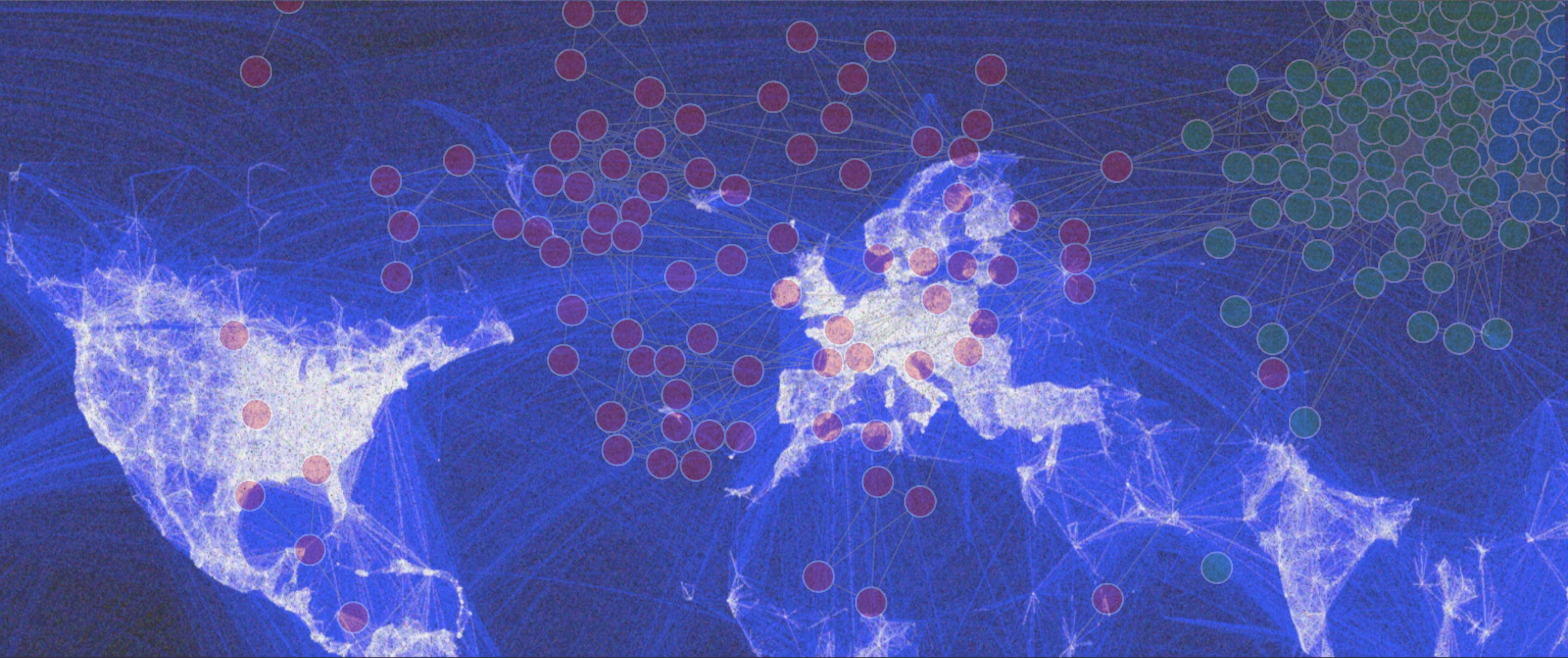


Complex Networks

Vincent Gauthier
Associate Professor
@vincentgauthier
http://complex.luxbulb.org
Telecom SudParis-Institut Mines-Telecom/CNRS SAMOVAR

Software

R

Matlab

Python

Networkx [Python]

Graph-tool [Python+C++]

GraphLab [Python+C++]

Graph Editor

Gephi

GraphViz

Pajek

Datasets

Mark Newman's network data set

Stanford Network Analysis Project

Carnegie Mellon CASOS data sets

NCEAS food web data sets

UCI NET data sets

Pajek data sets

Linkgroup's list of network data sets

Barabasi lab data sets

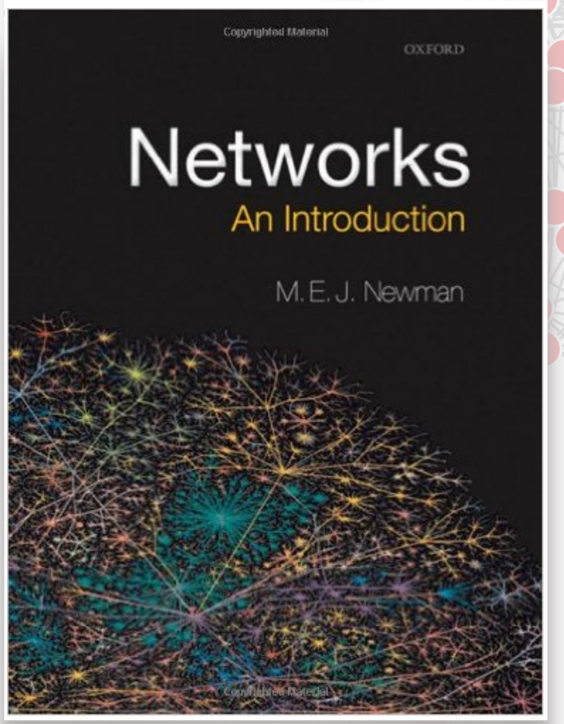
Jake Hofman's online network data sets Alex Arenas's data sets

Announcement

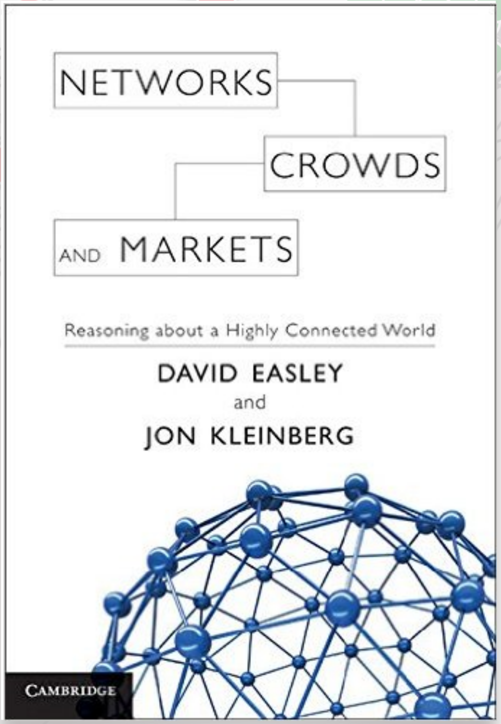
Tomorrow bring your laptop with you and install the following softwares:

http://continuum.io/downloads

http://gephi.github.io/users/download/



Networks An Introduction
Mark Newman, 2010.



Networks, Crowds, and Markets
Reasoning About a Highly Connected World
David Easley, Jon Kleinberg, 2010.

