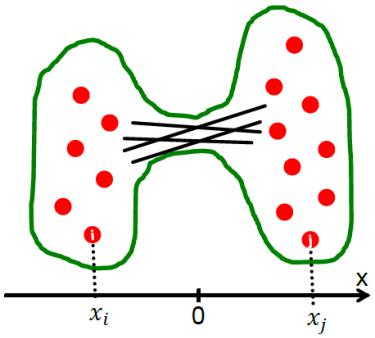
$$= \sum_{(i,j)\in E} (x_i^2 + x_j^2 - 2 x_i x_j) = \sum_{(i,j)\in E} (x_i - x_j)^2$$

Métodos espectrales

¿Qué más sabemos del segundo eigenvector?

- Vector unitario $\sum_i x_i^2 = 1$
- Ortogonal al primer eigenvector $\sum_i x_i \cdot 1 = \sum_i x_i = 0$

$$\lambda_2 = \min_{\substack{\text{All labelings}\\ \text{of nodes } i \text{ so}\\ \text{that } \sum x_i = 0}} \frac{\sum_{(i,j)\in E} (x_i - x_j)^2}{\sum_i x_i^2}$$



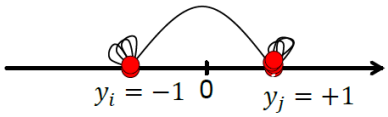
Queremos minimizar, por lo que asignaremos los valores x_i de forma que pocas aristas crucen 0 (queremos que x_i y x_j se compensen)

Métodos espectrales

PROBLEMA DE OPTIMIZACIÓN

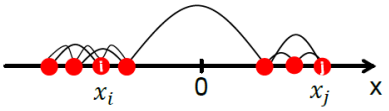
- Partición (A,B) como vector $y_i = \begin{cases} +1 & \text{if } i \in A \\ -1 & \text{if } i \in B \end{cases}$
- Minimización del corte

$$\arg\min_{y\in[-1,+1]^n} f(y) = \sum_{(i,j)\in E} (y_i - y_j)^2$$



- Relajación del problema: Teorema de Rayleigh

$$\min_{y\in\Re^n} f(y) = \sum_{(i,j)\in E} (y_i - y_j)^2 = y^T L y$$



Métodos espectrales

Bisección espectral (EIG1)

- Basado en el **autovector de Fiedler** F (el correspondiente al segundo autovalor más pequeño de la matriz laplaciana).
- Para cada valor x_i correspondiente al nodo n_i, si $x_i > \sigma$ lo asociamos al primer cluster; si no, al segundo.

L. Hagen & A.B. Kahng: "New spectral methods for ratio cut partitioning and clustering". IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 11(9):1074-1085, 1992.

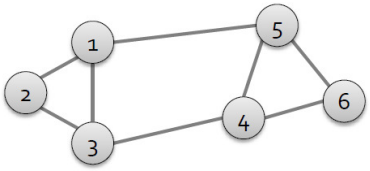
Métodos espectrales

$\lambda =$

0.0
1.0
3.0
3.0
4.0
5.0

$X =$

0.4	0.3	-0.5	-0.2	-0.4	-0.5
0.4	0.6	0.4	-0.4	0.4	0.0
0.4	0.3	0.1	0.6	-0.4	0.5
0.4	-0.3	0.1	0.6	0.4	-0.5
0.4	-0.3	-0.5	-0.2	0.4	0.5
0.4	-0.6	0.4	-0.4	-0.4	0.0



1	0.3
2	0.6
3	0.3
4	-0.3
5	-0.3
6	-0.6

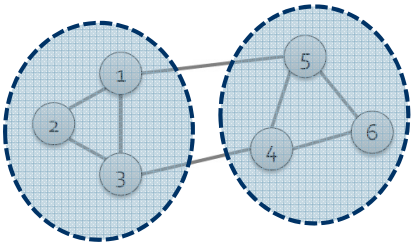
Split at 0:

