Node
Description

NODE

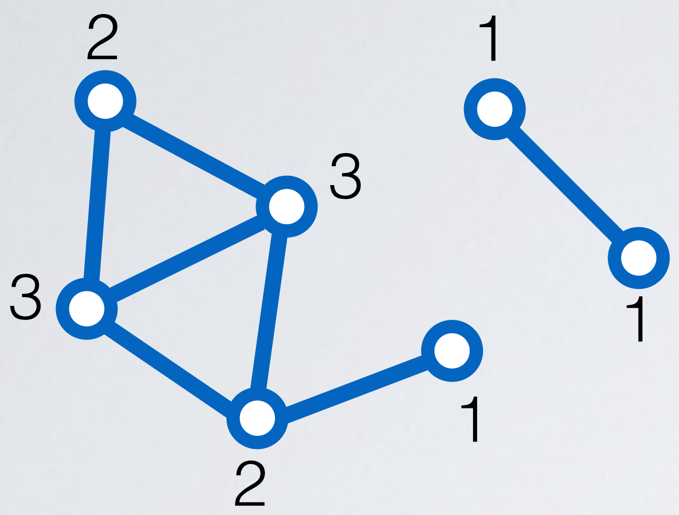
- Node *centrality measures* = being important in the network (not necessarily central in term of being in the center)

- Usage:
  - Discover important nodes
  - Rank nodes by importance
  - +machine learning => classification of nodes

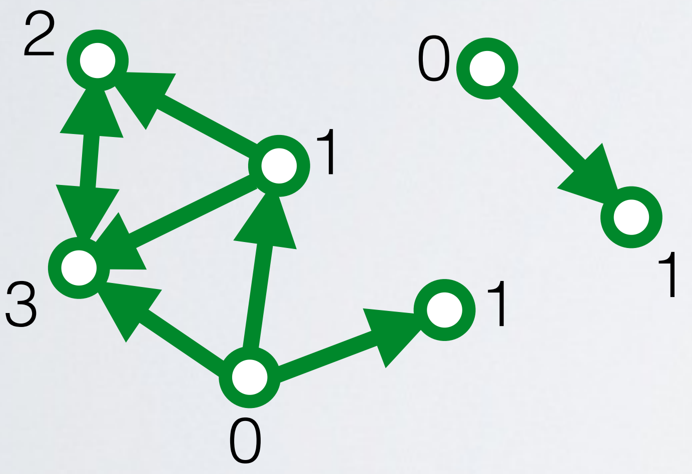
Degree centrality - recap

Number of connections of a node

- Undirected network

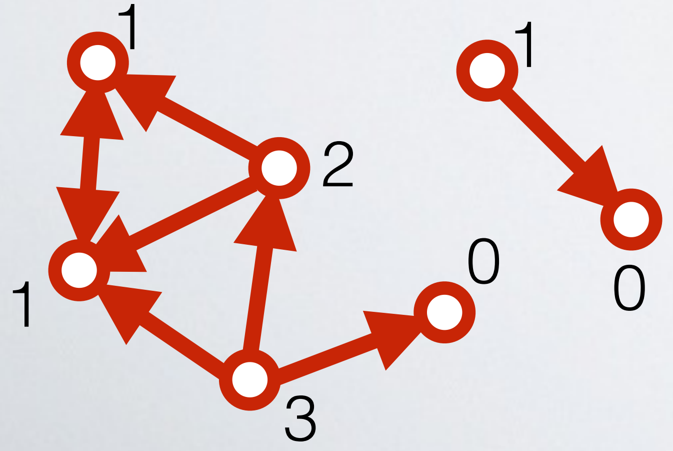


- Directed network



In degree

Out degree

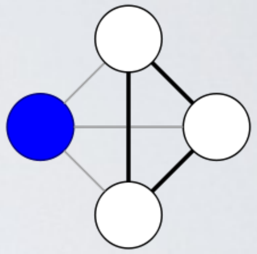


NODE DEGREE

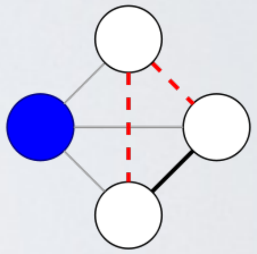
- Often enough to find important nodes
  - Main characters of a series talk with the more people
  - Largest airports have the most connections
  - …
- But not always
  - Facebook users with the most friends are spam
  - Webpages/wikipedia pages with most links are simple lists of references
  - …

NODE CLUSTERING COEFFICIENT

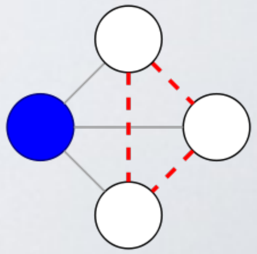
- **Clustering coefficient**: density of neighborhood
- Tells you if the neighbors of the node are connected
- Be careful!
  - Degree 2: value 0 or 1
  - Degree 1000: Not 0 or 1 (usually)
  - Ranking them is not meaningful
- Can be used as a proxy for "communities" belonging:
  - If node belong to single group: high CC
  - If node belong to several groups: lower CC



c = 1



c = 1/3

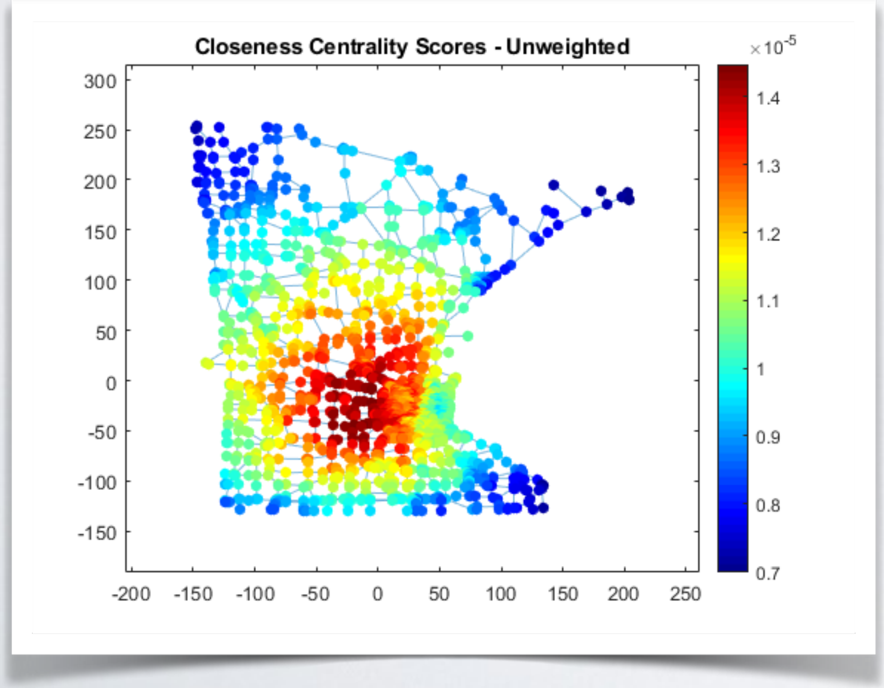


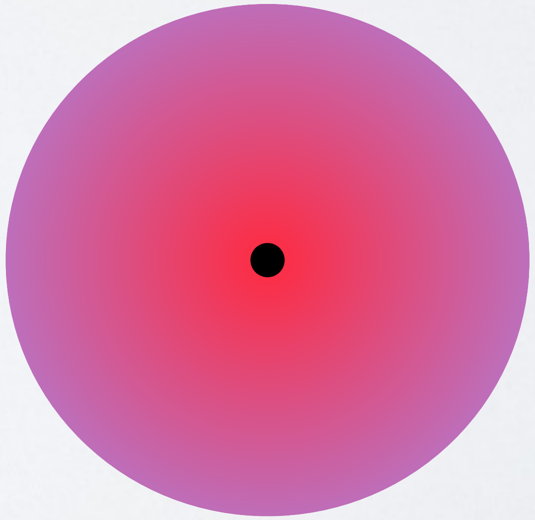
c = 0

FARNESS, CLOSENESS
HARMONIC CENTRALITY

FARNESS, CLOSENESS

- How close the node is to all other nodes
- Parallel with the center of a figure:
  - Center of a circle is the point of shorter average distance to any points in the circle





FARNESS, CLOSENESS

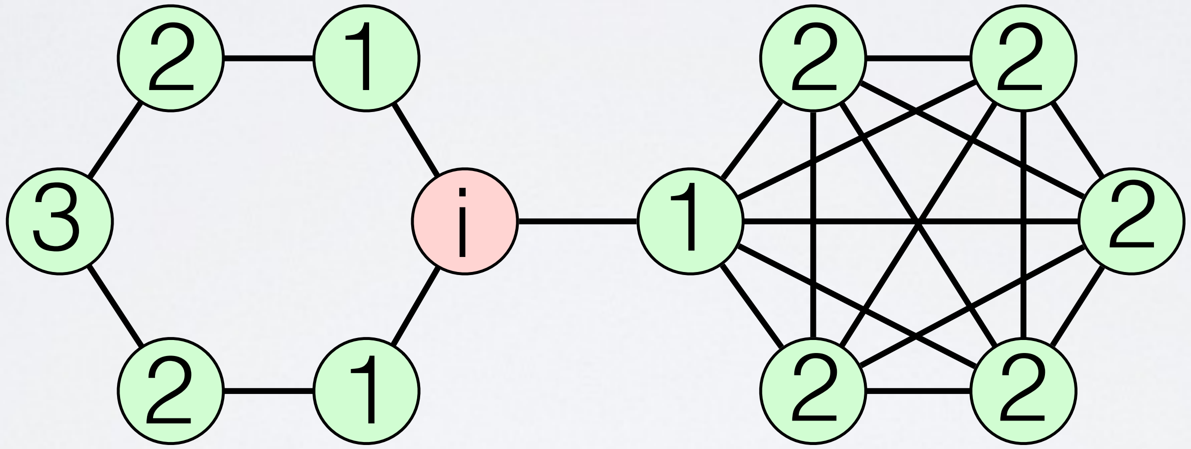
Farness: Average distance to all other nodes in the graph

$$\text{Farness}(u) = \frac{1}{N-1} \sum_{v \in V \setminus u} \ell_{u,v}$$

CLOSENESS CENTRALITY

Closeness: Inverse of the farness, i.e., how close the node is to all other nodes in term of shortest paths.