1. Definition of a network
2. Basics of graph theory
3. Describing a network
4. Network centrality
5. Eigen centrality, PageRank
6. Small world network (Watts-Strogatz)
7. Network Growth (Barabasi-Albert)
8. Social Network Analysis
9. Diffusion on network

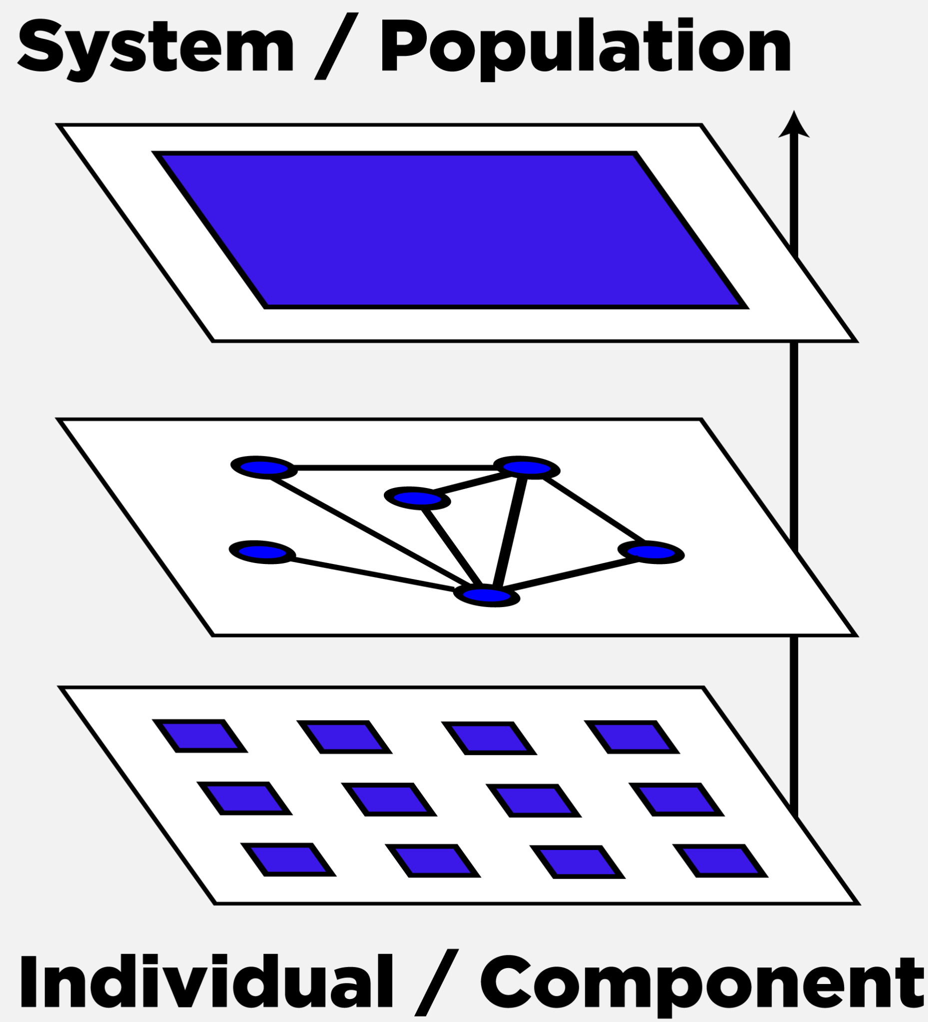
Lecture 1

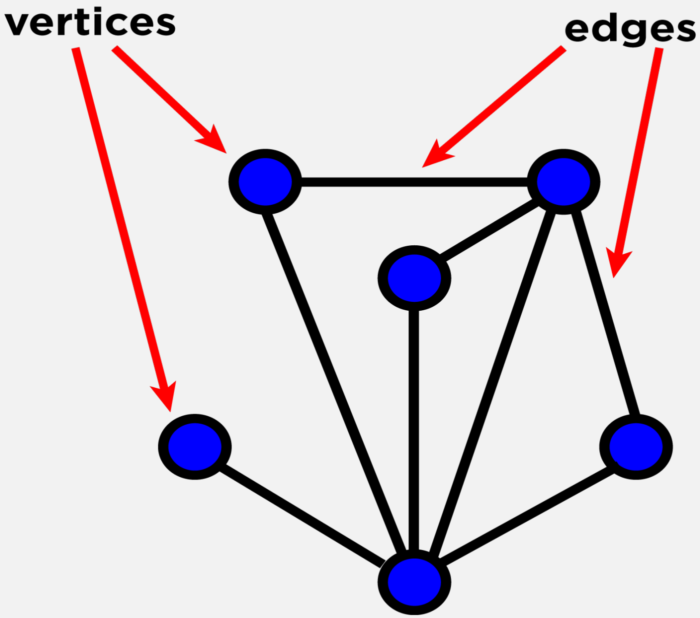
Lecture 2

Lecture 3

What are networks ?

- an approach
- a mathematical representation
- provide structure to complexity
- structure above
  - system / population
- structure bellow
  - Individual / component





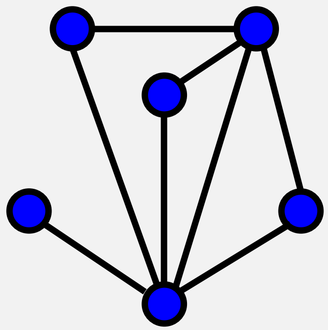
What is a vertex ?

V distinct objects (vertices / nodes)

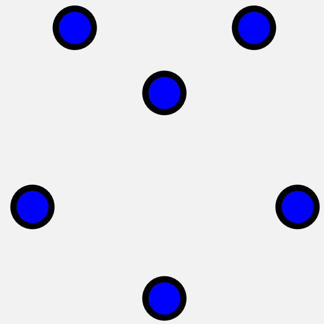
When are two vertices connected?

$$E \subseteq V \times V$$

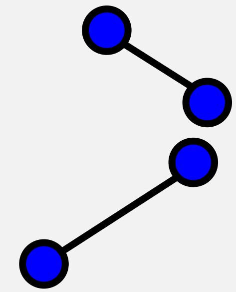
pairwise relations (edges / links / ties)



network



vertex



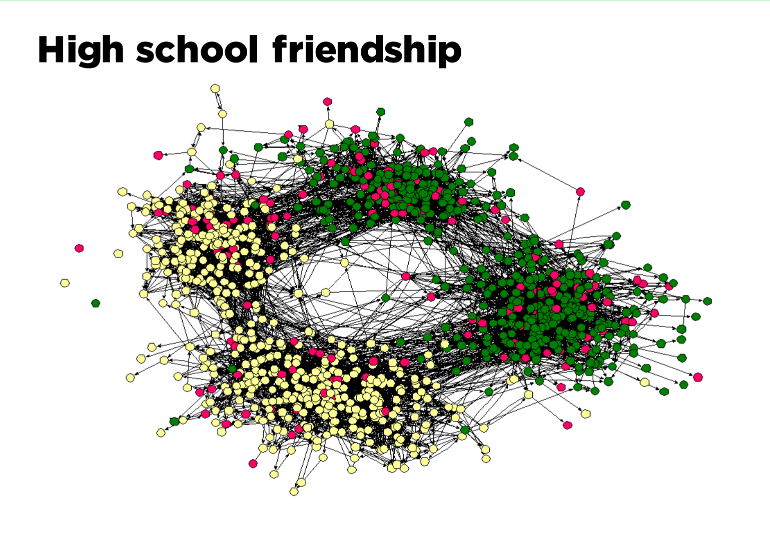
edges

Telecom	Internet (1)	Router	IP network adjacency
	Internet (2)	Autonomous system (ISP)	BGP connection
Information	Software	Function	Function call
	World wild web	Webpage	Hyperlink
	Document	Article, patent	Citation
Electricity	Power grid	Relay station	Transmission line
Transport	Rail system	Rail station	Railroad tracks
	Road network (1)	Intersection	Pavement
	Road network (2)	Named road	intersection
	Airport network	Airport	Non-stop flight
Social	Friendship network	Person	Friendship
Biological	Food web	Species	Predator, resources transfer