Cluster A: Positive points

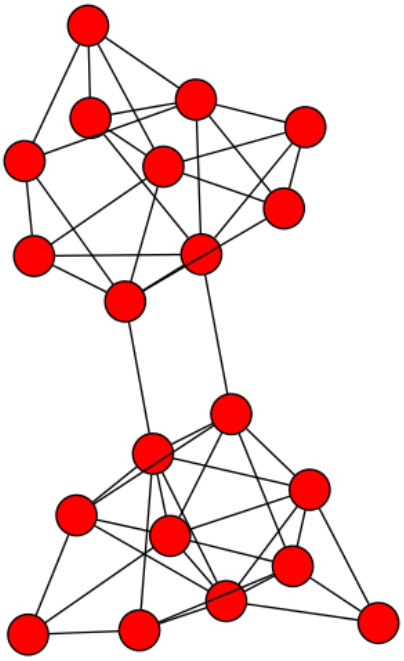
Cluster B: Negative points

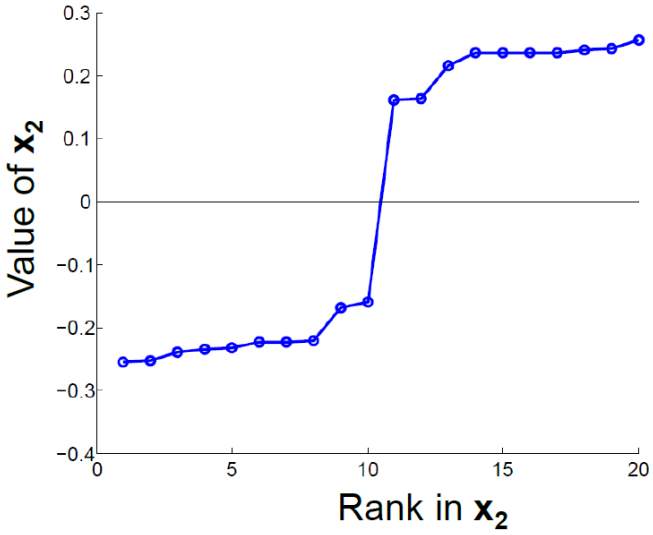
1	0.3
2	0.6
3	0.3

4	-0.3
5	-0.3
6	-0.6



Métodos espectrales

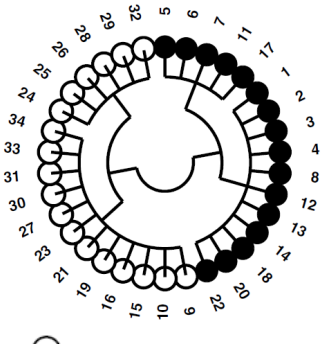


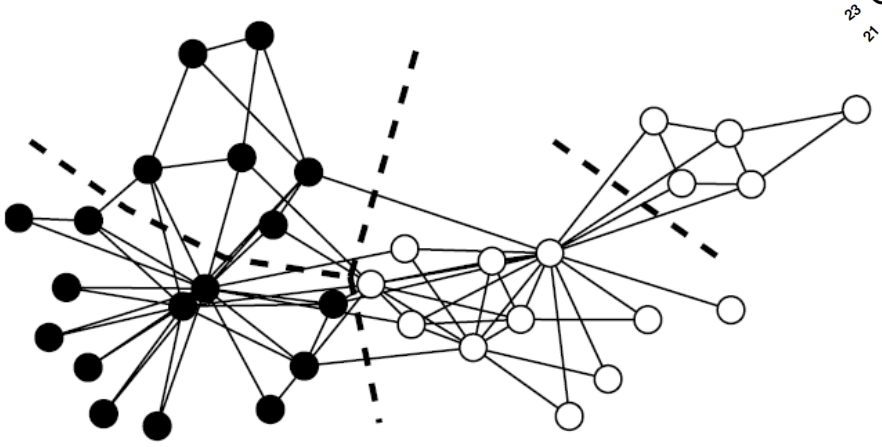


Métodos espectrales

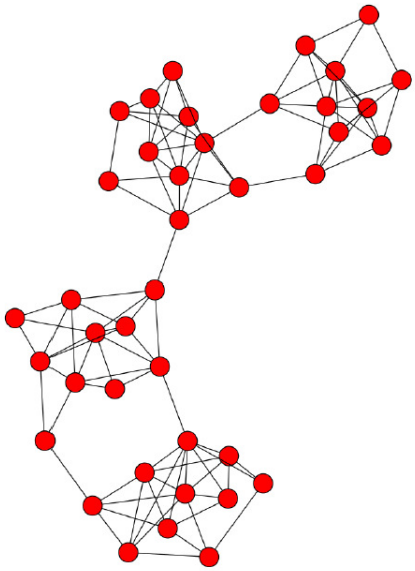
Ejemplo

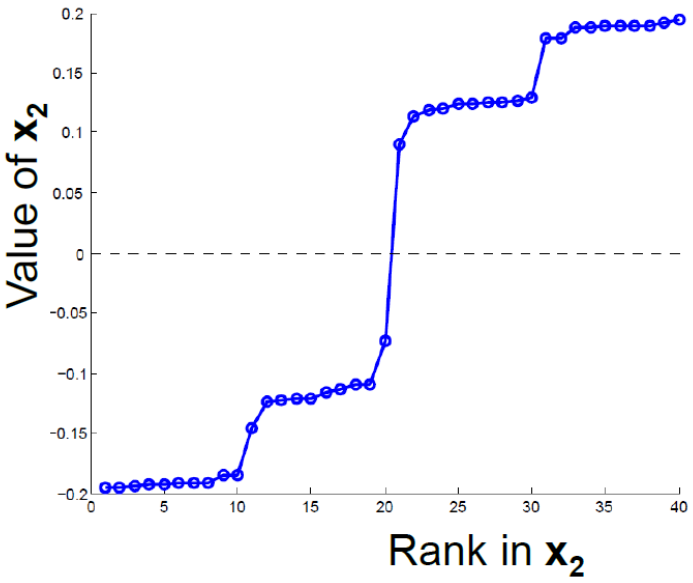
Zachary karate club



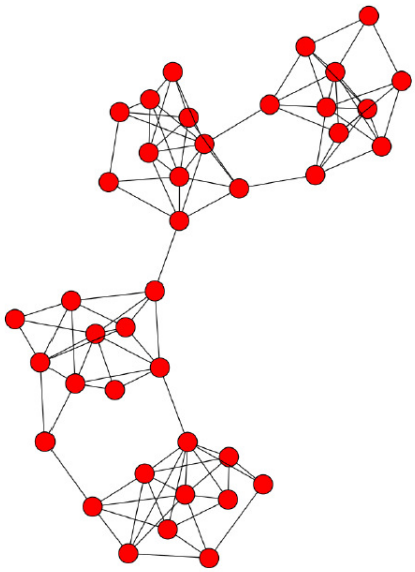


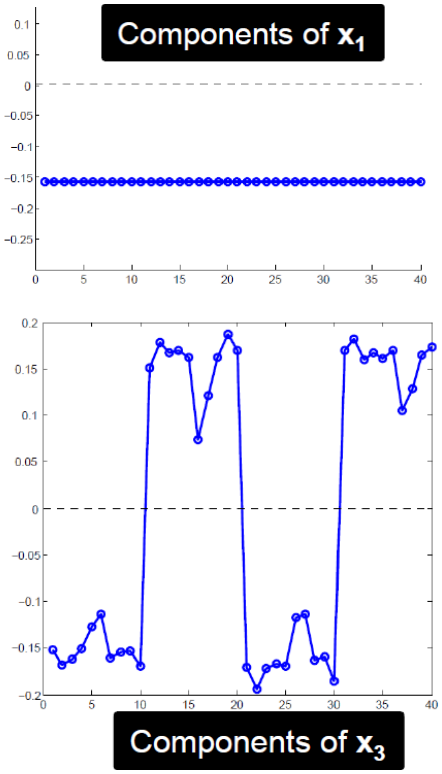
Métodos espectrales





Métodos espectrales





Métodos espectrales

Laplaciano de un grafo

$$L = D - A$$

PROPIEDADES

- El número de autovalores de L iguales a 0 coincide con el número de componentes conexas del grafo.
- Si tenemos k grupos bien definidos en nuestra red, los k primeros autovalores serán cercanos a 0 y sus autovectores asociados nos ayudarán a diferenciar claramente los grupos en un espacio k-dimensional.

Métodos espectrales

Laplaciano de un grafo

Estimación del número de clusters

Si tenemos k particiones bien definidas, los primeros k autovalores de la matriz laplaciana serán cercanos a 0, por lo que es de esperar que el autovalor k + 1 difiera bastante del autovalor k.

NOTA: Para que este método funcione medianamente bien, la red debe tener comunidades claramente definidas.

Métodos espectrales

Laplaciano de un grafo

Normalización

- Normalización simétrica

$$L_{sym} = D^{-1/2} L D^{-1/2}$$

- Normalización asimétrica (por caminos aleatorios)

$$L_{rw} = D^{-1} L$$

Métodos espectrales

IDEA GENERAL

1. Transformar el conjunto de nodos en un conjunto de puntos en un espacio métrico cuyas coordenadas se corresponderán a los k vectores propios más relevantes de la matriz laplaciana del grafo.
2. Agrupar dichos puntos mediante alguna técnica de particionamiento en el espacio métrico.

U. von Luxburg:
"A tutorial on spectral clustering"
Statistics and Computing, 17(4):395-416, 2007.

Métodos espectrales

Algoritmo genérico

1. Calculamos la matriz laplaciana L de nuestra red (normalizada o no).
2. Calculamos los autovalores y autovectores de L.
3. Formamos una matriz U con los k primeros autovectores de L como columnas.
4. Interpretamos las filas de U como vectores de un espacio métrico k-dimensional.
5. Agrupamos los vectores usando cualquier técnica de particionamiento en espacios métricos (p.ej. k-means).

Métodos espectrales

UKMeans

1. Calculamos la matriz laplaciana L sin normalizar.
2. Calculamos autovalores y autovectores de L.
3. Formamos una matriz U con los k primeros autovectores de L como columnas.
4. Interpretamos las filas de U como vectores en un espacio métrico k-dimensional.
5. Agrupamos los vectores usando K-Means.

Métodos espectrales

Algoritmo NJW (a.k.a. KNSC1)

1. Calculamos la matriz laplaciana normalizada simétrica.
2. Calculamos autovalores y autovectores de L_{sym}.
3. Formamos una matriz U con los k primeros autovectores de L como columnas.
4. Realizamos una nueva normalización U' de U.
5. Interpretamos las filas de U' como vectores en un espacio métrico k-dimensional.
6. Agrupamos los vectores de U' usando K-Means.

A.Y. Ng, M.I. Jordan & Y. Weiss: "On spectral clustering: Analysis and an algorithm". Advances in Neural Information Processing Systems, 2:849-856, 2002.

Detección de comunidades

Limitaciones de los métodos descritos

- **Escalabilidad**: Identificación de grandes comunidades.
- Existencia de **solapamiento** entre comunidades.
- Modelos **poco realistas**
  (los algoritmos realizan suposiciones demasiado simplificadas sobre las comunidades de una red, por lo que no funcionan bien con conjuntos de datos reales).
- Técnicas heurísticas **sin garantías**
  (incluso para los algoritmos que funcionan bien en la práctica, no existen garantías sobre la calidad de sus resultados).

Evaluación de resultados

Métricas de evaluación no supervisada

Evaluación global

- **Cohesión**

$$\text{cohesión}(C_i) = \sum_{u,v \in C_i} \text{proximidad}(u, v)$$

- **Separación**

$$\text{separación}(C_i, C_j) = \sum_{\substack{u \in C_i \\ v \in C_j}} \text{proximidad}(u, v)$$

Evaluación de resultados

Métricas de evaluación no supervisada

Evaluación individual de nodos y clusters

- **Coeficiente de silueta**

$$s(v_i) = \frac{b(v_i) - a(v_i)}{\max(a(v_i), b(v_i))}$$

a(v)
Distancia media del nodo
a los demás nodos de su cluster.

b(v)
Distancia mínima entre el nodo
y un cluster al que no pertenece.

$$s(C_j) = \frac{1}{m} \sum_{i=1}^{m} s(v_i)$$

$$s(G) = \frac{1}{c} \sum_{j=1}^{c} s(C_j)$$

Evaluación de resultados

Métricas de evaluación no supervisada

Evaluación individual de nodos y clusters

- **Conductancia**

$$\varphi(C_i) = \frac{\text{cut}(C_i)}{\min\left(\text{vol}(C_i), \text{vol}(\bar{C}_i)\right)}$$

$$\varphi(G) = \min\left(\varphi(C_i)\right), C_i \subseteq V$$

  - **… intra-cluster**

$$\alpha(C) = \min \varphi(G[C_i]), \quad i \in \{1, \dots, k\}$$

  - **… inter-cluster**

$$\sigma(C) = 1 - \max \varphi(C_i), \quad i \in \{1, \dots, k\}$$

Evaluación de resultados

Métricas de evaluación no supervisada

Evaluación individual de nodos y clusters

- **Cobertura**

$$cov(C_i) = \frac{w(C_i)}{w(G)}$$

- **Rendimiento**

$$perf(C) = 1 - \frac{2m\left(1 - 2cov(C)\right) + \sum_{i=1}^{k} |C_i|(|C_i| - 1)}{n(n-1)}$$

Evaluación de resultados

Modularidad Q

- Métrica de evaluación no supervisada que compara los enlaces internos de una comunidad frente a los enlaces que connectan la comunidad con el resto de la red.

$$Q = \frac{1}{2m} \sum_{vw} \left[A_{vw} - \frac{k_v k_w}{2m} \right] \delta(c_v, c_w)$$

Vértices en la misma comunidad

Matriz de adyacencia

Probabilidad de un enlace entre dos vértices (propocional a sus grados)