$$\text{Closeness}(u) = \frac{N-1}{\sum_{v \in V \setminus u} \ell_{u,v}}$$

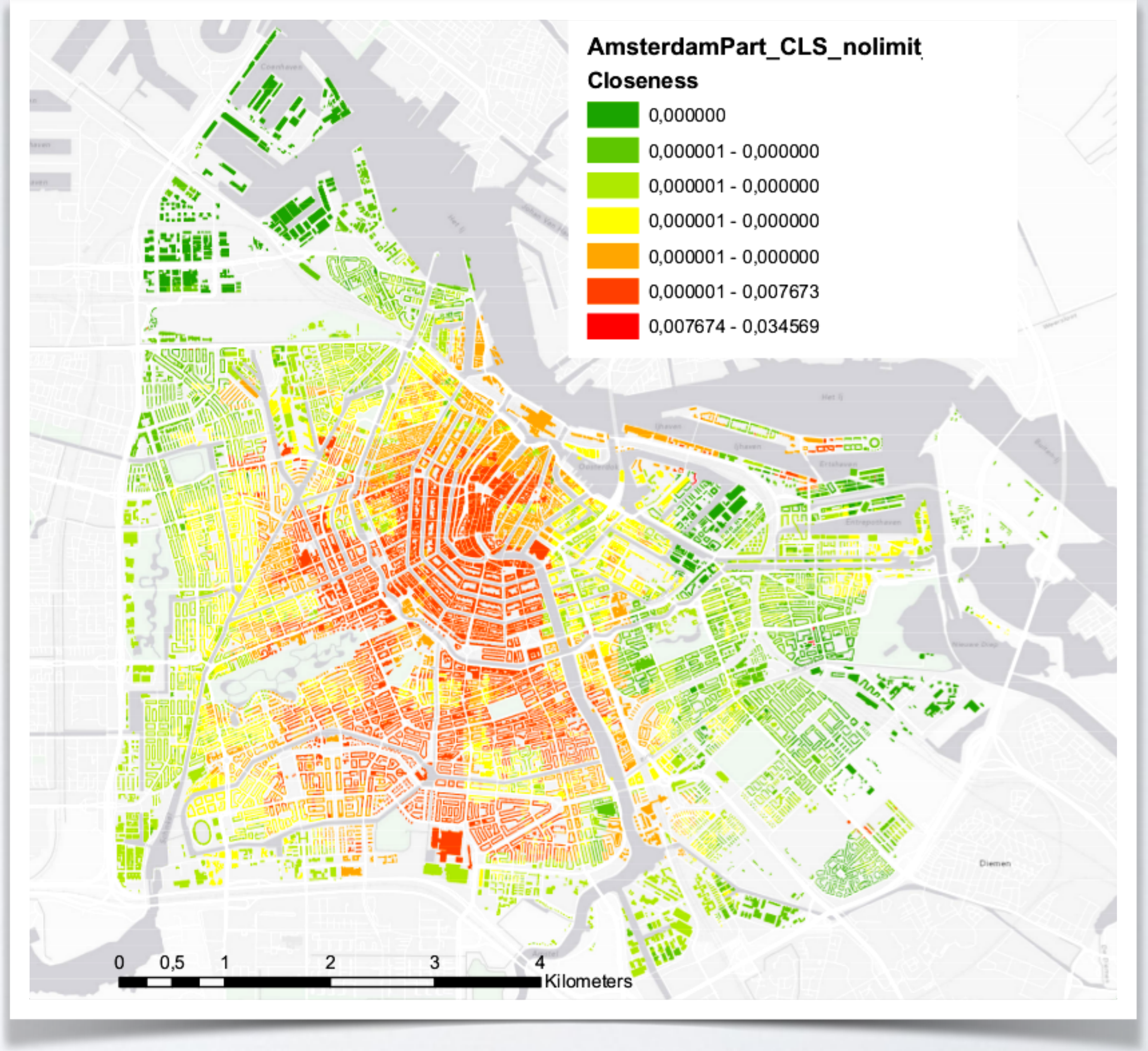


$$C_{cl}(i) = \frac{12-1}{(3 \times 1 + 7 \times 2 + 1 \times 3)} = \frac{11}{20} = 0.55$$

CLOSENESS CENTRALITY

Closeness: Inverse of the farness, i.e., how close the node is to all other nodes in term of shortest paths.

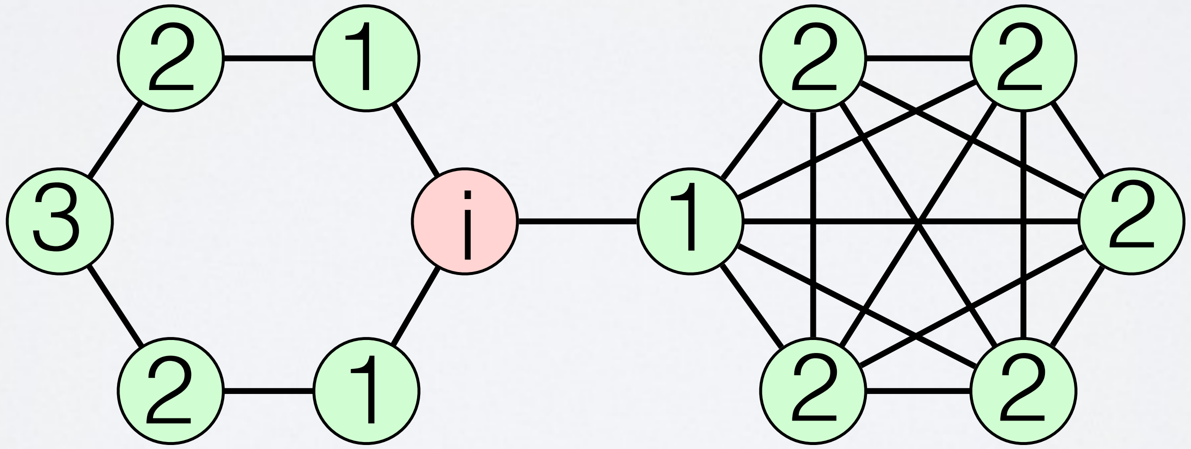
$$\text{Closeness}(u) = \frac{N-1}{\sum_{v \in V \setminus u} \ell_{u,v}}$$



Harmonic Centrality

Harmonic centrality: A variant of the closeness defined as the average of the inverse of distance to all other nodes (Harmonic mean). Well defined on disconnected network with $\frac{1}{\infty} = 0$. Its interpretation is the same as the closeness.

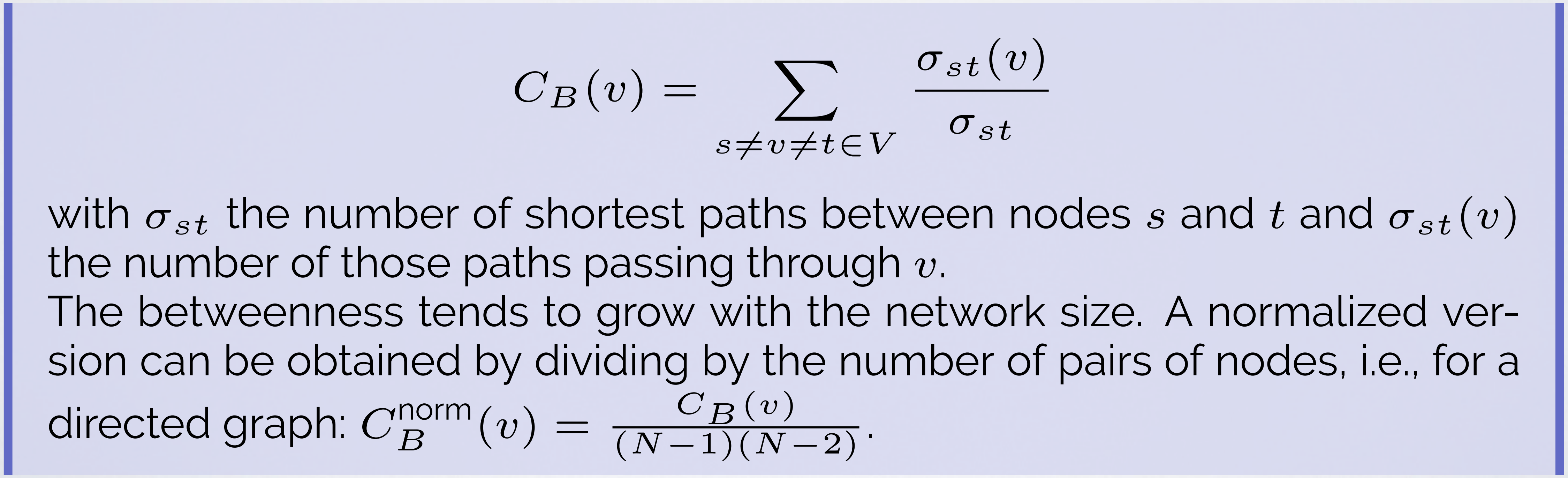
$$\text{Harmonic}(u) = \frac{1}{N-1} \sum_{v \in V \setminus u} \frac{1}{\ell_{u,v}}$$



$$C_h(i) = \frac{1}{12-1}\left(3 \times \frac{1}{1} + 7 \times \frac{1}{2} + 1 \times \frac{1}{3}\right) = \frac{41}{66} = 0.6212$$

BETWEENNESS CENTRALITY

- Measure how much the node plays the role of a bridge
- Betweenness of *u: f*raction of all the shortest paths between all the pairs of nodes going through u.



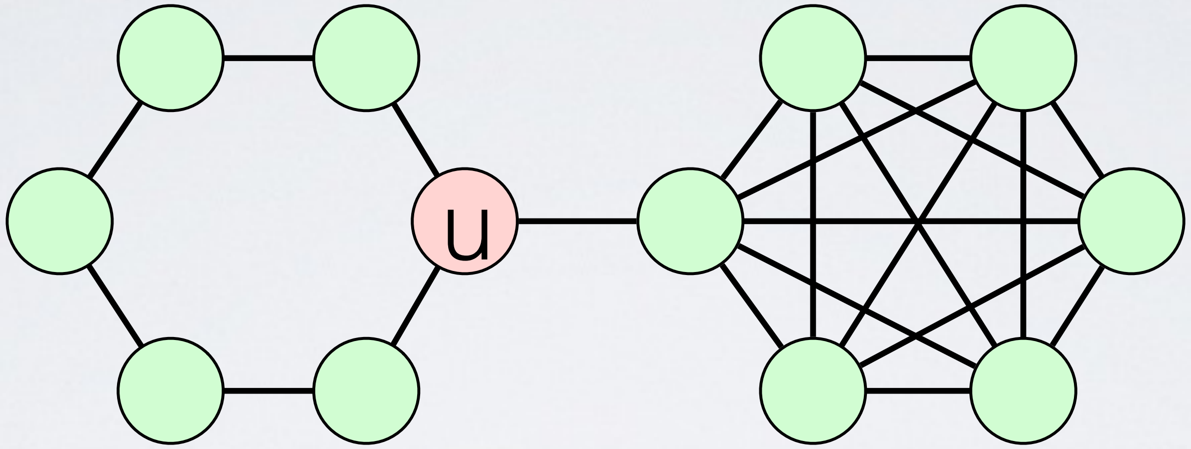
$$C_B(v) = \sum_{s \neq v \neq t \in V} \frac{\sigma_{st}(v)}{\sigma_{st}}$$

with σ_{st} the number of shortest paths between nodes s and t and $\sigma_{st}(v)$ the number of those paths passing through v.
The betweenness tends to grow with the network size. A normalized version can be obtained by dividing by the number of pairs of nodes, i.e., for a directed graph: $C_B^{\text{norm}}(v) = \frac{C_B(v)}{(N-1)(N-2)}$.