Social Networks I

vertex: person

edges: Friendship, collaboration, communication, exchanges



Social Network II

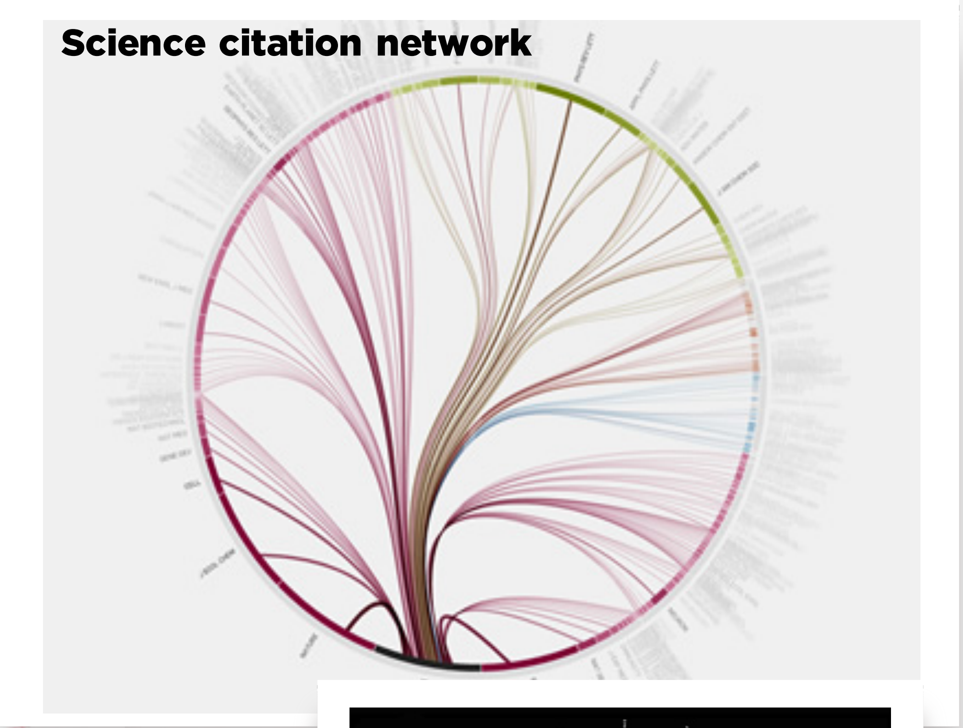
Science collaboration network

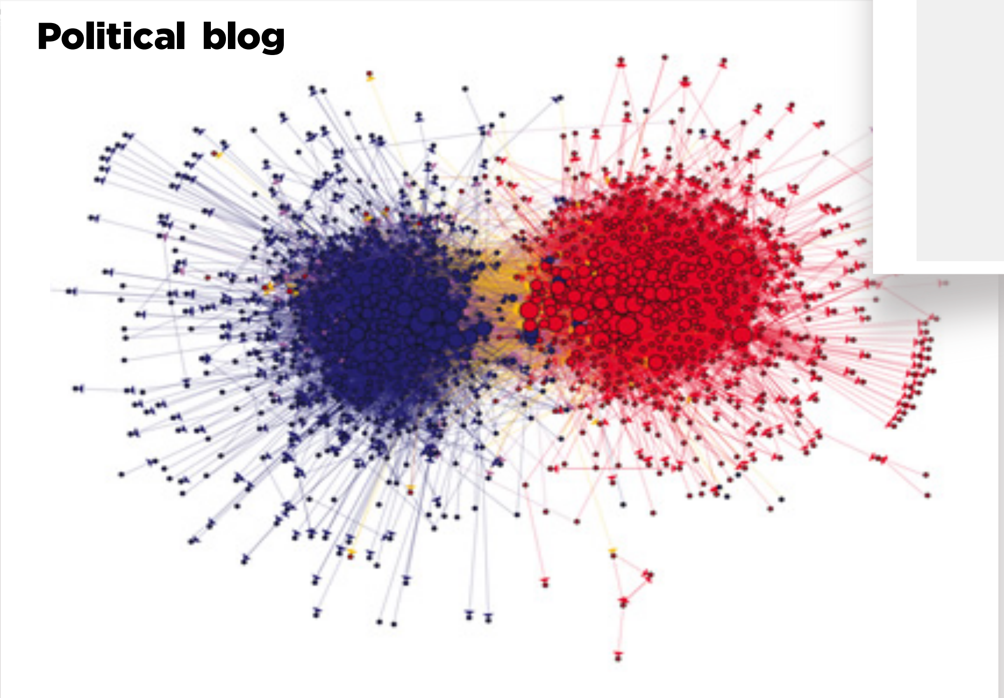
Community structure reveal the spatial clustering of collaboration network

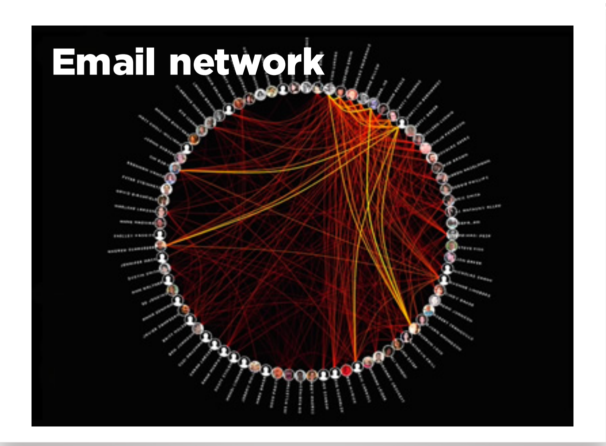
Information Networks

vertex: books, blogs, webpages, etc.

edges: citations, hyperlinks, recommendations, similarity, etc.