NOTA: En una red completamente aleatoria, Q=0

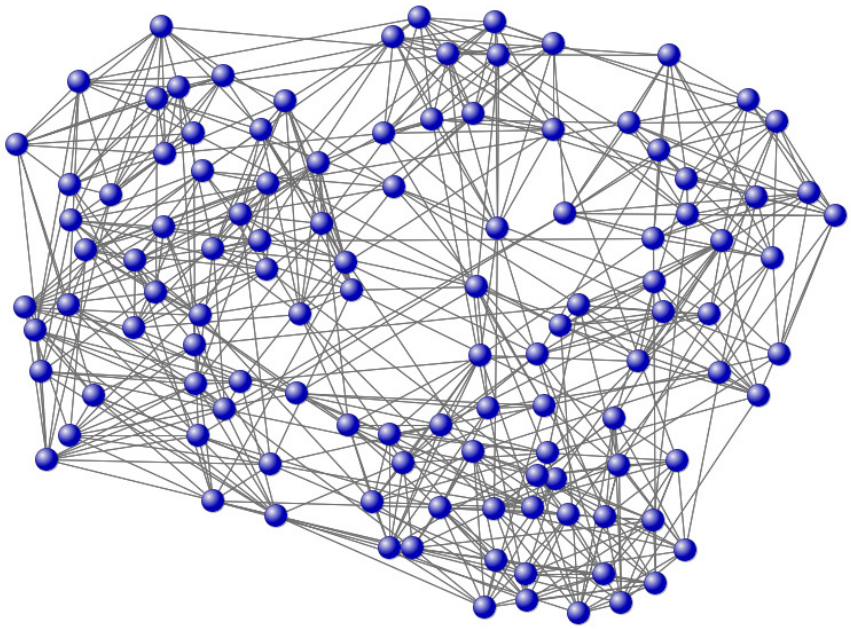
Evaluación de resultados

Métricas de evaluación no supervisada

Enfoque	Medida	Ref.	Rango	Características
Análisis Global	Cohesión	[79]	$[0, \infty]$	Mide las distancias entre nodos dentro de un cluster, se buscan valores pequeños, varía dependiendo de la medida de proximidad.
	Separación	[79]	$[0, \infty]$	Mide las distancias de los nodos del *cluster* con respecto a aquellos que no pertenece, se busca el máximo posible, varía dependiendo de la medida de proximidad.
Análisis individual	Coeficiente de silueta	[68]	$[-1,1]$	Adecuada para comunidades altamente conectadas. Alta complejidad y fallos con nodos hoja.
	Conductancia	[32]	[0,1]	Medición de cuellos de botella, adecuado para *clusters* de gran tamaño; fallos en la evaluación de *clusters* con pocos nodos, pequeños y/o muy grandes.
	Cobertura	[5]	[0,1]	Peso del *cluster*, basado en los cortes mínimos; fallos en la evaluación de *clusters* con pocos nodos, pequeños y/o muy grandes.
	Rendimiento	[5]	[0,1]	Número de nodos adyacentes, densidad; falla en redes de gran tamaño con numero alto de *clusters*.
	Coeficiente de agrupamiento	[64]	[0,1]	Búsqueda de estructuras conexas (triángulos).
	Modularidad	[52]	[0,1]	Comparación del cluster con estructura aleatoria.

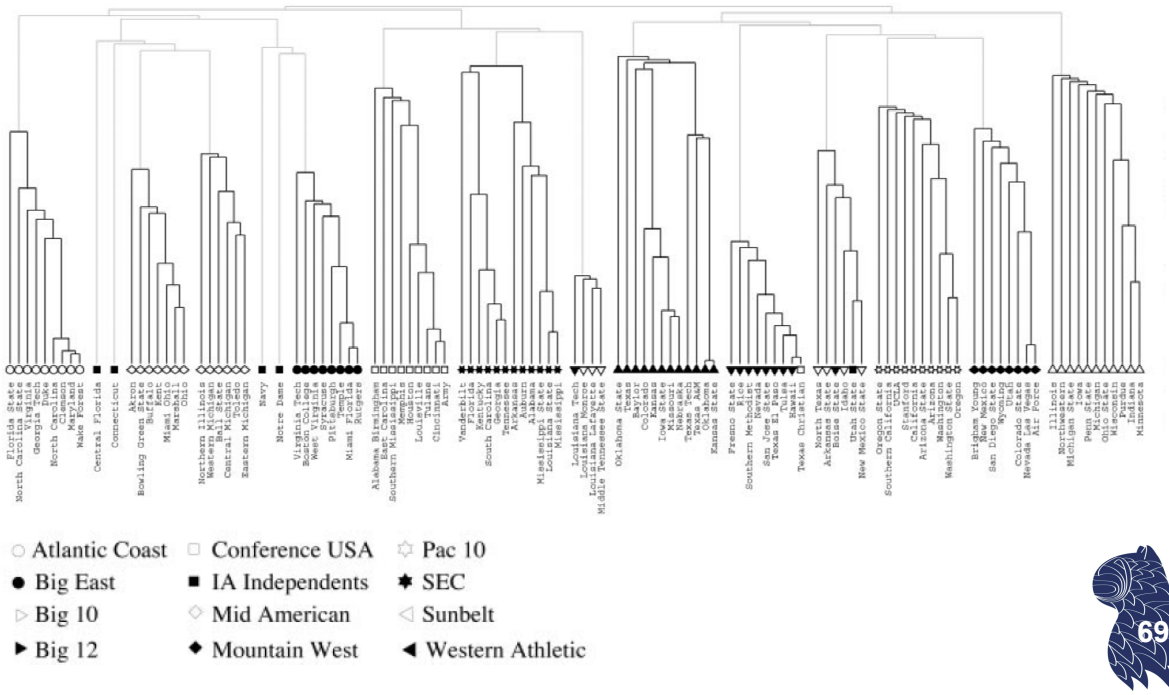
Evaluación de resultados

Red de fútbol americano (115 nodos, 613 enlaces)



Evaluación de resultados

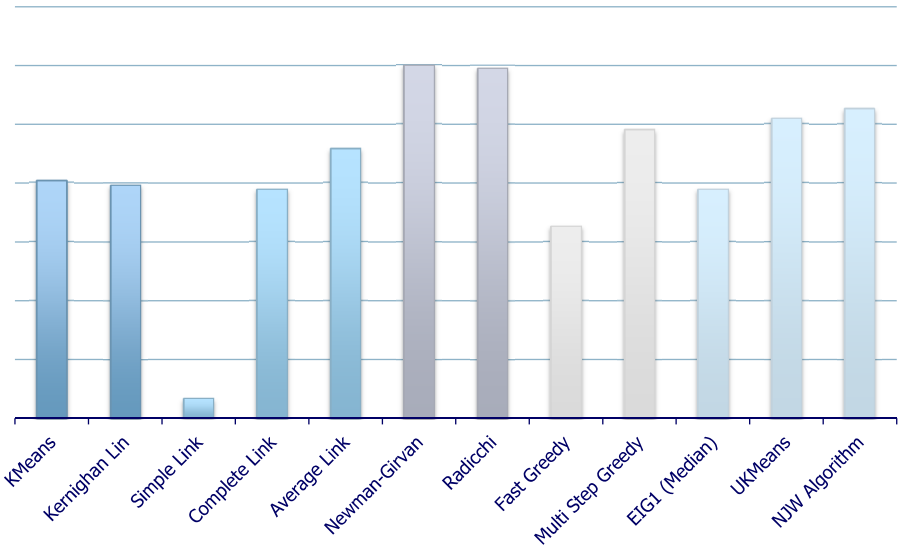
Red de fútbol americano (115 nodos, 613 enlaces)



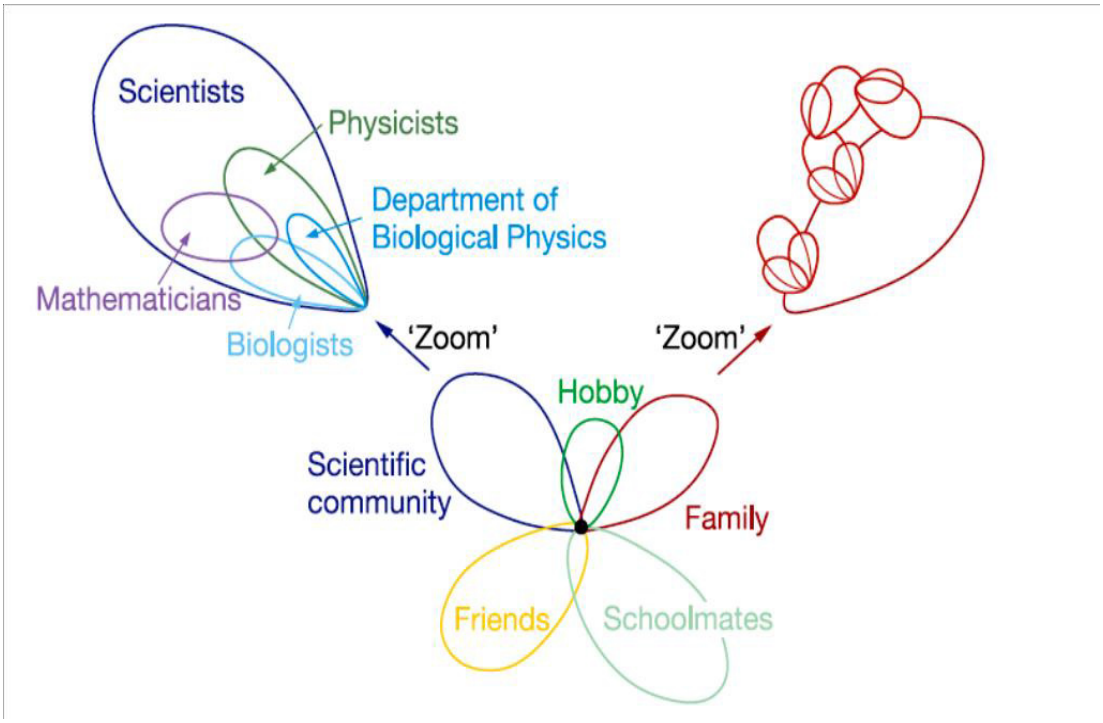
Evaluación de resultados

Red fútbol americano (115 nodos, 613 enlaces)