Betweenness Centrality

$$C_B(v) = \sum_{s \neq v \neq t \in V} \frac{\sigma_{st}(v)}{\sigma_{st}}$$

directed graph: $C_B^{\text{norm}}(v) = \frac{C_B(v)}{(N-1)(N-2)}$.



$$C_B(u) = 2\frac{5 * 6 + 1 + \frac{1}{2} + \frac{1}{2}}{11 * 10} = \frac{64}{110}$$

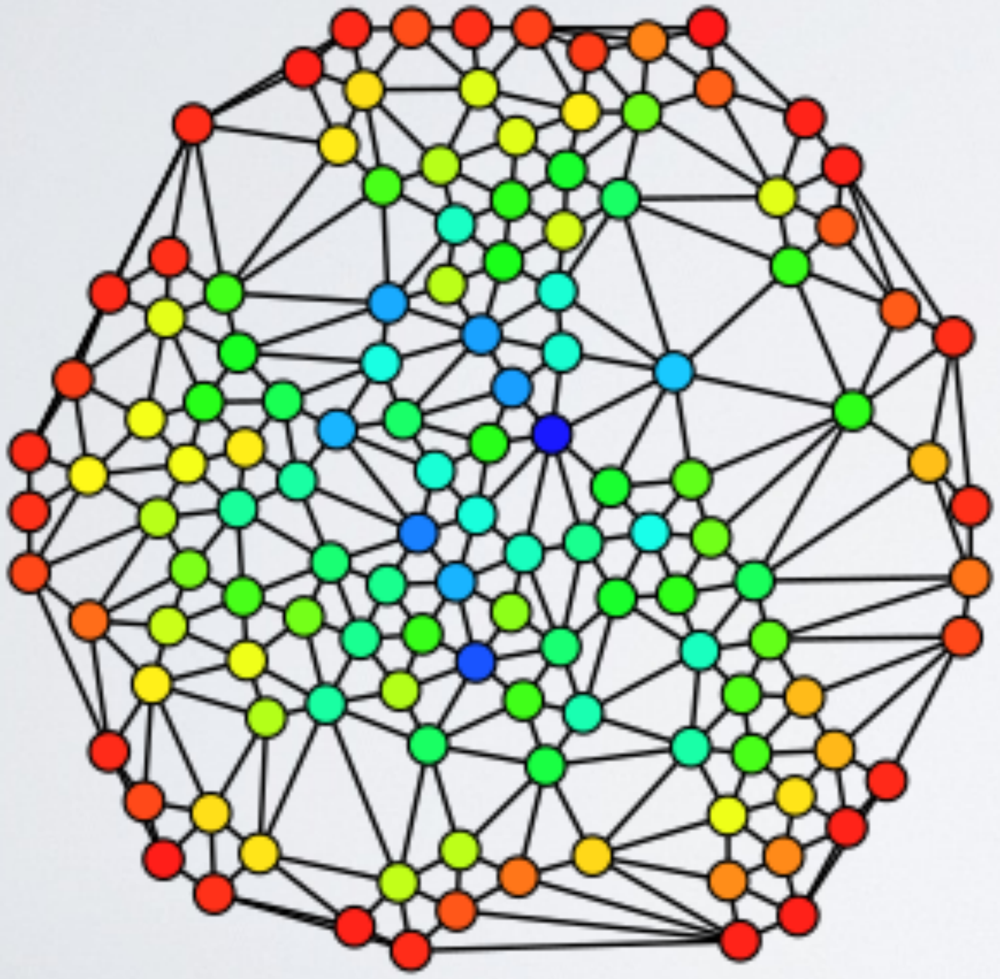
Exact computation:

Floyd-Warshall: *$O(n^3)$ time complexity*
$O(n^2)$ space complexity

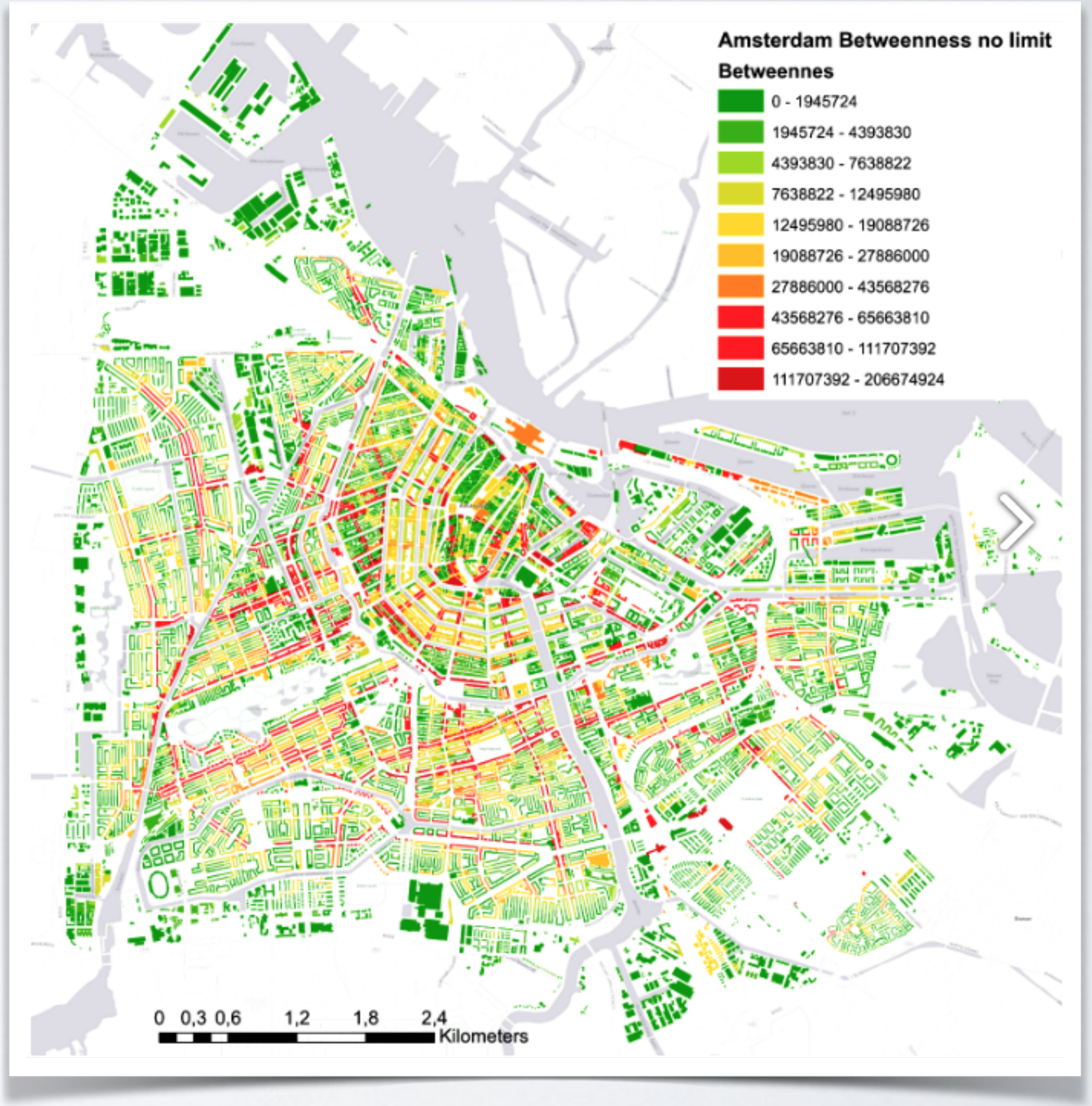
Approximate computation

Dijskstra: *O(n(m+n log n)) time complexity*

BETWEENNESS CENTRALITY



(blue higher)

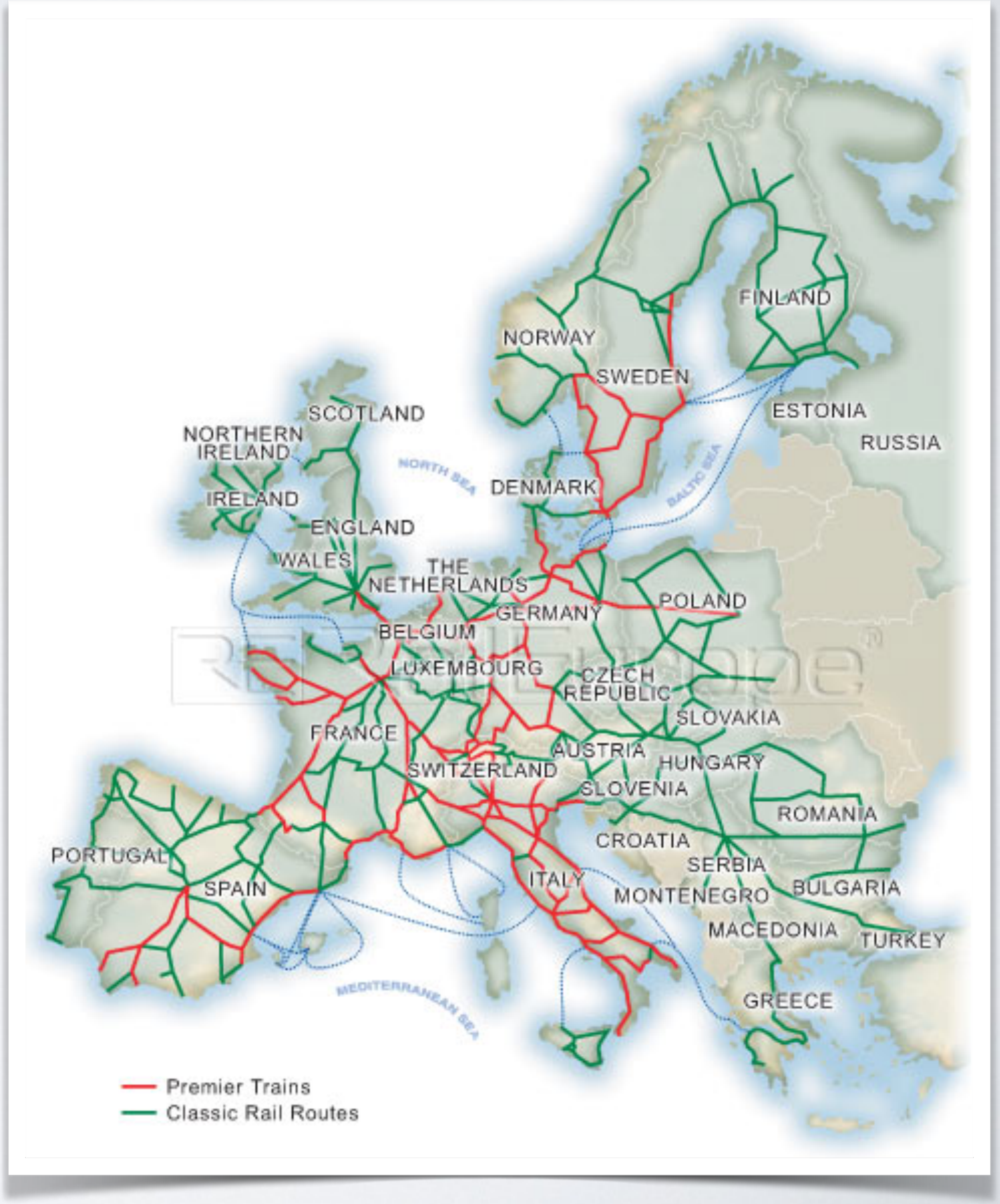


(red higher)

EDGE - BETWEENNESS

Same definition as for nodes

Can you guess the edge of highest betweenness in the European rail network ?