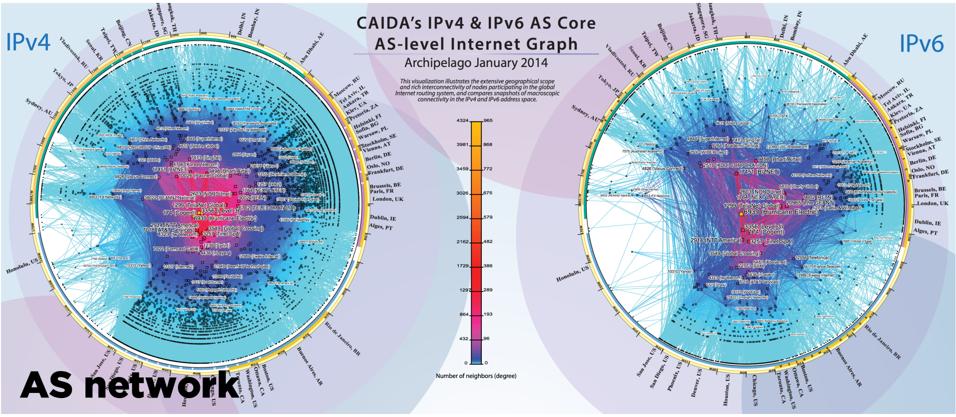


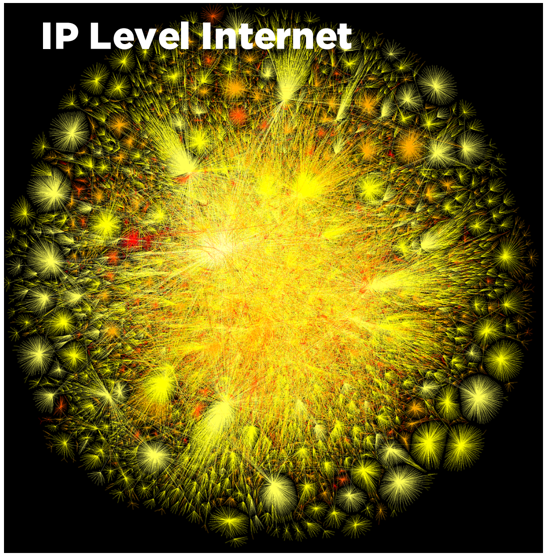


Communication Networks

vertex: network router, ISP, email address, mobile phone number, etc

Edges: exchange of information

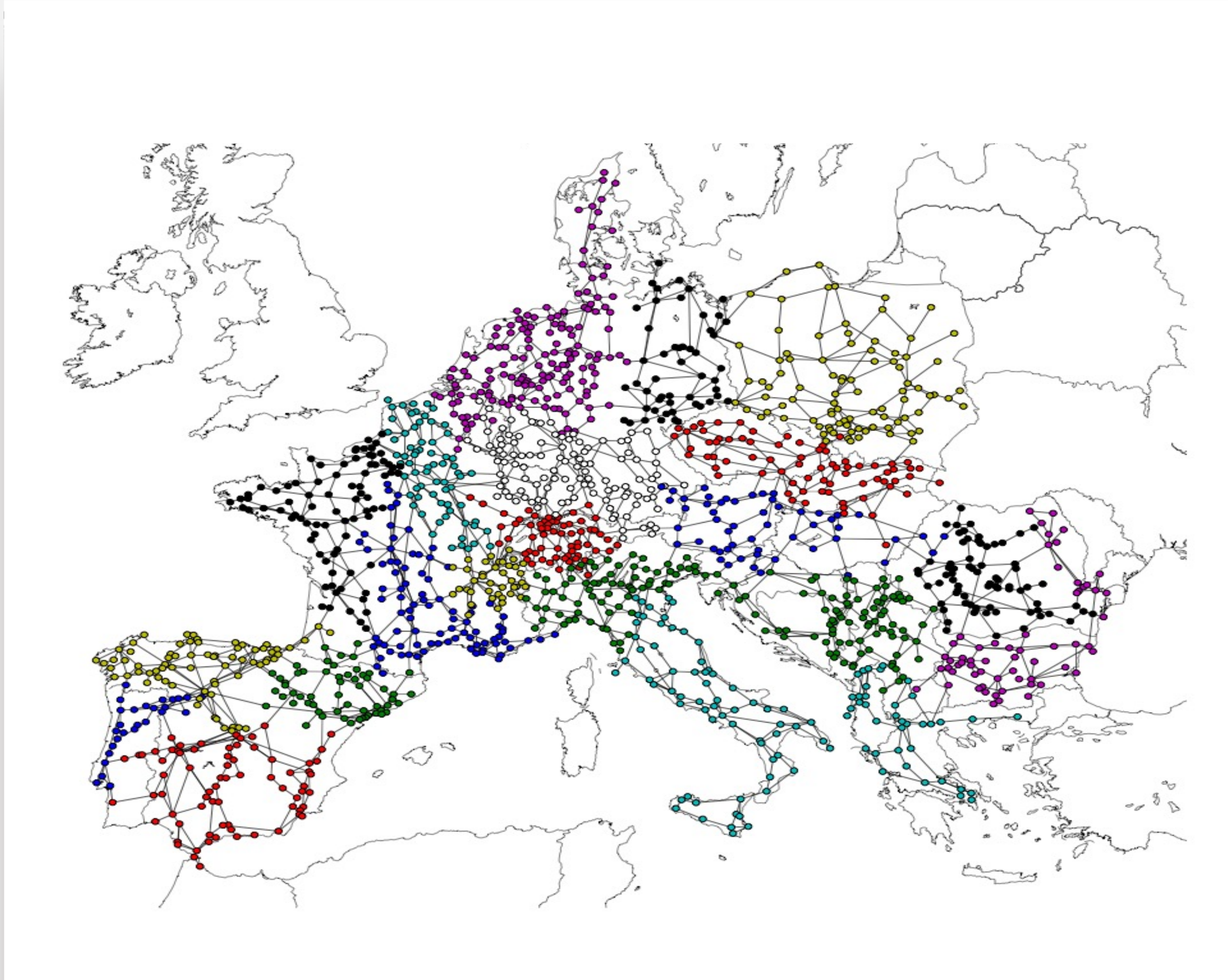




Infrastructure network

vertex: Power Plan, transformer

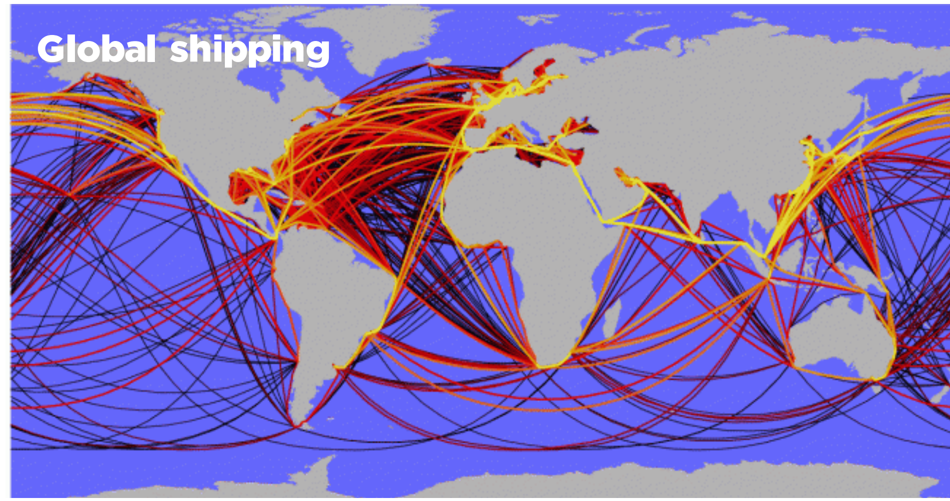
Edges: Electrical lines