Modularidad

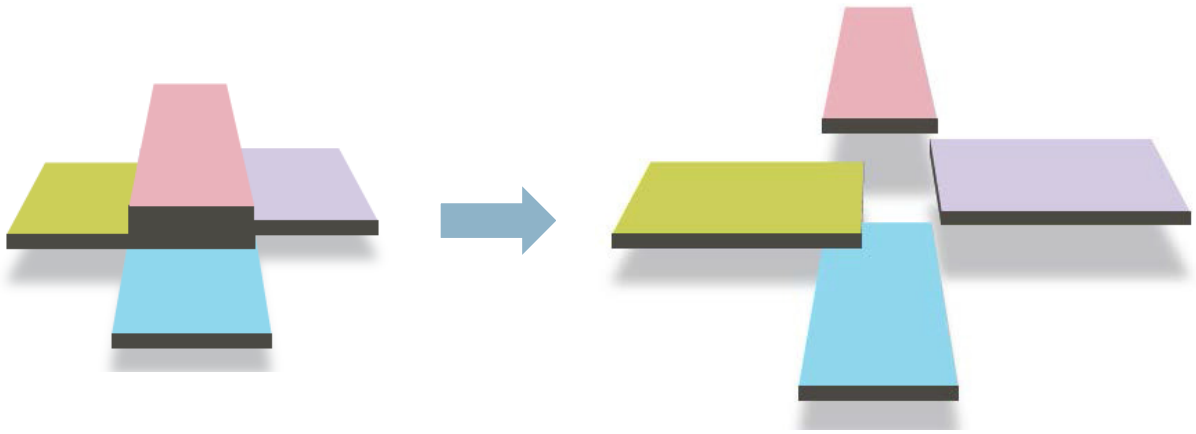


Comunidades solapadas



Comunidades solapadas

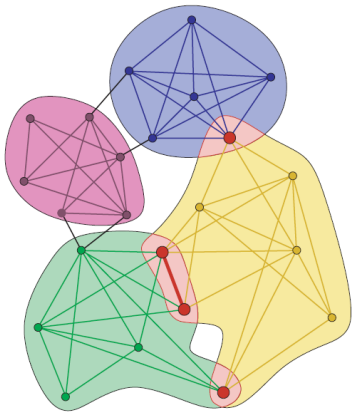
Comunidades en una red real



Leskovec, Rajamaran & Ullman:
"Mining of Massive Datasets"
Stanford University

CPM

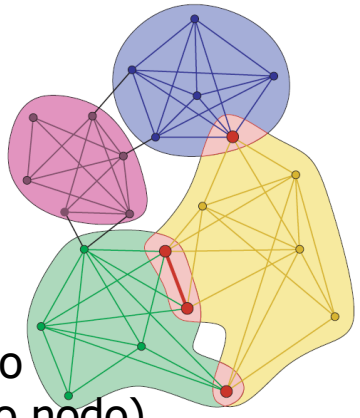
Clique Percolation Method
[Palla et al., Nature'2005]



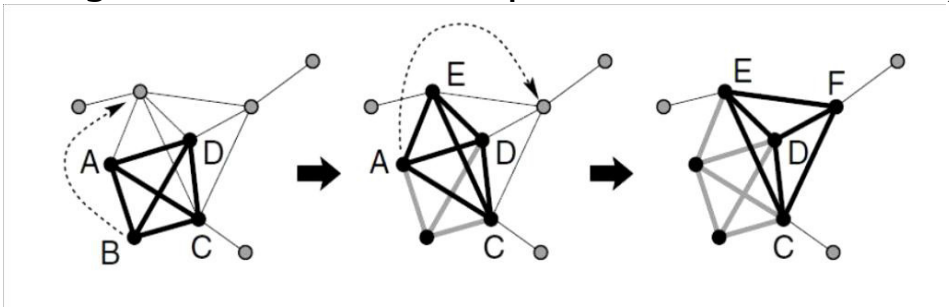
- Si de un k-clique eliminamos un enlace, se obtienen dos (k-1)-cliques solapados que comparten k-2 nodos.
- La unión de estos conjuntos de nodos solapados forma una cadena de cliques.
- IDEA (similar a Radicchi): Las aristas existentes dentro de una comunidad tienden a formar cliques; las aristas que conectan nodos de distintas comunidades, no.

CPM

Clique Percolation Method
[Palla et al., Nature'2005]

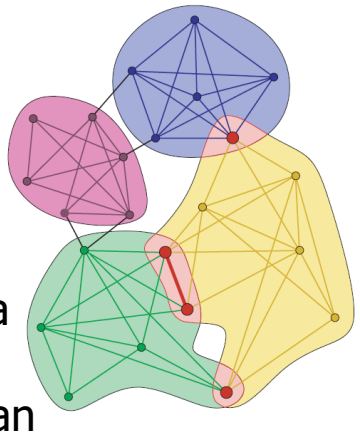


- ALGORITMO
  Encontrar cliques adyacentes
  para formar una cadena de cliques
  (es posible rotar/pivotar los k-cliques a lo largo de la cadena reemplazando un solo nodo).



CPM

Clique Percolation Method
[Palla et al., Nature'2005]



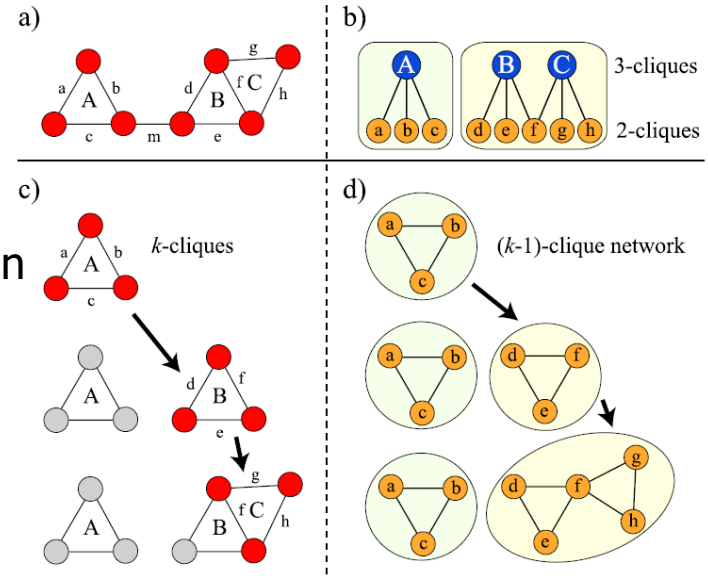
- IMPLEMENTACIÓN
  Matriz de adyacencia de k-cliques
  (número de nodos compartidos por cada par de cliques) filtrada (a 0 para valores <= k-1), a partir de la cual se determinan fácilmente las comunidades solapadas (conectividad).

- CPM es de orden exponencial (búsqueda de cliques), si bien CFinder (http://www.cfinder.org/) ofrece una versión aproximada más eficiente, $O(n_{cliques}^2)$

CPM Secuencial

[Kumpula et al., Physical Review E 2008]

- IDEA:
  Comprobar la formación de k-cliques conforme se añaden aristas.
- Algoritmo escalable, prácticamente lineal.

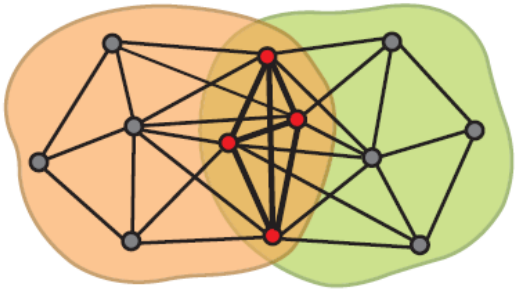


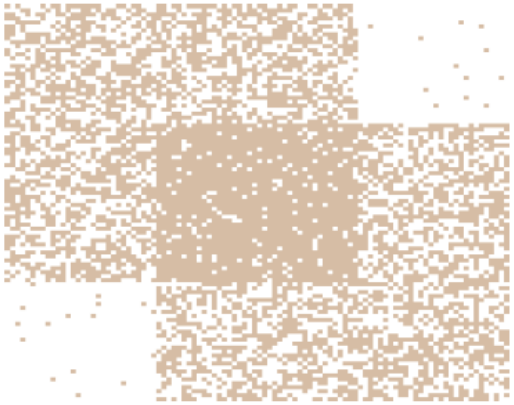
Alternativas

- CPMw para redes con pesos [Farkas, NJP'2007]
- "Maximal cliques" como núcleos de comunidades que luego se fusionan. [Shang et al., CPL'2010]
- MOSES basado en modelos estadísticos [McDaid & Hurley, ASONAM'2010]
- Algoritmo de fuzzy clustering basado en una relación difusa [Sun et al., Information Sciences 2011]
- CONA en dos etapas: primero se buscan comunidades no solapadas y luego se buscan vínculos entre ellas [Wu et al., Physica A 2012]

BigCLAM

IDEA:
La densidad de las aristas en las zonas solapadas es mayor…



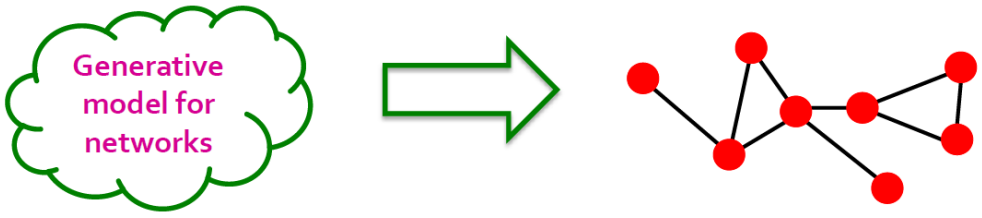


Yang & Leskovec: "Overlapping Community Detection at Scale: A Nonnegative Matrix Factorization Approach". ACM International Conference on Web Search and Data Mining (WSDM), 2013.