RECURSIVE DEFINITIONS

RECURSIVE DEFINITIONS

- Recursive importance:
  - **Important nodes** are those connected ***to important nodes***
- Several centralities based on this idea:
  - Eigenvector centrality
  - PageRank
  - …

RECURSIVE DEFINITION

- We would like scores such as :
  - Each node has a score (centrality),
  - If every node "sends" its score to its neighbors, the sum of all scores received by each node will be equal to its original score

$$C_u^{t+1} = \frac{1}{\lambda} \sum_{v \in N_u^{in}} C_v^t \tag{1}$$

- With λ a normalisation constant

RECURSIVE DEFINITION

- This problem can be solved by what is called the *power method:*
  - 1) We initialize all scores to random values
  - 2)Each score is updated according to the desired rule, until reaching a stable point (after normalization)

- Why does it converge?
  - Perron-Frobenius theorem (see next slide)
  - =>True for undirected graphs with a single connected component

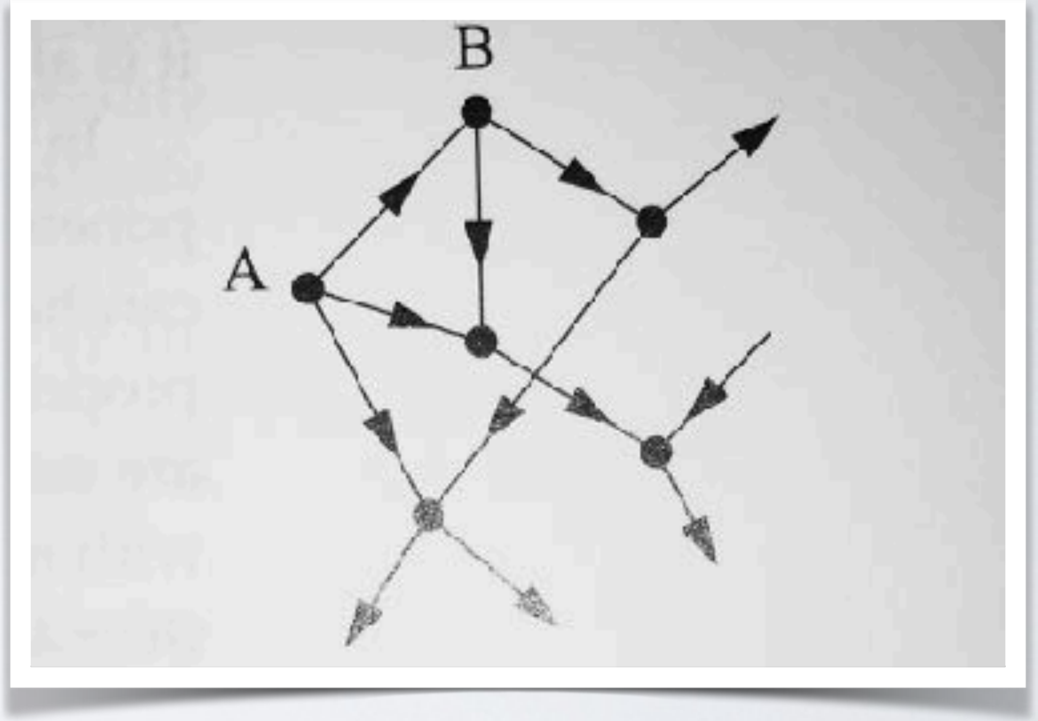
EIGENVECTOR CENTRALITY

- What we just described is called the Eigenvector centrality
- A couple eigenvector (x) and eigenvalue (λ) is defined by the following relation: $Ax = \lambda x$
  - x is a column vector of size *n,* which can be interpreted as the scores of nodes
- What Perron-Frobenius algorithm says is that the power method will always converge to the *leading eigenvector*, i.e., the eigenvector associated with the highest eigenvalue

Eigenvector Centrality

Some problems in case of directed network:

- Adjacency matrix is asymmetric
- 2 sets of eigenvectors (Left & Right)
- 2 leading eigenvectors
  - Use right eigenvectors : consider nodes that are pointing towards you



But problem with source nodes (0 in-degree)

-Vertex A is connected but has only outgoing link = Its centrality will be 0

-Vertex B has outgoing and an incoming link, but incoming link comes from A = Its centrality will be 0

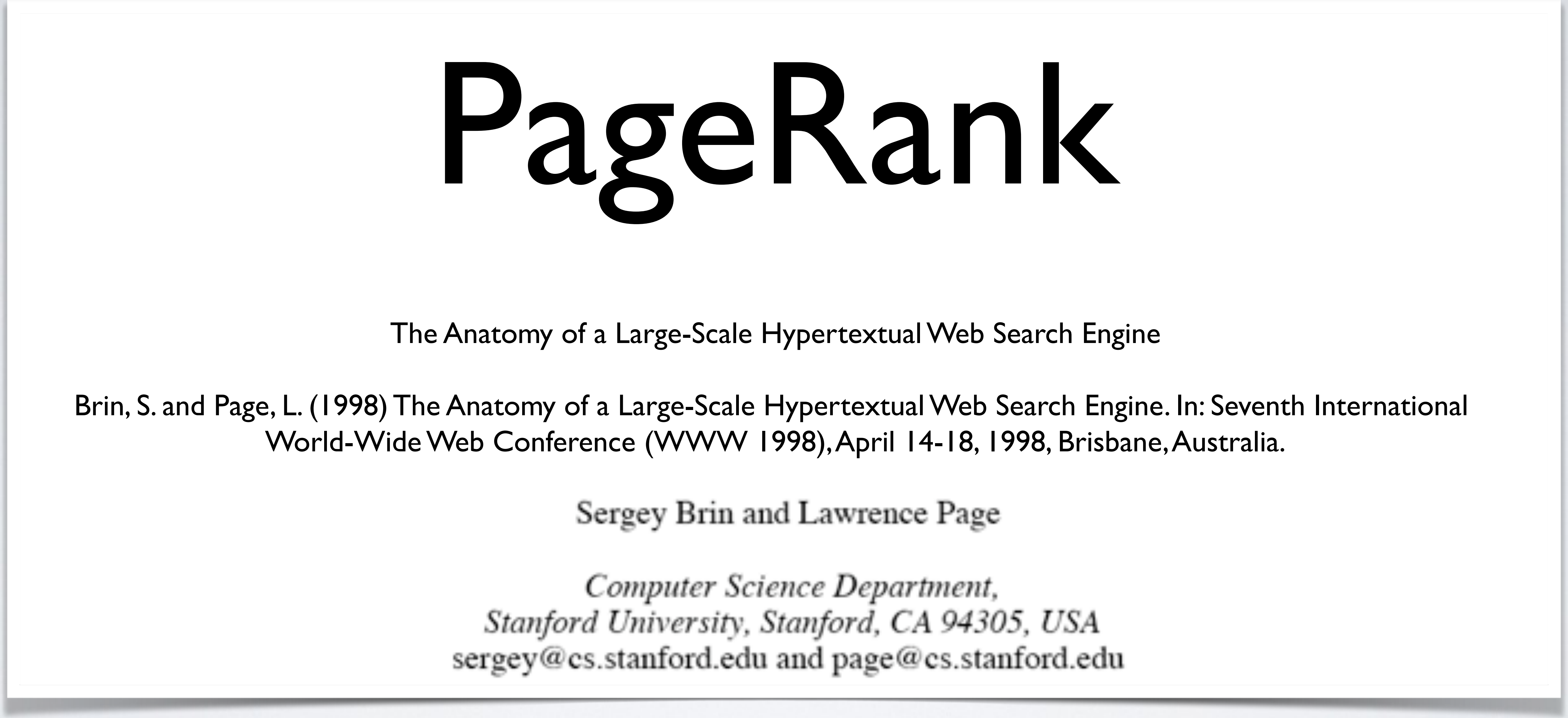
-etc.

Solution: Only in strongly connected component

Note: Acyclic networks (citation network) do not have strongly connected component

PageRank Centrality

- Eigenvector centrality generalised for directed networks



PageRank

The Anatomy of a Large-Scale Hypertextual Web Search Engine

Brin, S. and Page, L. (1998) The Anatomy of a Large-Scale Hypertextual Web Search Engine. In: Seventh International World-Wide Web Conference (WWW 1998), April 14-18, 1998, Brisbane, Australia.

Sergey Brin and Lawrence Page

Computer Science Department,
Stanford University, Stanford, CA 94305, USA
sergey@cs.stanford.edu and page@cs.stanford.edu

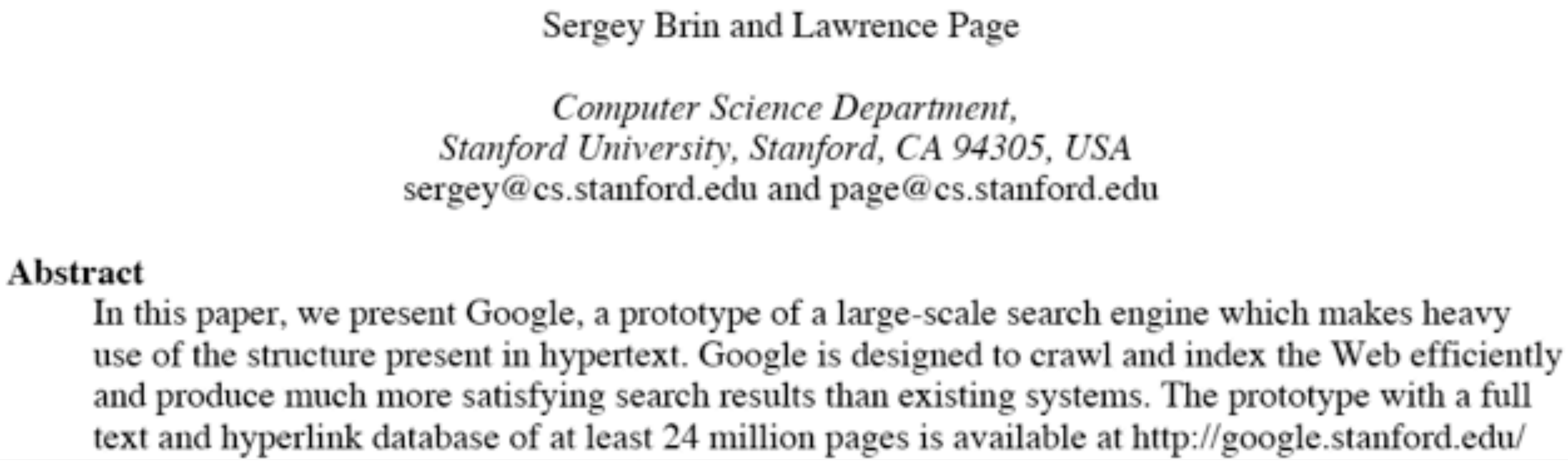
PageRank Centrality

- Eigenvector centrality generalised for directed networks

PageRank

The Anatomy of a Large-Scale Hypertextual Web Search Engine

Brin, S. and Page, L. (1998) The Anatomy of a Large-Scale Hypertextual Web Search Engine. In: Seventh International World-Wide Web Conference (WWW 1998), April 14-18, 1998, Brisbane, Australia.