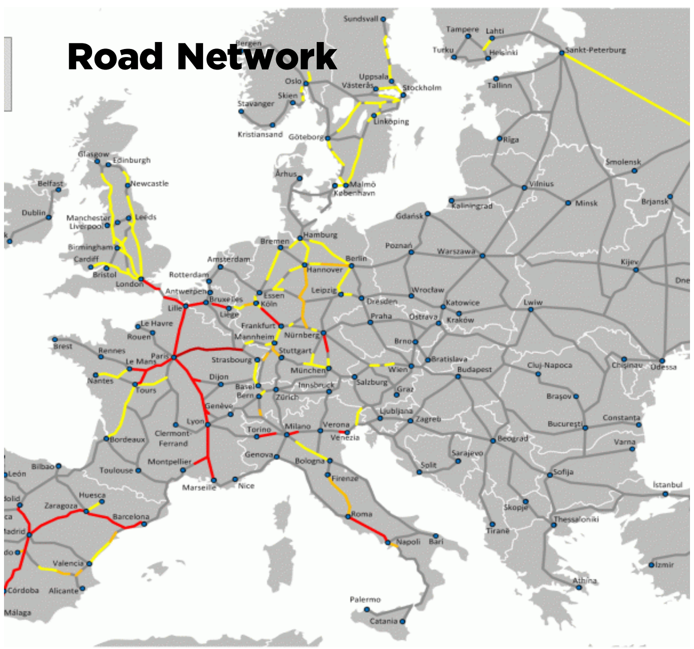


Transport Networks

vertex: city, airport, junction, railway, station, river confluence, etc.

Edges: physical transportation of material



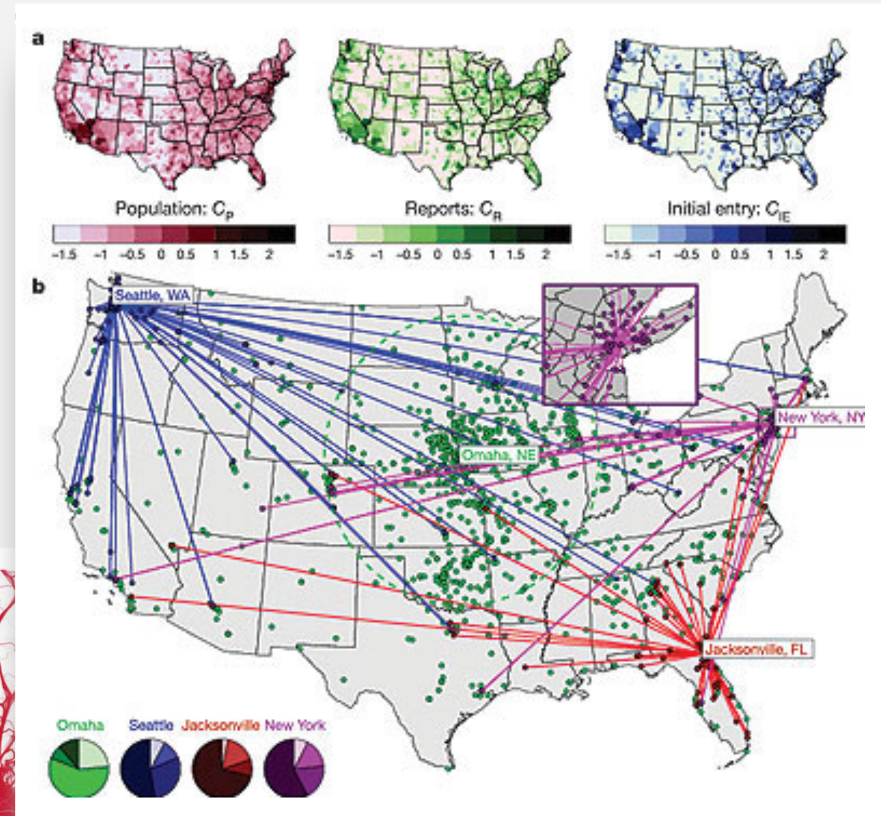


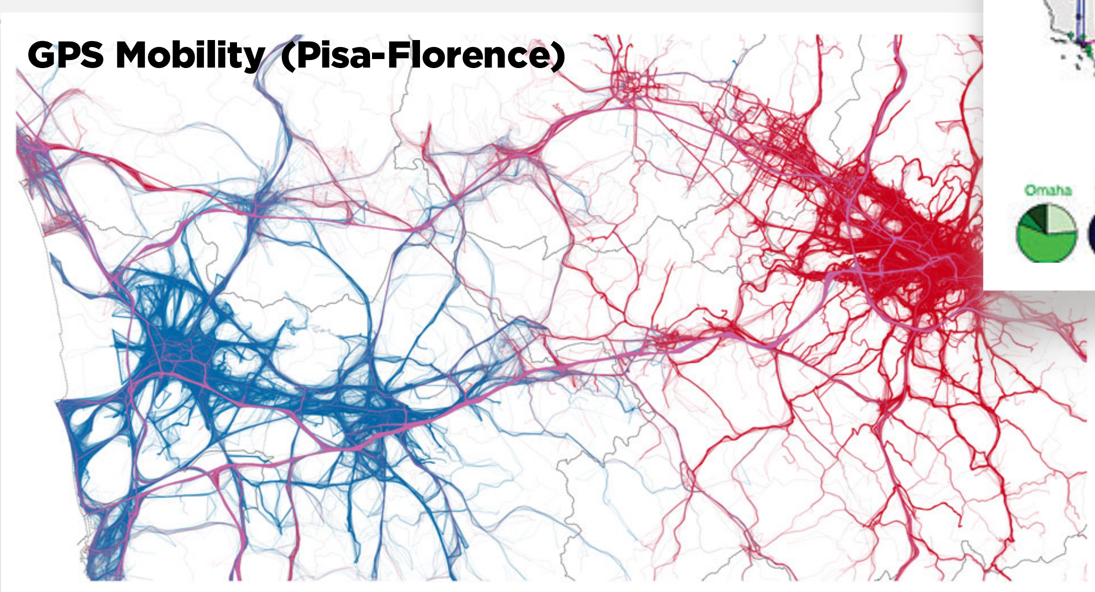
Transport Networks

Continued

Human mobility

Vertex: person, place, city, road intersections
Edges: physical transportation network, mobility