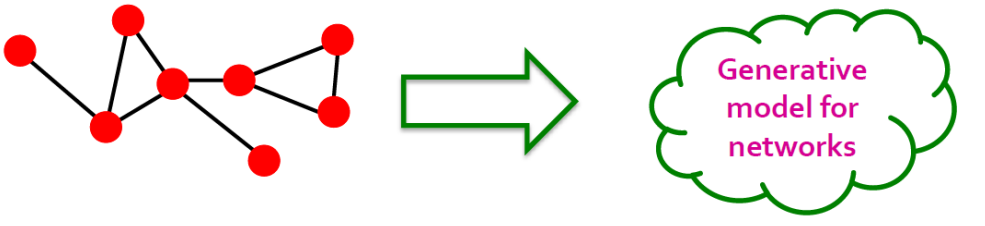
BigCLAM

PLAN

- Dado un modelo, podemos generar una red

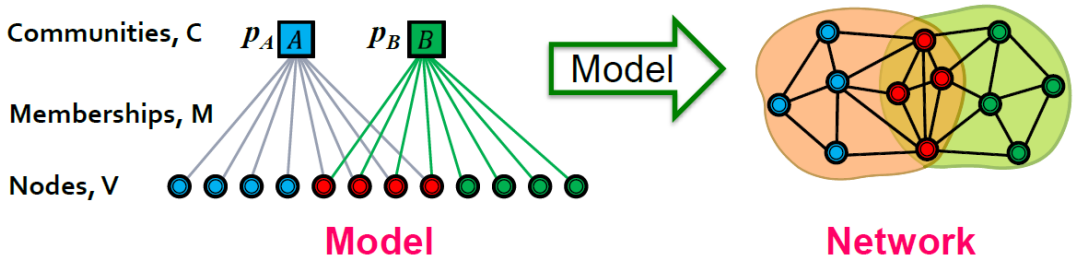


- Dada una red, podemos encontrar el "mejor" modelo



BigCLAM

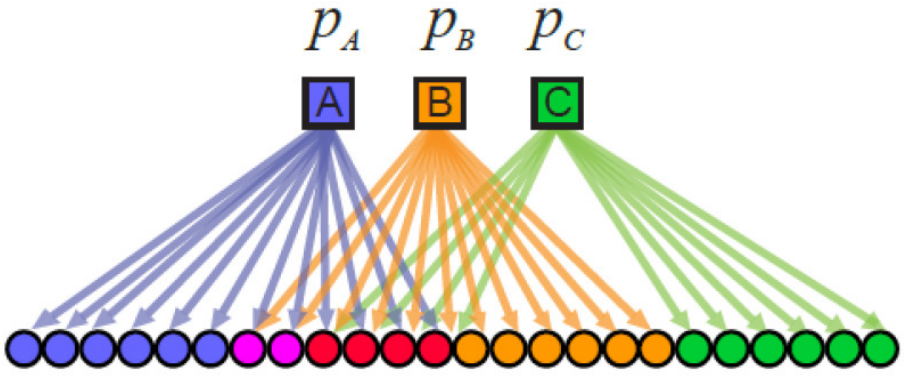
AGM [Affiliation Graph Model]



- Generación de enlaces: Para cada par de nodos de una misma comunidad A, creamos un enlace entre ellos con probabilidad p_A.
- Modelo cuyos parámetros estimaremos para detectar las comunidades.

BigCLAM

AGM [Affiliation Graph Model]

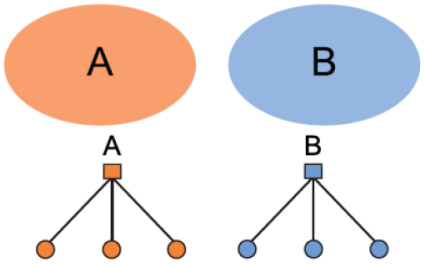


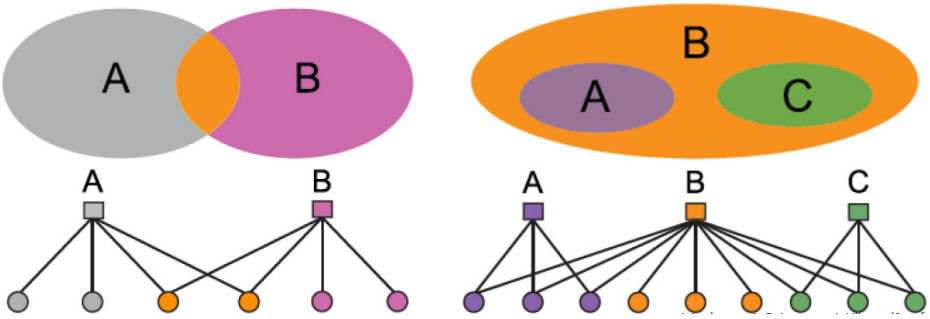
$$P(u,v) = 1 - \prod_{c \in M_u \cap M_v} (1 - p_c)$$

If $\boldsymbol{u}, \boldsymbol{v}$ share no communities: $\boldsymbol{P(u,v) = \varepsilon}$

BigCLAM

AGM [Affiliation Graph Model]

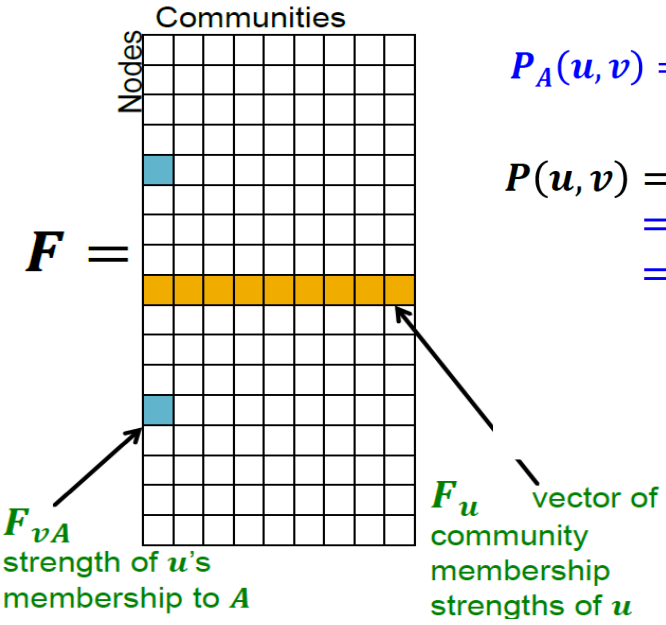




Modelo versátil:
Comunidades no solapadas, solapadas y anidadas

BigCLAM

- Parámetros del modelo F_{nA}: Fuerza con la que un nodo n pertenece a una comunidad A.



$$P_A(u,v) = 1 - \exp(-F_{uA} \cdot F_{vA})$$

$$P(u,v) = 1 - \prod_C (1 - P_C(u,v)) = 1 - \exp(-\sum_C F_{uC} \cdot F_{vC}) = 1 - \exp(-F_u \cdot F_v^T)$$

BigCLAM

PROBLEMA

Dada una red, estimar F

- Maximizar likelihood P(G|F)

$$\boldsymbol{arg\,max_F} \prod_{\boldsymbol{(u,v)\in E}} \boldsymbol{p(u,v)} \prod_{\boldsymbol{(u,v)\notin E}} \boldsymbol{(1-p(u,v))}$$

- Log-likelihood l(F) = log P(G|F)

$$l(F) = \sum_{(u,v)\in E} \log(1-\exp(-F_u F_v^T)) - \sum_{(u,v)\notin E} F_u F_v^T$$

BigCLAM 1.0

ALGORITMO

- Fila de F

$$l(F_u) = \sum_{v\in\mathcal{N}(u)} \log(1-\exp(-F_u F_v^T)) - \sum_{v\notin\mathcal{N}(u)} F_u F_v^T$$

- Gradiente de una fila de F

$$\nabla l(F_u) = \sum_{v\in\mathcal{N}(u)} F_v \frac{\exp(-F_u F_v^T)}{1-\exp(-F_u F_v^T)} - \sum_{v\notin\mathcal{N}(u)} F_v$$

- Maximización: **Gradiente ascendente coordinado**
  Algoritmo iterativo

$$\boldsymbol{F_u \leftarrow F_u + \eta\,\nabla l(F_u)}$$

BigCLAM 2.0

- Problema:
  Calcular el gradiente $\nabla l(F_u)$ requiere tiempo lineal con respecto al tamaño de la red.