Sergey Brin and Lawrence Page

Computer Science Department,
Stanford University, Stanford, CA 94305, USA
sergey@cs.stanford.edu and page@cs.stanford.edu

Abstract

In this paper, we present Google, a prototype of a large-scale search engine which makes heavy use of the structure present in hypertext. Google is designed to crawl and index the Web efficiently and produce much more satisfying search results than existing systems. The prototype with a full text and hyperlink database of at least 24 million pages is available at http://google.stanford.edu/

PageRank Centrality

(Side notes)

-“We chose our system name, Google, because it
is a common spelling of googol, or 10^{100} and fits well with our goal of building very large-scale search “

-“[…] at the same time, search engines have migrated from the academic domain to the commercial. **Up until now most search engine development has gone on at companies with little publication of technical details. This causes search engine technology to remain largely a black art and to be advertising oriented (see Appendix A). With Google, we have a strong goal to push more development and understanding into the academic realm.”**

-“[...], we expect that advertising funded search engines will be inherently biased towards the advertisers and away from the needs of the consumers."

PAGERANK

- 2 main improvements over eigenvector centrality:
  - In directed networks, problem of source nodes
    - => Add a constant centrality gain for every node
  - Nodes with very high centralities give very high centralities to all their neighbors (even if that is their only in-coming link)
    - => What each node "is worth" is divided equally among its neighbors (normalization by the degree)

$$C_u^{t+1} = \frac{1}{\lambda} \sum_{v \in N_u^{in}} C_v^t \quad => \quad C_u^{t+1} = \alpha \sum_{v \in N_u^{in}} \frac{C_v^t}{k_v^{out}} + \beta$$

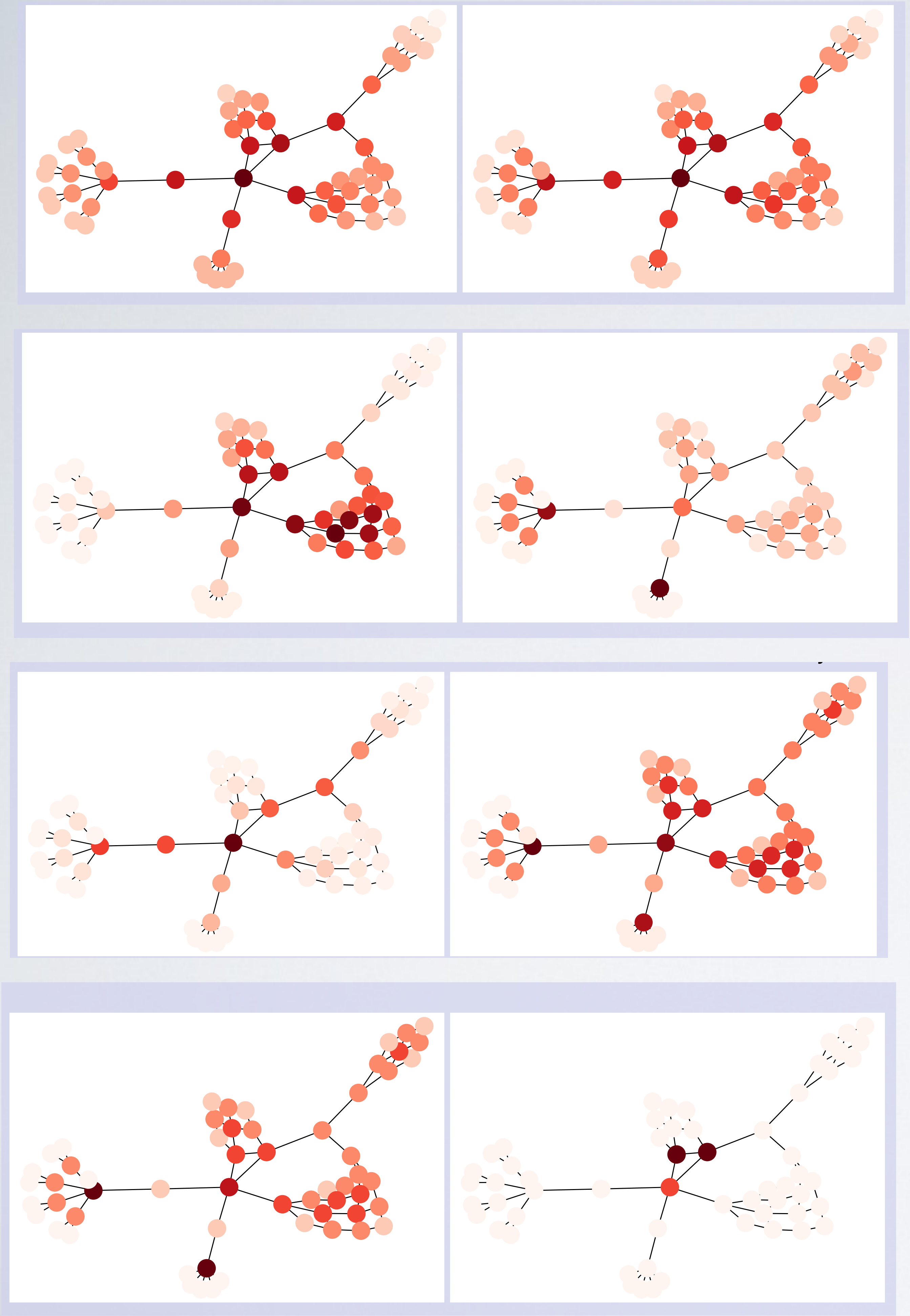
With by convention β=1 and α a parameter (usually 0.85) controlling the relative importance of β

PAGERANK

- Then how do Google rank when we do a research?
- Compute pagerank (using the power method for scalability)
- Create a subgraph of documents related to our topic
- Of course now it is certainly much more complex, but we don't really know:
  "Most search engine development has gone on at companies with little publication of technical details. This causes search engine technology to remain largely a black art" [Page, Brin, 1997]

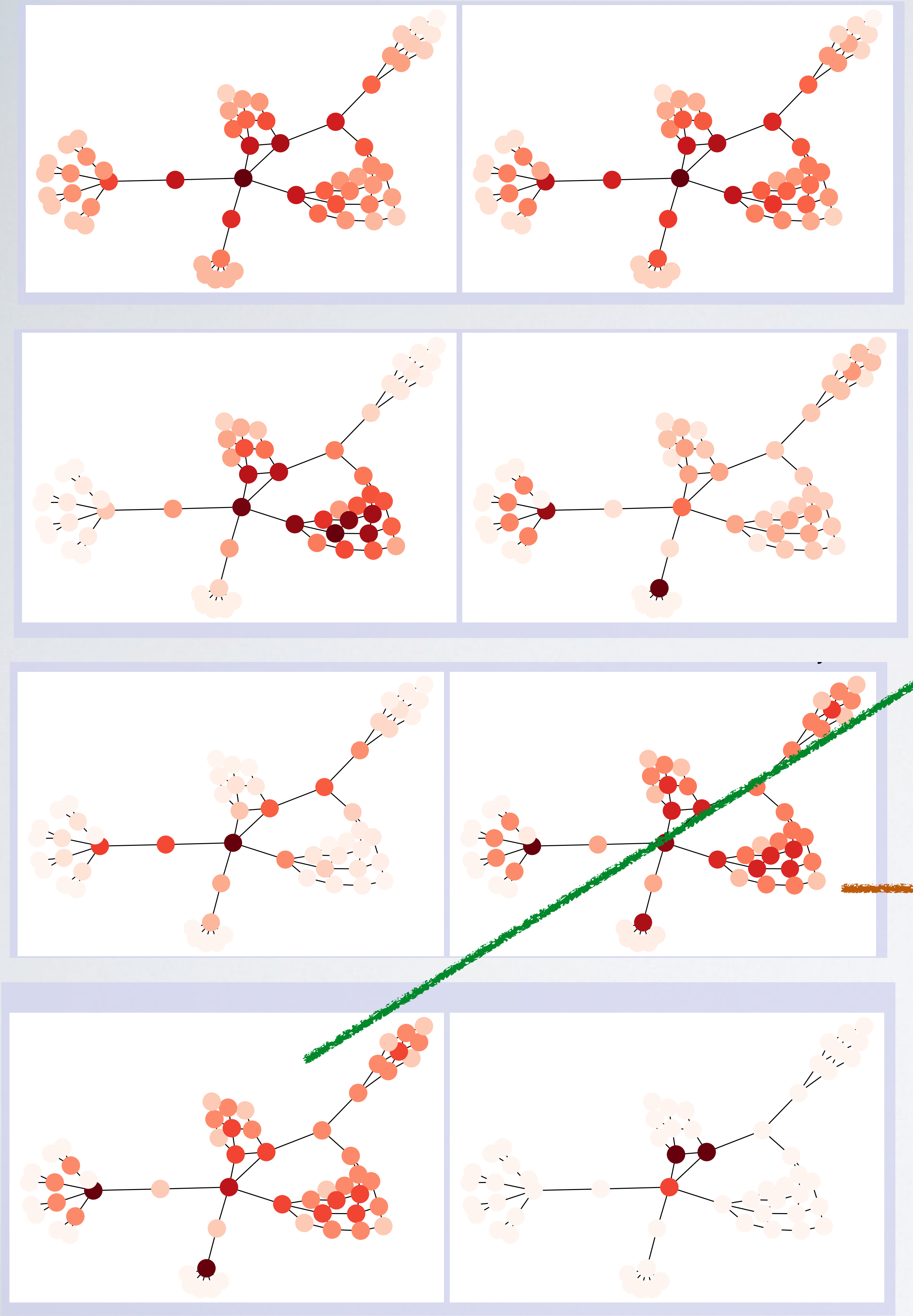
OTHERS

- Many other centralities have been proposed
- The problem is how to interpret them ?
- Can be used as supervised tool:
  - Compute many centralities on all nodes
  - Learn how to combine them to find chosen nodes
  - Discover new similar nodes
  - (roles in social networks, key elements in an infrastructure, …)



Which is which ?