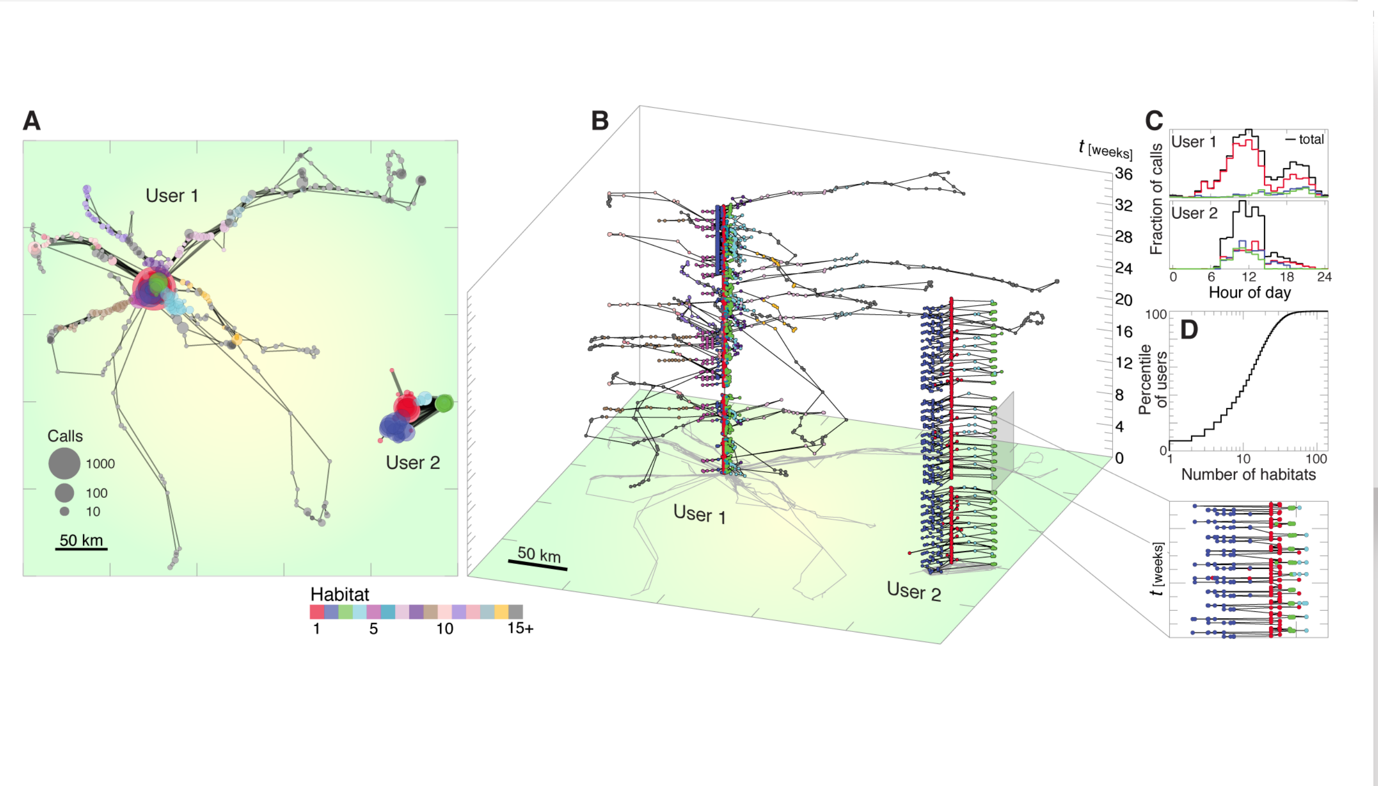


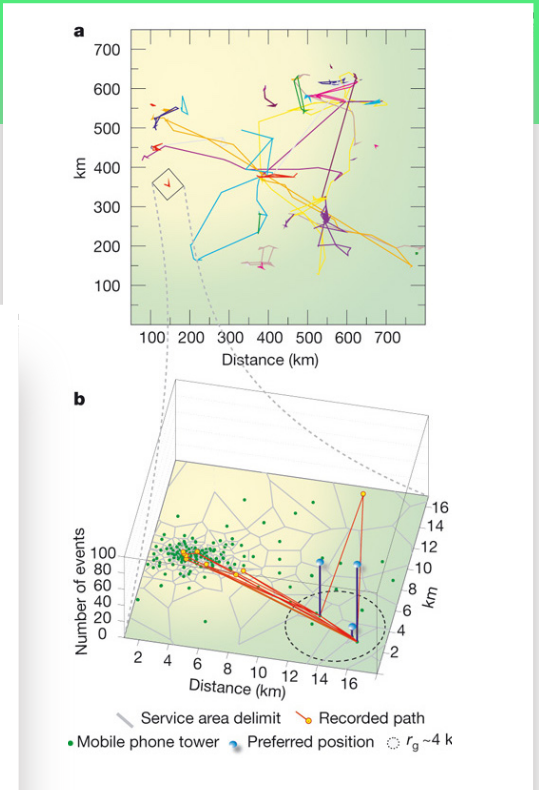
Human mobility

continued (Cell phone data)

Vertex: place

Edge: physical mobility





📘 **Mesoscopic Structure and Social Aspects of Human Mobility**
James P. Bagrow, Yu-Ru Lin
PlosOne 2012.

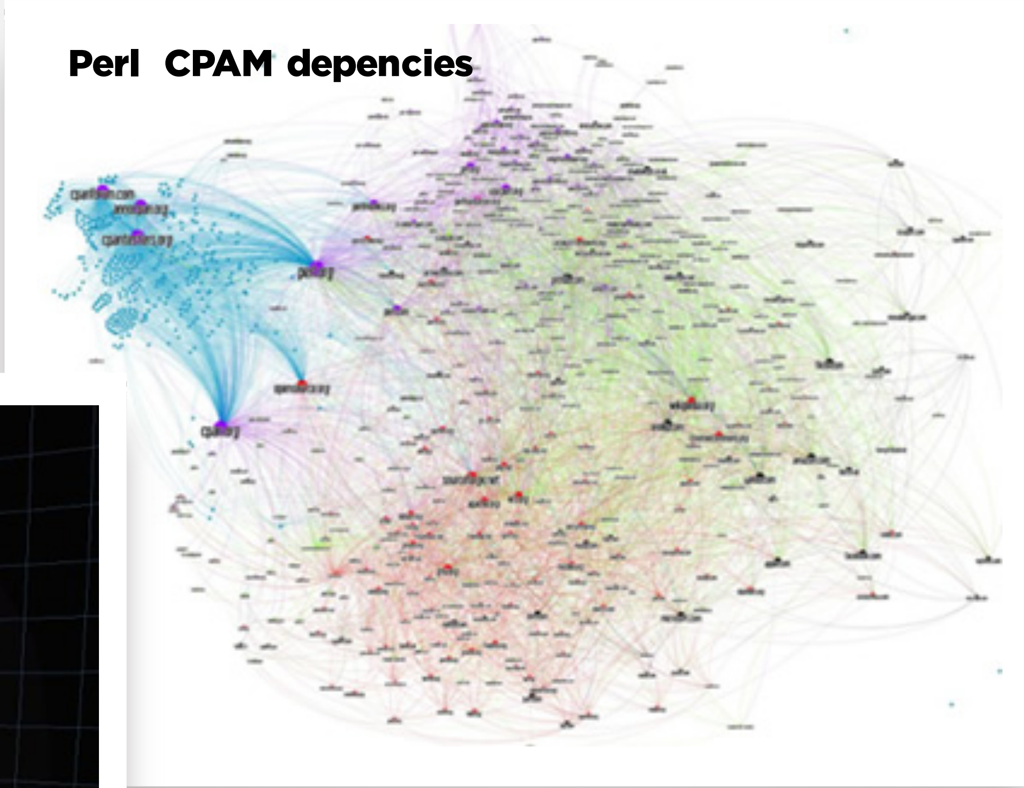
📘 **Understanding individual human mobility patterns**
Marta C. González1 César A. Hidalgo1& Albert-László Barabási
Nature, 2008.