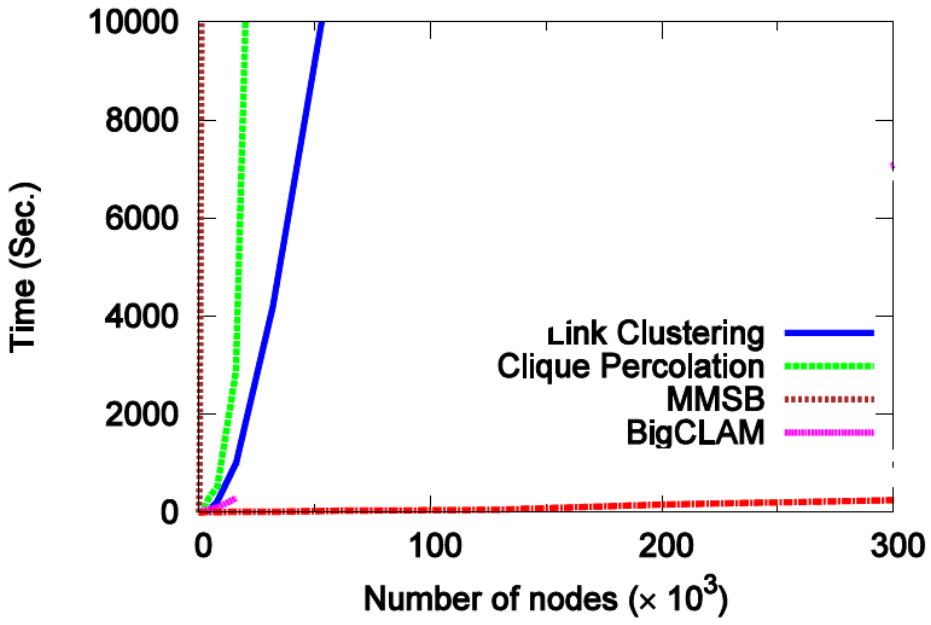
- Solución:

$$\sum_{v \notin \mathcal{N}(u)} F_v = (\sum_{v} F_v - F_u - \sum_{v \in \mathcal{N}(u)} F_v)$$

  Podemos precalcular $\Sigma_v F_v$ para obtener en tiempo lineal con respecto al grado de los nodos |N(u)|

BigCLAM

Resultado: Algoritmo escalable



Aplicaciones

Blogs políticos

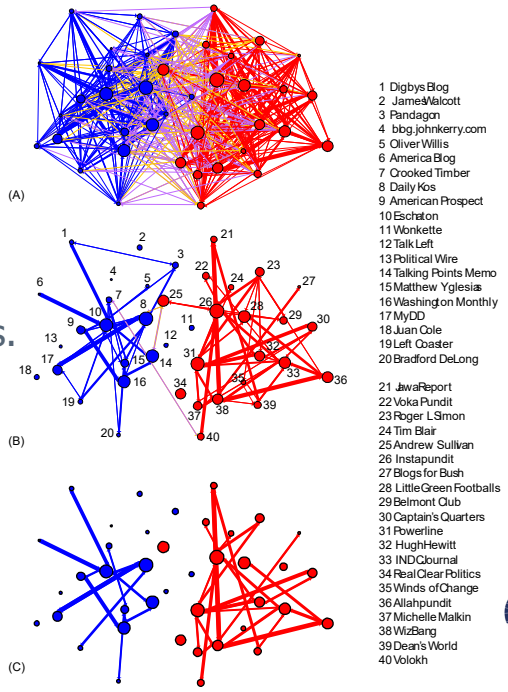
A) All citations between blogs.

B) Blogs with at least 5 citations in both directions.

C) Edges further limited to those exceeding 25 combined citations.

only 15% of the citations bridge communities

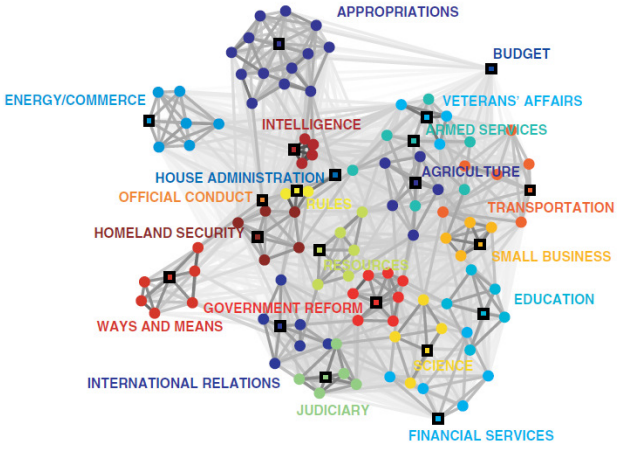
L. A. Adamic & N. Glance: **The political blogosphere and the 2004 US Election** LinkKDD'2005

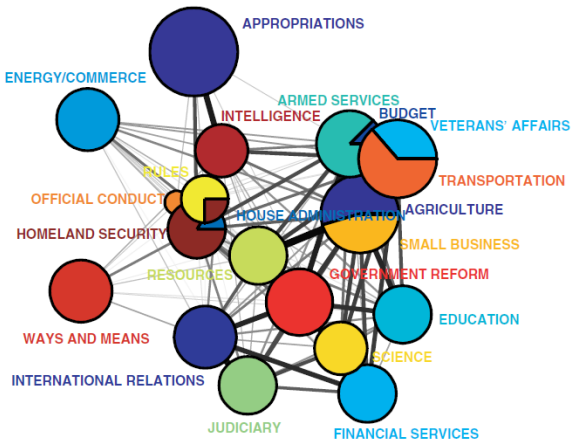


Aplicaciones

Comités y subcomités

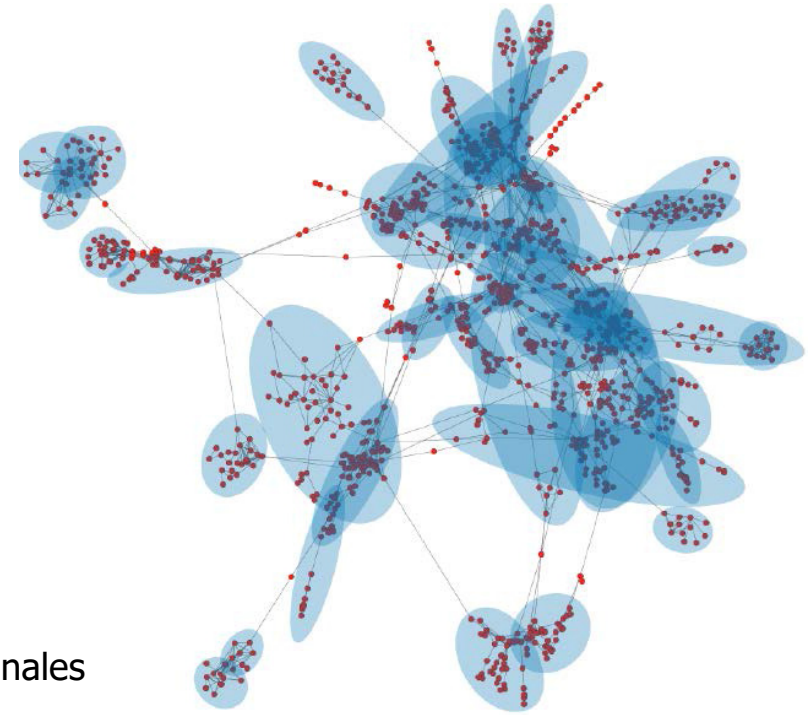
U.S. House of Representatives 2003-2004





Aplicaciones

Redes de interacción de proteínas (PPI)