Degree
Clustering coefficient
Closeness
Harmonic Centrality
Betweenness
Katz
Eigenvector
PageRank



Which is which ?

Degree

Clustering coefficient

Closeness

Harmonic Centrality

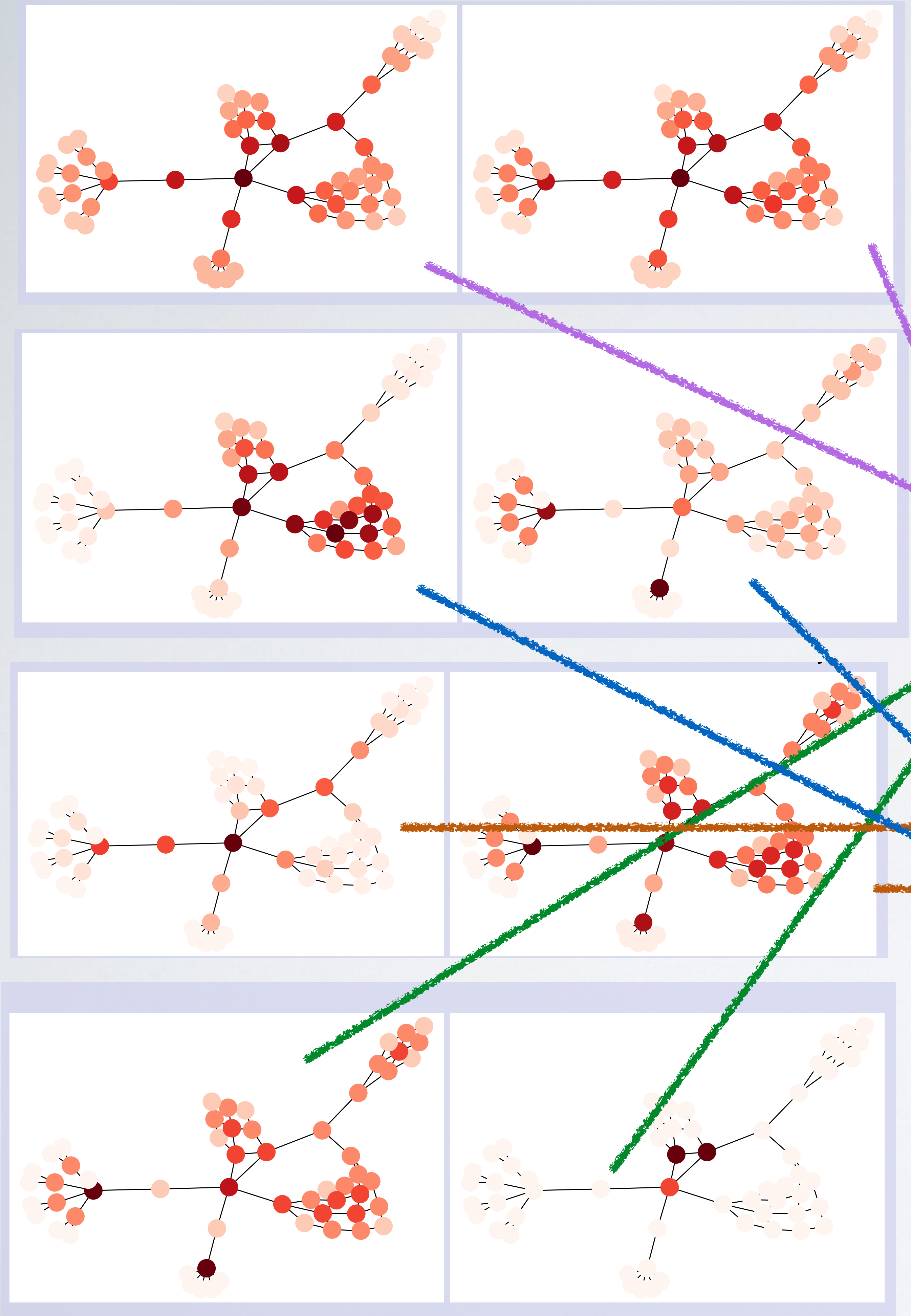
Betweenness

Katz

Eigenvector

PageRank

Which is which ?



Degree

Clustering coefficient

Closeness

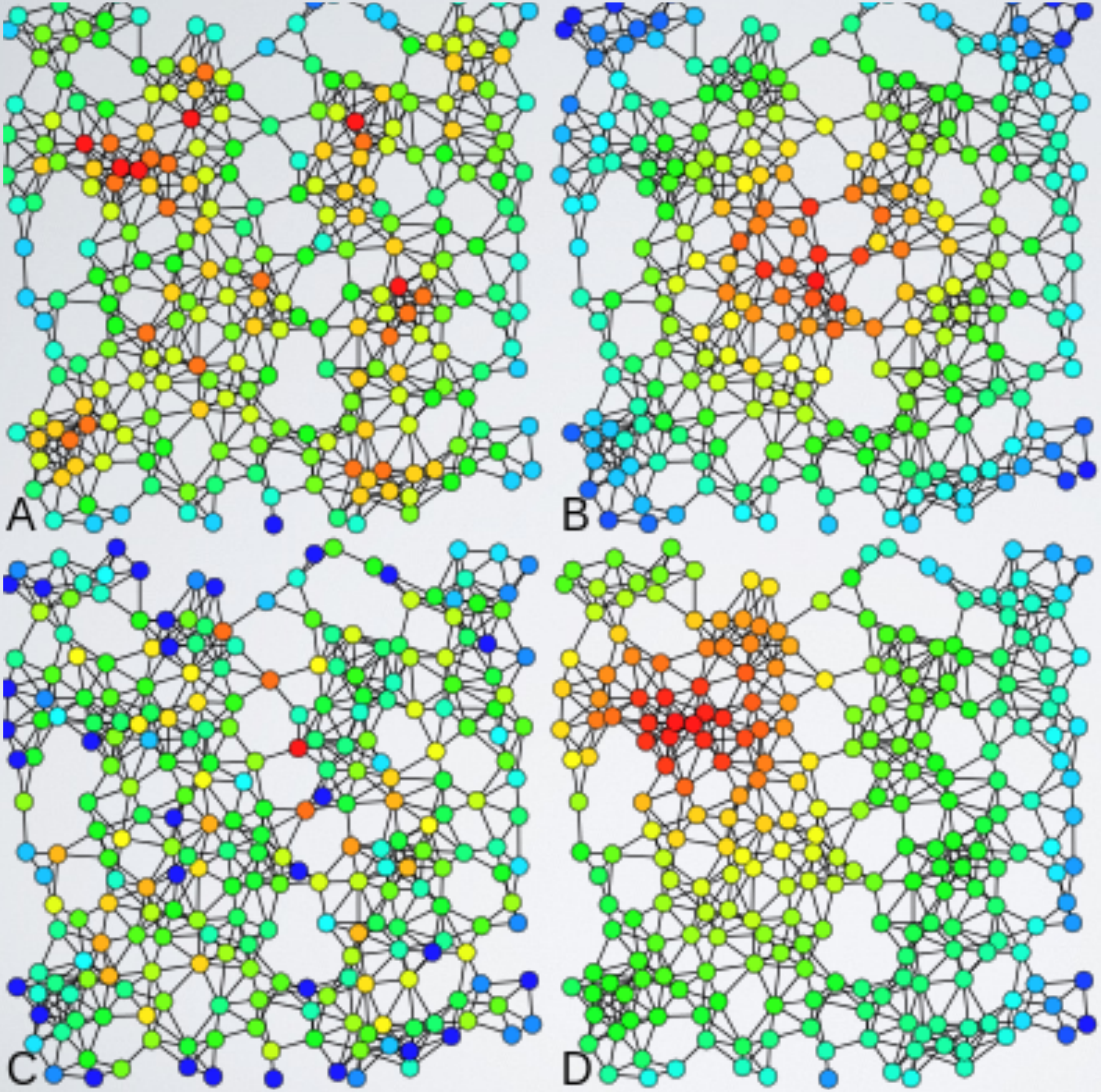
Harmonic Centrality

Betweenness

Katz

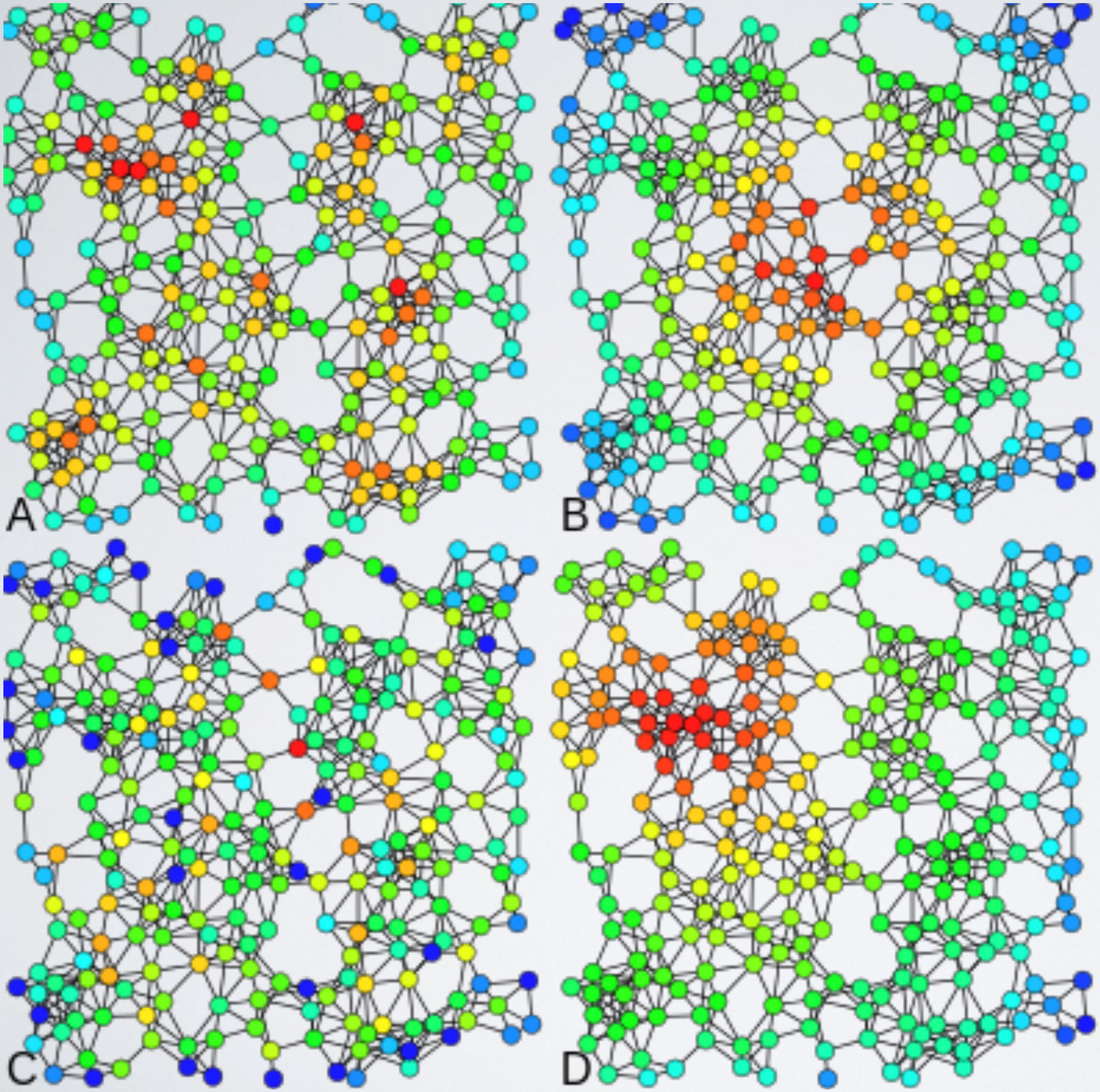
Eigenvector

PageRank



Try again :)