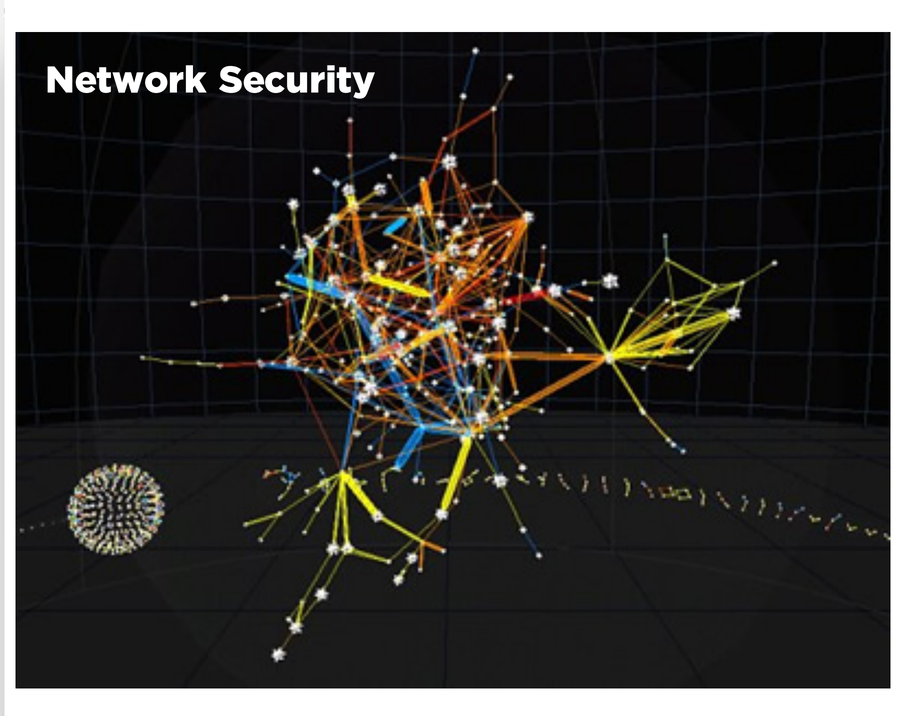
Human Mobility
continued

Computer Systems

vertex: computer, library, code block, etc.

Edges: dependency, attack



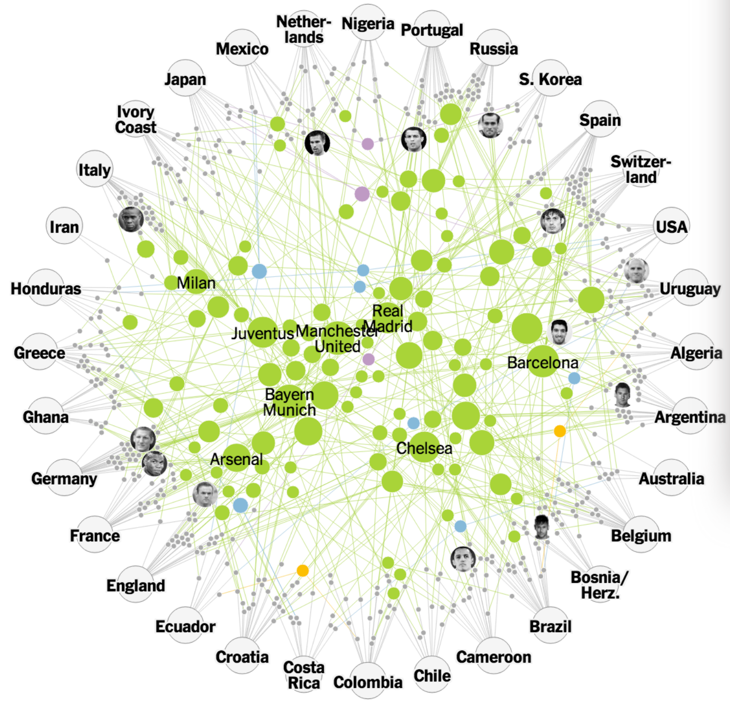


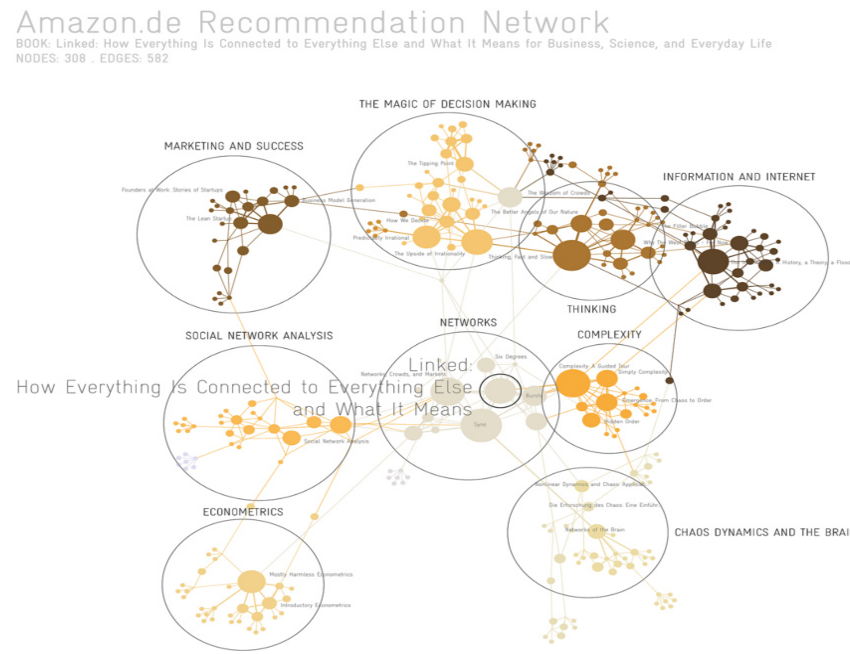
Bipartite Relations

vertex: club, country, customer, books.