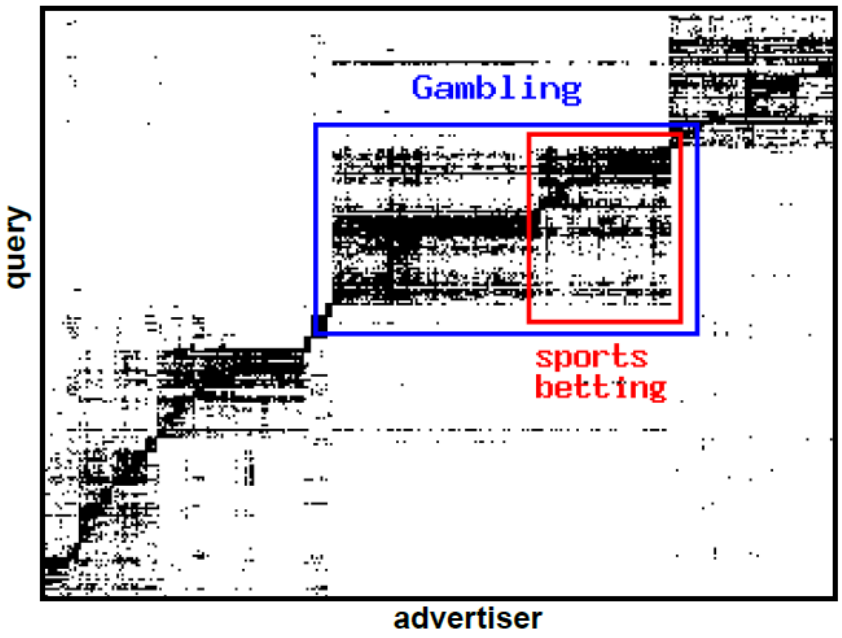
Nodos
Proteínas

Enlaces
Interacciones

Comunidades
Módulos funcionales

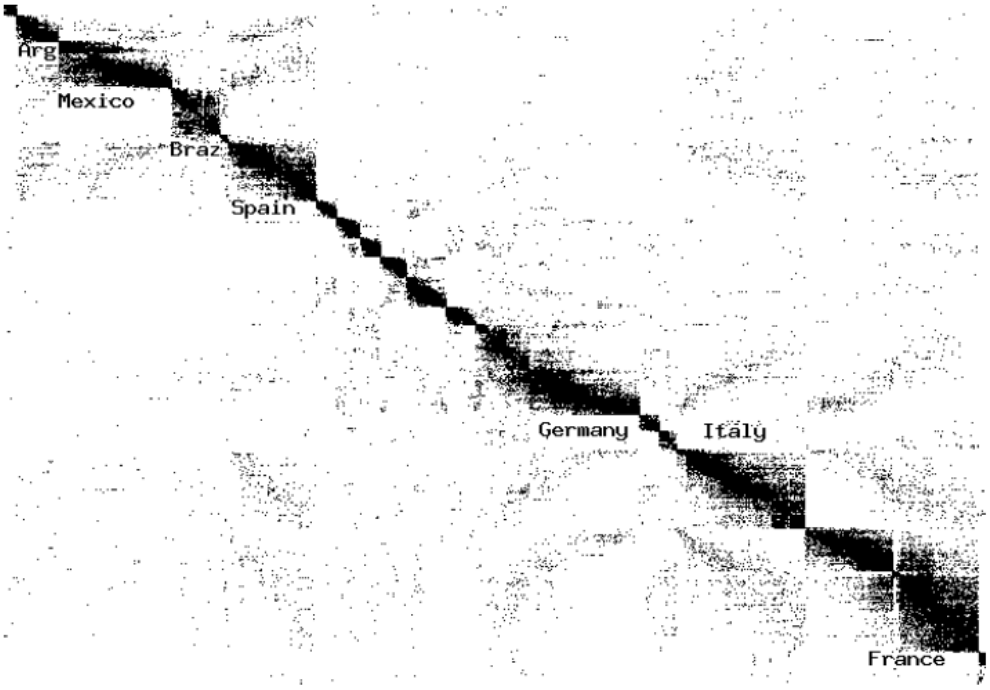
Aplicaciones

Query-advertiser graphs (micromarkets)



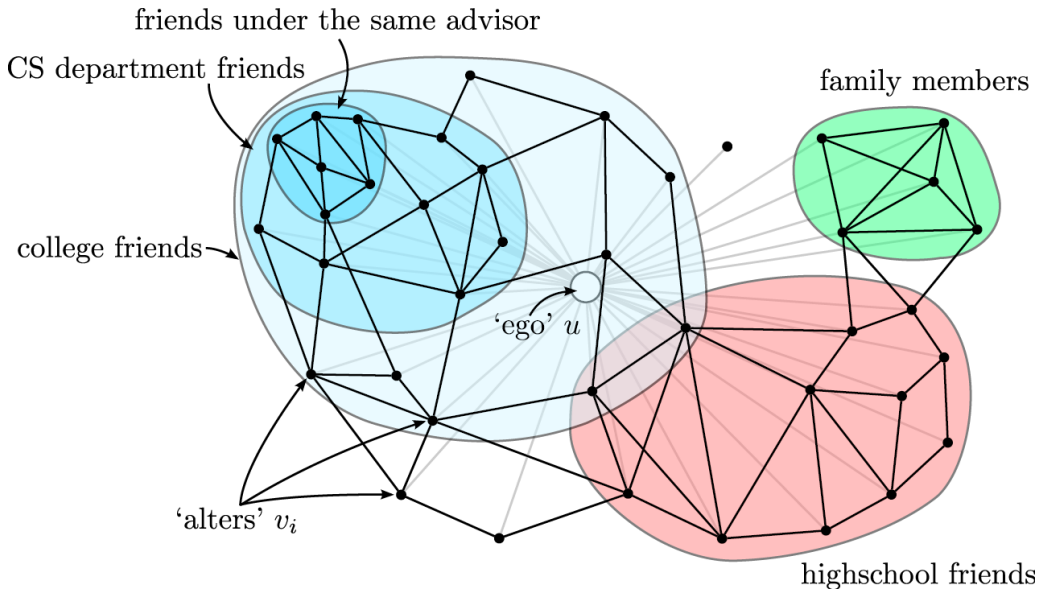
Aplicaciones

Movie-actor network



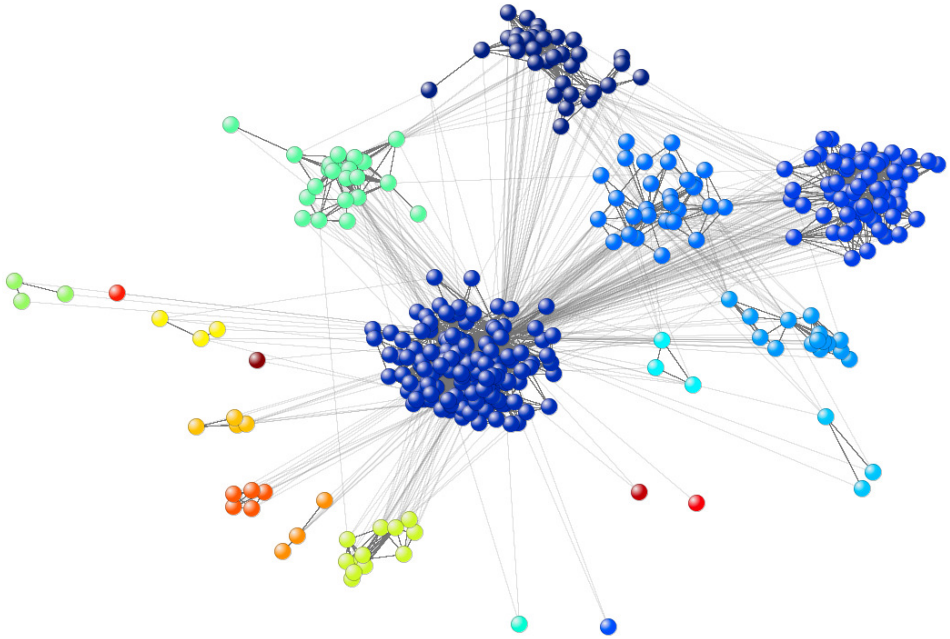
Aplicaciones

Círculos sociales



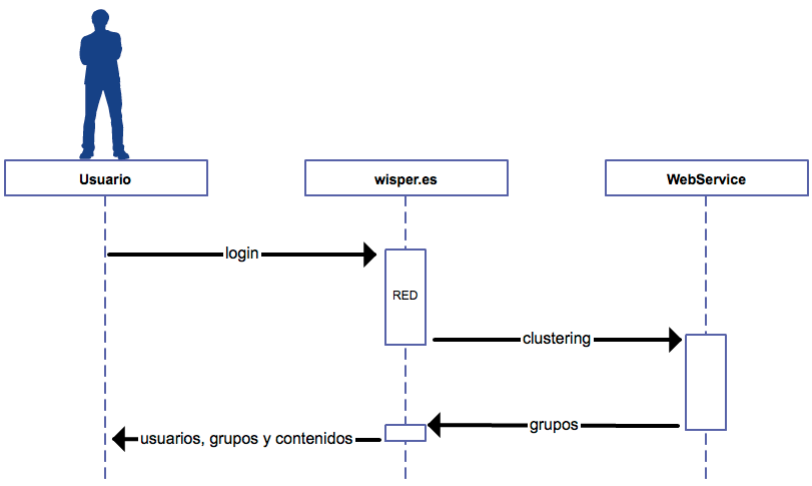
Aplicaciones

Comunidades en una red de amigos en Facebook



Aplicaciones

wisper.es

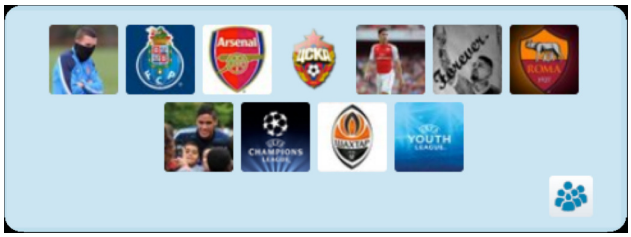


Aplicaciones

wisper.es

Aplicaciones

wisper.es



NOESIS

Network-Oriented Exploration,
Simulation, and Induction System

http://noesis.ikor.org

Agradecimientos

Julio Omar Palacio Niño

Detección de comunidades en redes:
Algoritmos y aplicaciones

MSc Thesis, September 2013

Department of Computer Science and Artificial Intelligence
University of Granada (Spain)