Degree
Betweenness
Closeness
Eigenvector



Try again :)

A: Degree
B:Closeness
C: Betweenness
D: Eigenvector

ASSORTATIVITY - HOMOPHILY

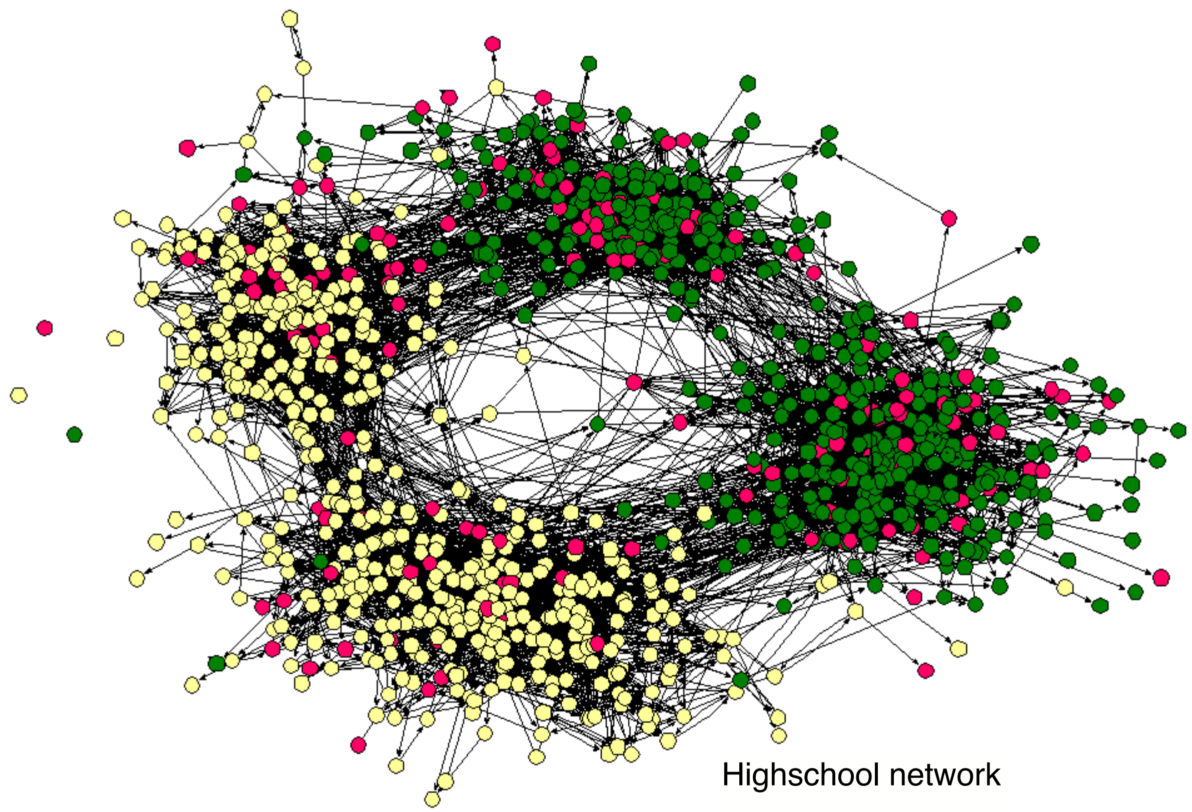
Homophily - Assortativity

"birds of a feather flock together"

- Property of (social) networks that nodes of the same attitude tends to be connected with a higher probability than expected
- It appears as correlation between vertex properties of $x(i)$ and $x(j)$ if $(i,j)\in E$

Vertex properties

- age
- gender
- nationality
- political beliefs
- socioeconomic status
- habitual place
- obesity
- …



Highschool network

Colored by ethnic groups (J Moody)

Homophily - Assortative mixing

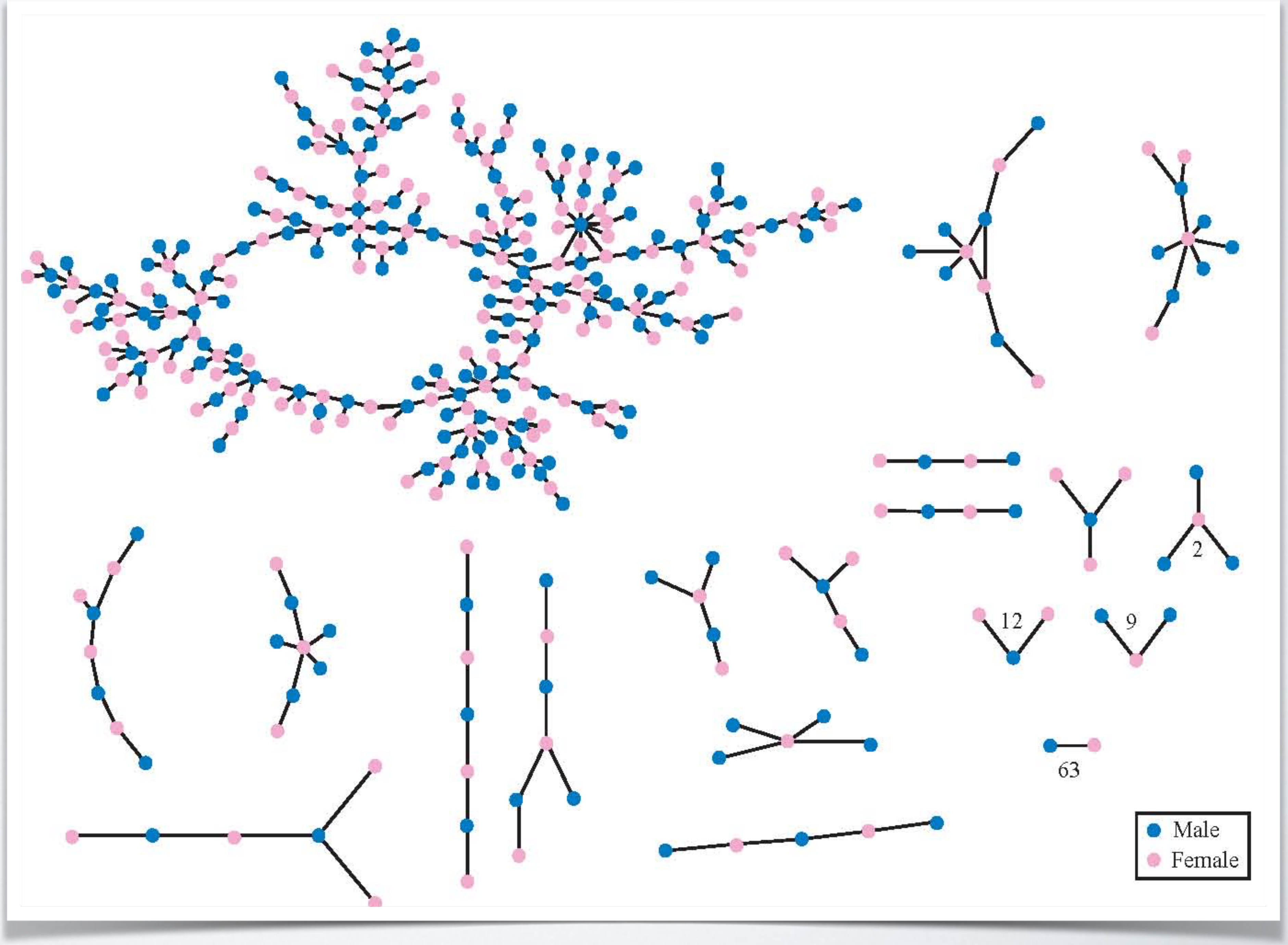
"Opposites attract"

Disassortativity - Heterophily

- Opposite of homophily: dissimilar nodes tend to be connected

Examples

- Sexual/Sentimental networks
- Predator - prey ecological networks



Homophily - Assortativity

Note on interpreting homophily

Homophily can be a link creation mechanism (nodes have a preference to connect with similar ones, so the network end up to be assortative), or a consequence of influence phenomenons (because nodes are connected, they tend to influence each other and thus become more similar).
Without access to the dynamic of the network and its properties, it is not possible to differentiate those effects.

Homophily - Assortative mixing

Categorical attributes

$$r = \frac{\sum_i e_{ii} - \sum_i a_i^2}{1 - \sum_i a_i^2}$$

e_{ii}: fraction of edges between nodes with same attributes

a_i: fraction of all edges having at least an end with property i.
=>Sum of degrees of nodes with property i divided by L

No assortative mixing : r=0 ($e_{ij} = a_i^2$)

Perfectly assortative: r=1

Assortative: r>0

Homophily - Assortative mixing

Assortativity index - Example

Let's see a fictional example of how to compute the assortativity index. Nodes are individuals, edges represent for instance some social interaction. Columns/Rows correspond to blood types, and numbers are expressed in fraction of the total number of edges.