Aarón Rosas Rodríguez & Francisco Javier Gijón Moleón

Algoritmos paralelos
para la detección de comunidades en redes

Proyecto de Fin de Carrera, septiembre de 2014

ETSIIT, Universidad de Granada

Bibliografía

Detección de comunidades

- Michelle Girvan & Mark E.J. Newman: **Community structure in social and biological networks**, PNAS 99(12):7821–7826 (2002)
- Aaron Clauset, Mark E. J. Newman & Cristopher Moore: **Finding community structure in very large networks**, Physical Review E 70(6):066111 (2004)
- Jure Leskovec, Kevin J. Lang, Anirban Dasgupta & Michael W. Mahoney: **Statistical Properties of Community Structure in Large Social and Information Networks**, International World Wide Web Conference, WWW'08 (2008). Extended version: **Community structure in large networks: Natural cluster sizes and the absence of large well-defined clusters**, arxiv:0810.1355 (2008)
- Martin Rosvall & Carl T. Bergstrom: **Maps of random walks on complex networks reveal community structure**, PNAS 105(4):1118-1123 (2008)
- Jure Leskovec, Kevin J. Lang & Michael W. Mahoney: **Empirical Comparison of Algorithms for Network Community Detection,** WWW 2010 (2010)
- Santo Fortunato: **Community detection in graphs**. Physics Reports, 486(3-5):75-174, (2010).
- S. Arora, R. Ge, S. Sachdeva & G. Schoenebeck: **Finding overlapping communities in social networks: toward a rigorous approach**. Proceedings of the 13th ACM Conference on Electronic Commerce, EC '12, pp. 37-54 (2012)
- Jierui Xie, Stephen Kelley & Boleslaw K. Szymanski: **Overlapping community detection in networks: The state-of-the-art and comparative study**. ACM Computing Surveys 45:4, Article 43, August 2013. DOI 10.1145/2501654.2501657

Bibliografía

Clique percolation

- Imre Derényi, Gergely Palla & Tamás Vicsek: **Clique percolation in random networks**. Physical Review Letters, 94:160202, 2005.
- Gergely Palla, Imre Derényi, Illés Farkas & Tamás Vicsek: **Uncovering the overlapping community structure of complex networks in nature and society**, Nature 435(7043):814-818, 2005.
- Balzs Adamcsek, Gergely Palla, Ills J. Farkas, Imre Derényi & Tamás Vicsek. **CFinder: locating cliques and overlapping modules in biological networks**. Bioinformatics, 22(8):1021–1023, 2006.
- Jussi M. Kumpula, Mikko Kivelä, Kimmo Kaski & Jari Saramäki: **Sequential algorithm for fast clique percolation**. Phys. Rev. E, 78:026109, 2008.
- Fergal Reid, Aaron F. McDaid & Neil J. Hurley: **Percolation computation in complex networks**. CoRR, abs/1205.0038, 2012.
- Illés Farkas, Dániel Ábel, Gergely Palla & Tamás Vicsek: **Weighted network modules**, New Journal of Physics, Volume 9, June 2007.
- Ming-Sheng, Shang; Duan-Bing, Chen; Tao, Zhou. **Detecting Overlapping Communities Based on Community Cores in Complex Networks**. Chinese Physics Letters, 27(5):58901-58904, May 2010.
- Aaron F. McDaid & Neil J. Hurley: **Detecting Highly Overlapping Communities with Model-Based Overlapping Seed Expansion**. ASONAM'2010, 112-119, 2010.

Bibliografía

Aplicaciones

e.g. Protein complexes in PPI Networks

- Gary D. Bader & Christopher W.V. Hogue: **An automated method for finding molecular complexes in large protein interaction networks**. BMC Bioinformatics, 4(1):1–27, 2003.
- Md Altaf-Ul-Amin, Yoko Shinbo, Kenji Mihara, Ken Kurokawa & Shigehiko Kanaya: **Development and implementation of an algorithm for detection of protein complexes in large interaction networks**. BMC Bioinformatics, 7(1):113,2006.
- Min Li, Jian-er Chen, Jian-xin Wang, Bin Hu & Gang Chen: **Modifying the dpclus algorithm for identifying protein complexes based on new topological structures**. BMC Bioinformatics, 9(1):1–16, 2008
- Min Wu, Xiaoli Li, Chee-Keong Kwoh & See-Kiong Ng: **A core-attachment based method to detect protein complexes in ppi networks**. BMC Bioinformatics, 10(1):1–16, 2009.

Bibliografía

Redes: Orígenes & Aplicaciones (redes sociales, WWW…)

- Stanley Milgram: **The small world problem**.
  Psychology Today, 2:60-67 (1967)
- Phillip W. Anderson: **More is different**.
  Science, 177:393-396 (1972)
- Mark S. Granovetter: **The strength of weak ties**.
  American Journal of Sociology, 78:1360-1380 (1973)
- Stanley Wasserman & Katherine Faust: **Social Network Analysis: Methods and Applications**. Cambridge University Press, 1994
- John P. Scott: **Social Network Analysis**, 2nd edition.
  Sage Publications Ltd., 2000.
- Andrei Broder, Ravi Kumar, Farzin Maghoul, Prabhakar Raghavan, Sridhar Rajagopalan, Raymie Stata, Andrew Tomkins & Janet Wiener: **Graph structure in the Web**. Computer Networks 33:309–320 (2000)
- Steven H. Strogatz: **Exploring Complex Networks**.
  Nature, 410:268-275 (2001)
- Albert-Laszlo Barabasi: **Linked: How Everything Is Connected to Everything Else and What It Means**. Plume, 2003. ISBN 0452284392
- Duncan J. Watts: **Six Degrees: The Science of a Connected Age**. W. W. Norton & Company, 2004. ISBN 0393325423
- Jure Leskovec, Jon M. Kleinberg & Christos Faloutsos: **Graphs over time: densification laws, shrinking diameters and possible explanations**. KDD'2005

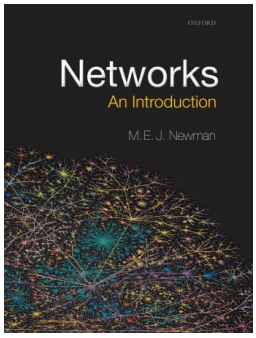
Bibliografía

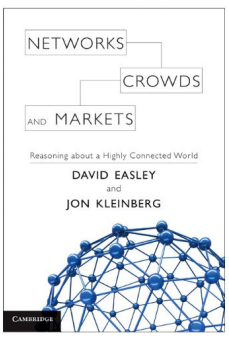
Modelos de redes

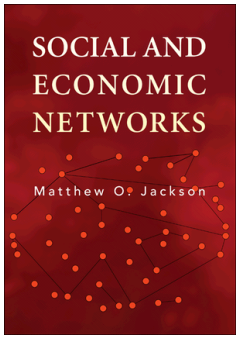
- Paul Erdös & Alfred Rényi: **On the evolution of random graphs**. Mathematical Institute of the Hungarian Academy of Sciences, 5:17-61 (1960) reprinted in Duncan, Barabasi & Watts (eds.): "The Structure and Dynamics of Networks"
- Ray Solomonoff & Anatol Rapoport: **Connectivity of random nets**. Bulletin of Mathematical Biophysics, 13:107-117 (1951) reprinted in Duncan, Barabasi & Watts (eds.): "The Structure and Dynamics of Networks"
- Duncan J. Watts & Steven H. Strogatz: **Collective dynamics of 'small-world' networks**. Nature, 393:440-442 (1998)
- Albert-László Barabási & Réka Albert: **Emergence of scaling in random networks**. Science, 286:509-512 (1999)
- Réka Albert, Hawoong Jeong & Albert-László Barabási: **Error and attack tolerance of complex networks**. Nature 406:378-382 (2000)
- M.E.J. Newman, S.H. Strogatz & D.J. Watts: **Random graphs with arbitrary degree distributions and their applications**. Physical Review E, 64:026118 (2001)
- M.E.J. Newman, S.H. Strogatz & D.J. Watts: **Random graphs models of social networks**. PNAS 99:2566-2572 (2002)
- Erzsébet Ravasz & Albert-László Barabási: **Hierarchical organization in complex networks**. Physical Review E, 67:026112 (2003)
- Mark Newman: **The structure and function of complex networks**. SIAM Review 45:167-256 (2003)

Bibliografía – Libros de texto

- David Easley & Jon Kleinberg: **Networks, Crowds, and Markets: Reasoning About a Highly Connected World**. Cambridge University Press, 2010. ISBN 0521195330 http://www.cs.cornell.edu/home/kleinber/networks-book/
- Mark Newman: **Networks: An Introduction**. Oxford University Press, 2010. ISBN 0-19-920665-1
- Matthew O. Jackson: **Social and Economic Networks**, Princeton University Press, 2008. ISBN 0-691-13440-5







Bibliografía – Libros divulgativos

- Albert-Laszlo Barabási: **Linked: How Everything Is Connected to Everything Else and What It Means**. Plume, 2003. ISBN 0452284392
- Duncan J. Watts: **Six Degrees: The Science of a Connected Age**. W. W. Norton & Company, 2004. ISBN 0393325423
- Albert-Laszlo Barabási: **Bursts: The Hidden Pattern Behind Everything We Do**. Dutton, 2010. ISBN 0525951601