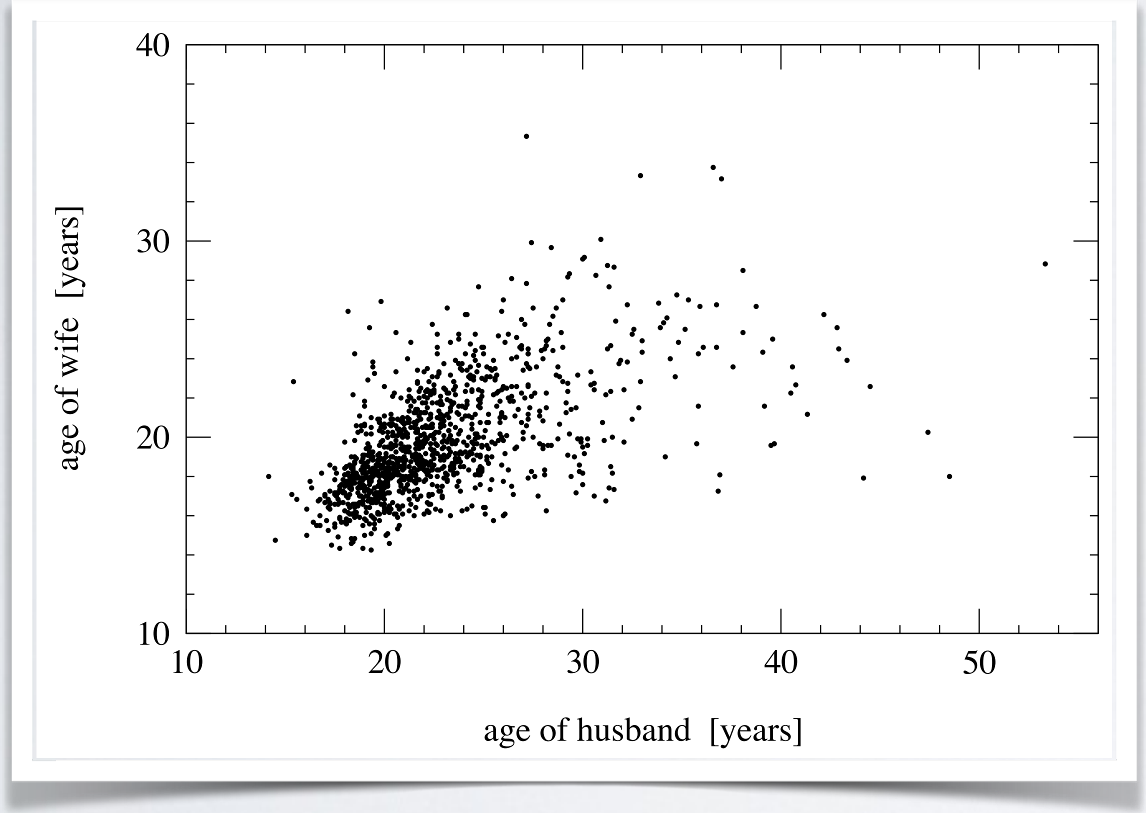
Blood Types	A	AB	B	O	a_i
A	0.30	0.05	0.1	0.05	*0.5*
AB	0.05	0.05	0	0	*0.1*
B	0.1	0	0.2	0	*0.3*
O	0.05	0	0	0.05	*0.1*
a_i	*0.5*	*0.1*	*0.3*	*0.1*	**1**

$$r = \frac{(0.3+0.05+0.2+0.05)-(0.5^2+0.1^2+0.3^2+0.1^2)}{1-(0.5^2+0.1^2+0.3^2+0.1^2)} = \frac{0.6+0.36}{1-0.36} = 0.375$$

Homophily - Assortative mixing

Numeric attributes

Pearson correlation coefficient of properties
at both extremities of edges



e_{xy}: fraction of edges joining nodes with values x and y

$$\sum_{xy} e_{xy} = 1, \qquad \sum_{y} e_{xy} = a_x, \qquad \sum_{x} e_{xy} = b_y$$

$$r = \frac{\sum_{xy} xy(e_{xy} - a_x b_y)}{\sigma_a \sigma_b},$$

with σ_a standard deviation of a_x

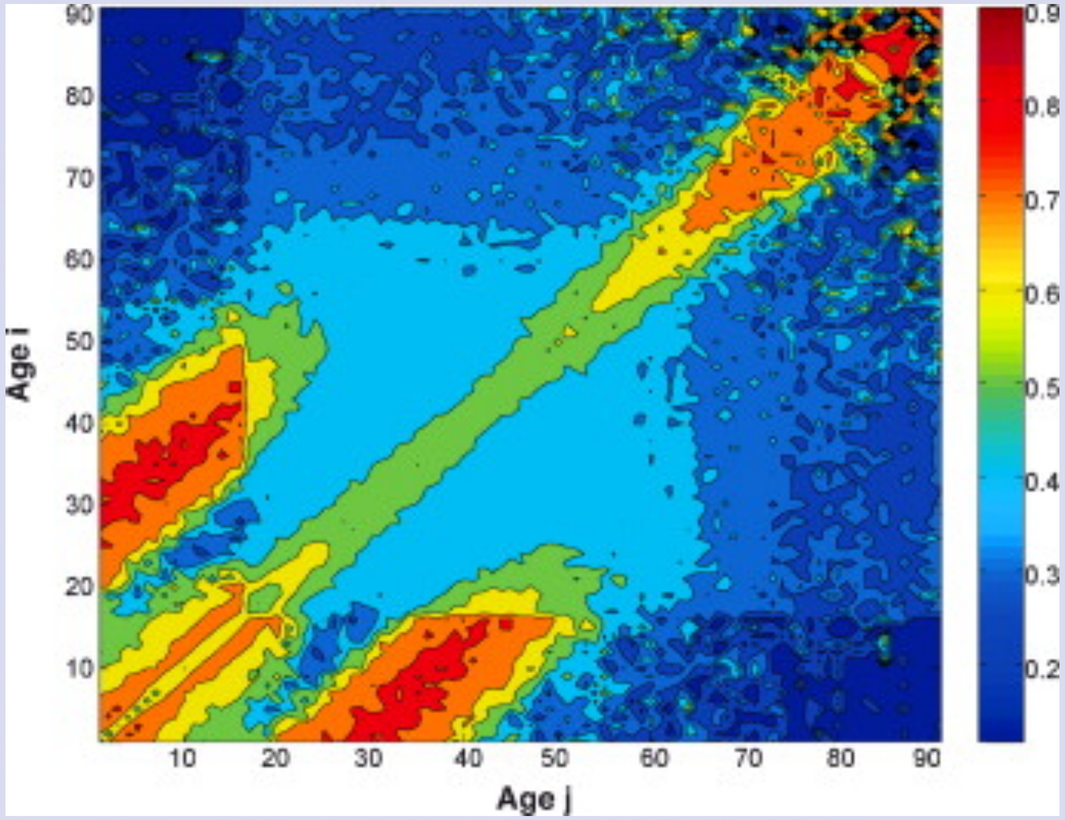
(Here, discrete version)

Mixing patterns

Beyond assortative and disassortative, we can study more generally **Mixing patterns**,
=>preference of nodes with attribute **a** to connect with nodes with attribute **b** (where a,b can be identical or different)

Mixing Patterns - example

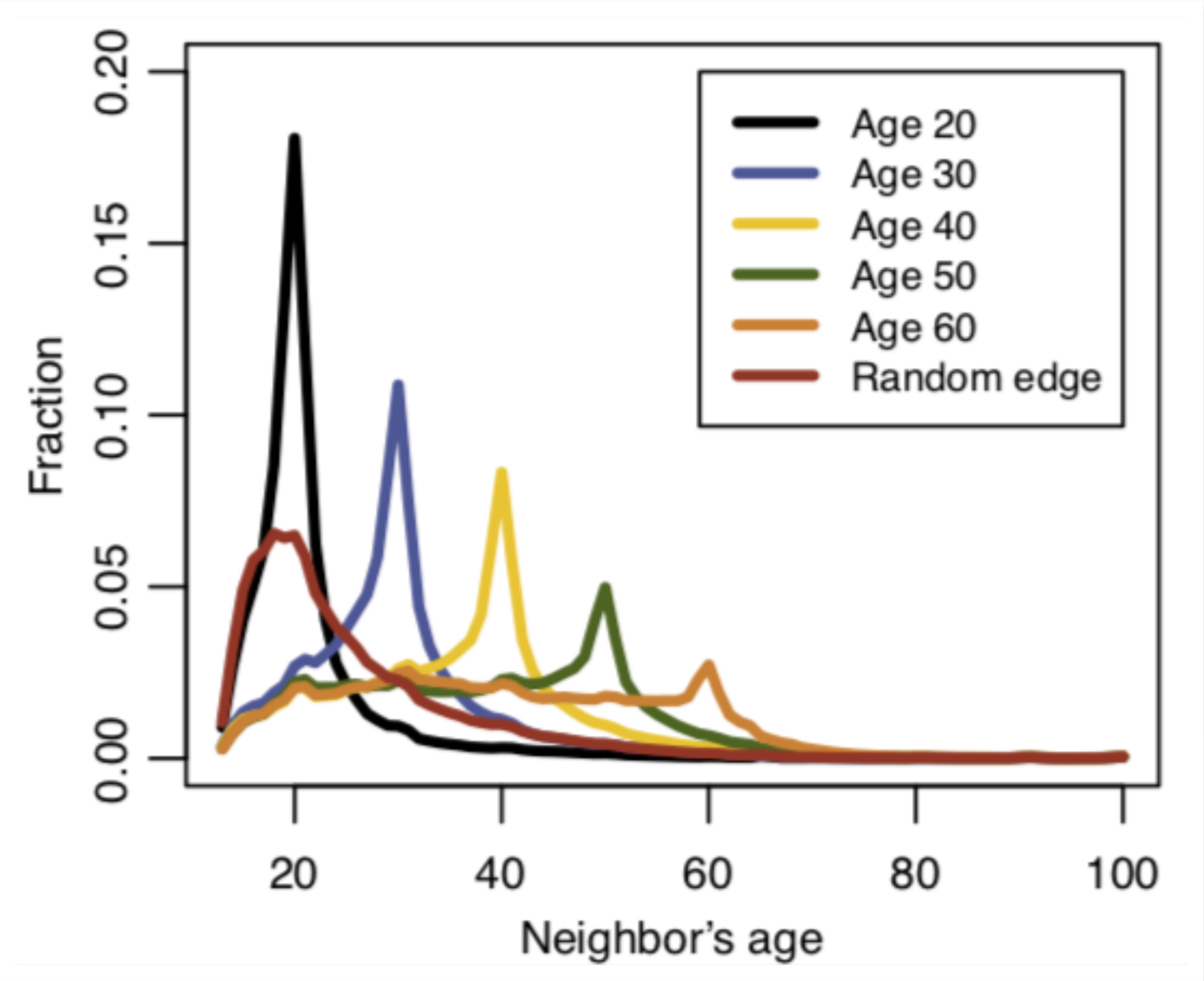
Example of mixing patterns of age in a network of interaction between individuals, reproduced from[a].



We can see that there is some level of assortativity (hig hvalues on the diagonal), but that there are also some more complex mixing patterns, for instance between age 10 and 40, approximately, here interpreted as child-parents relationships.

[a] Del Valle et al. 2007.

Mixing patterns



- [The Anatomy of the Facebook Social Graph, Ugander et al. 2011]