edges: player, attack, buying relationship

Football Relatiosnship



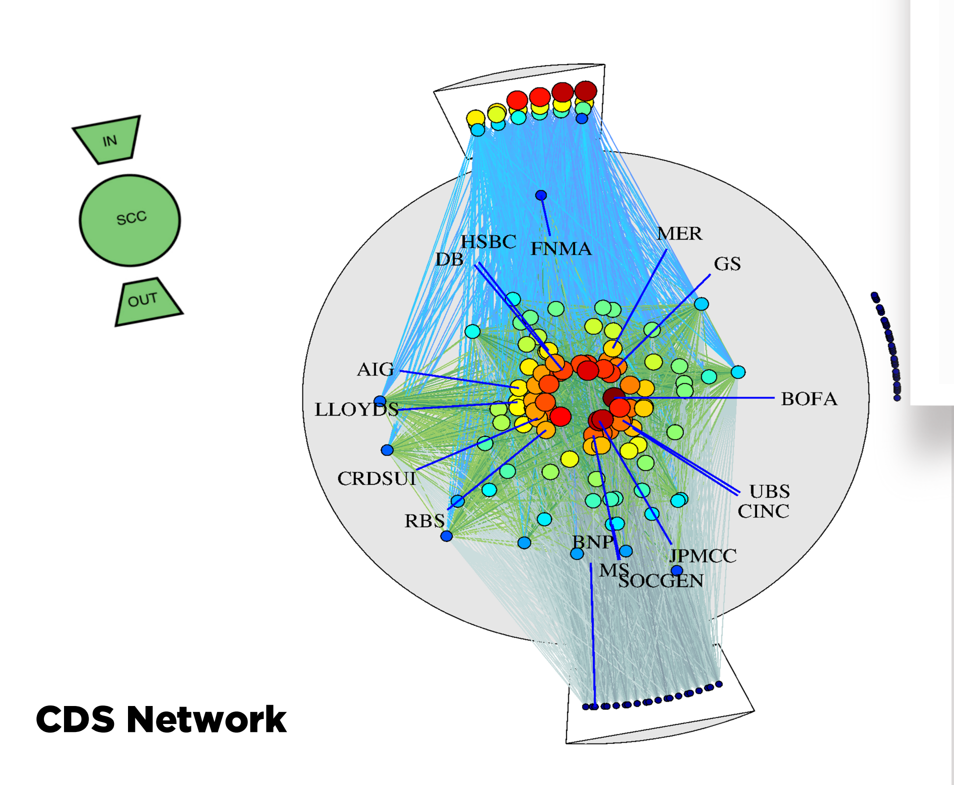


Amazon Recommendation

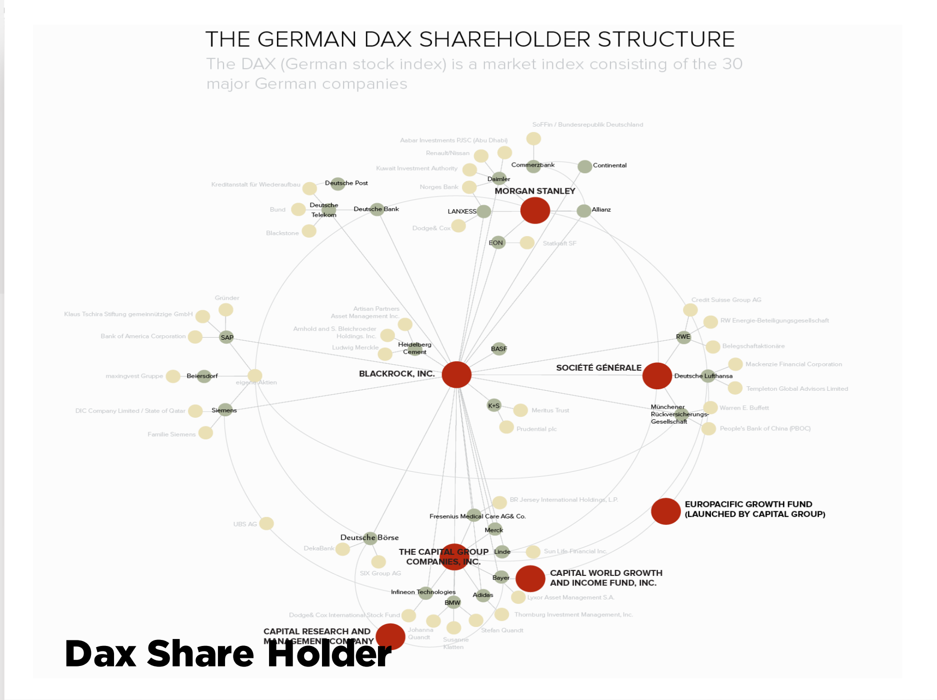
Where goes the money flow

vertex: banks, companies

Edges: share, loan



CDS Network

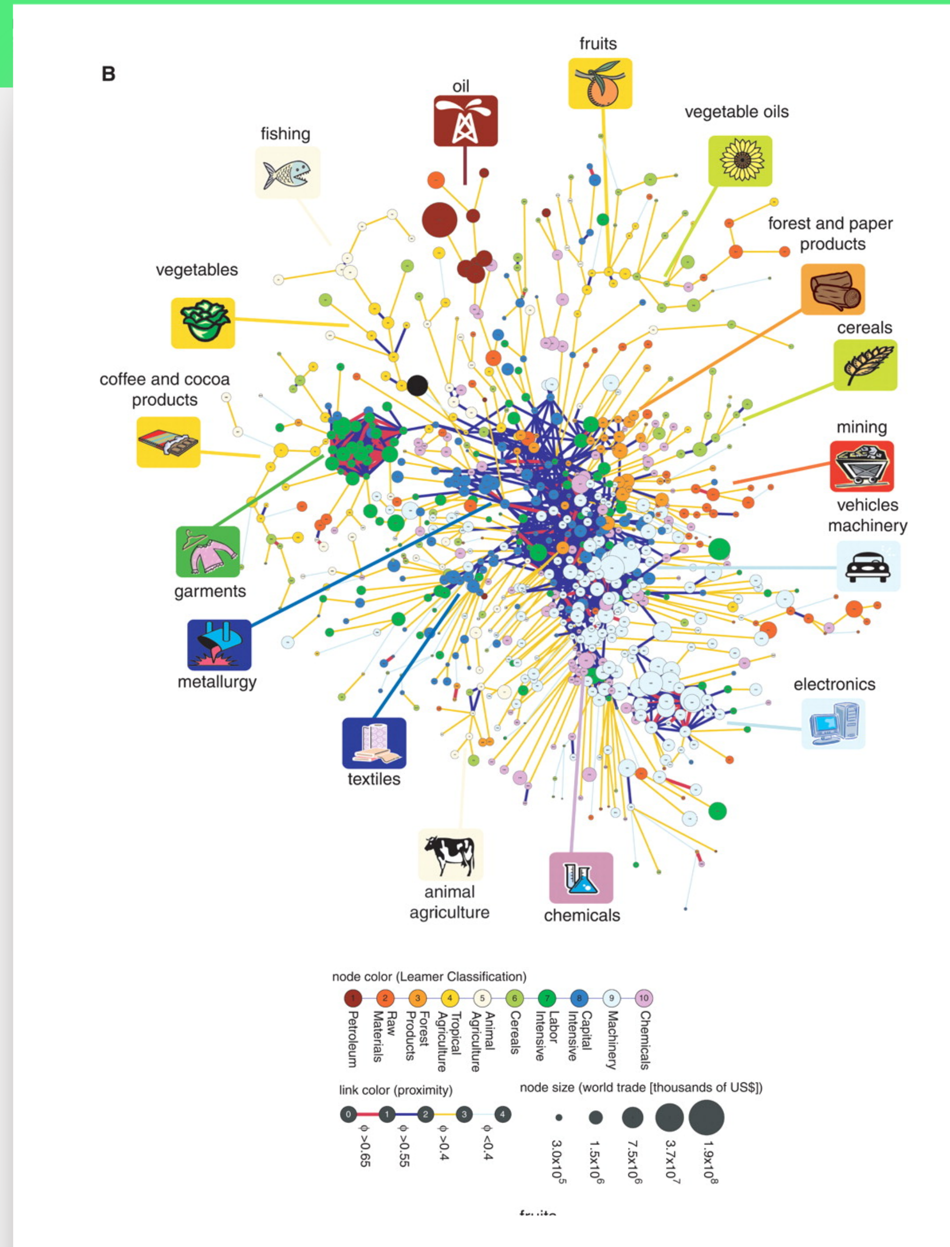


Dax Share Holder

Economy

[Cesar Hidalgo: "Why Information Grows"]

- How do products depend on each other, and how does this network evolve?
- How do countries depend on each other for water, energy, people (immigration), investments?

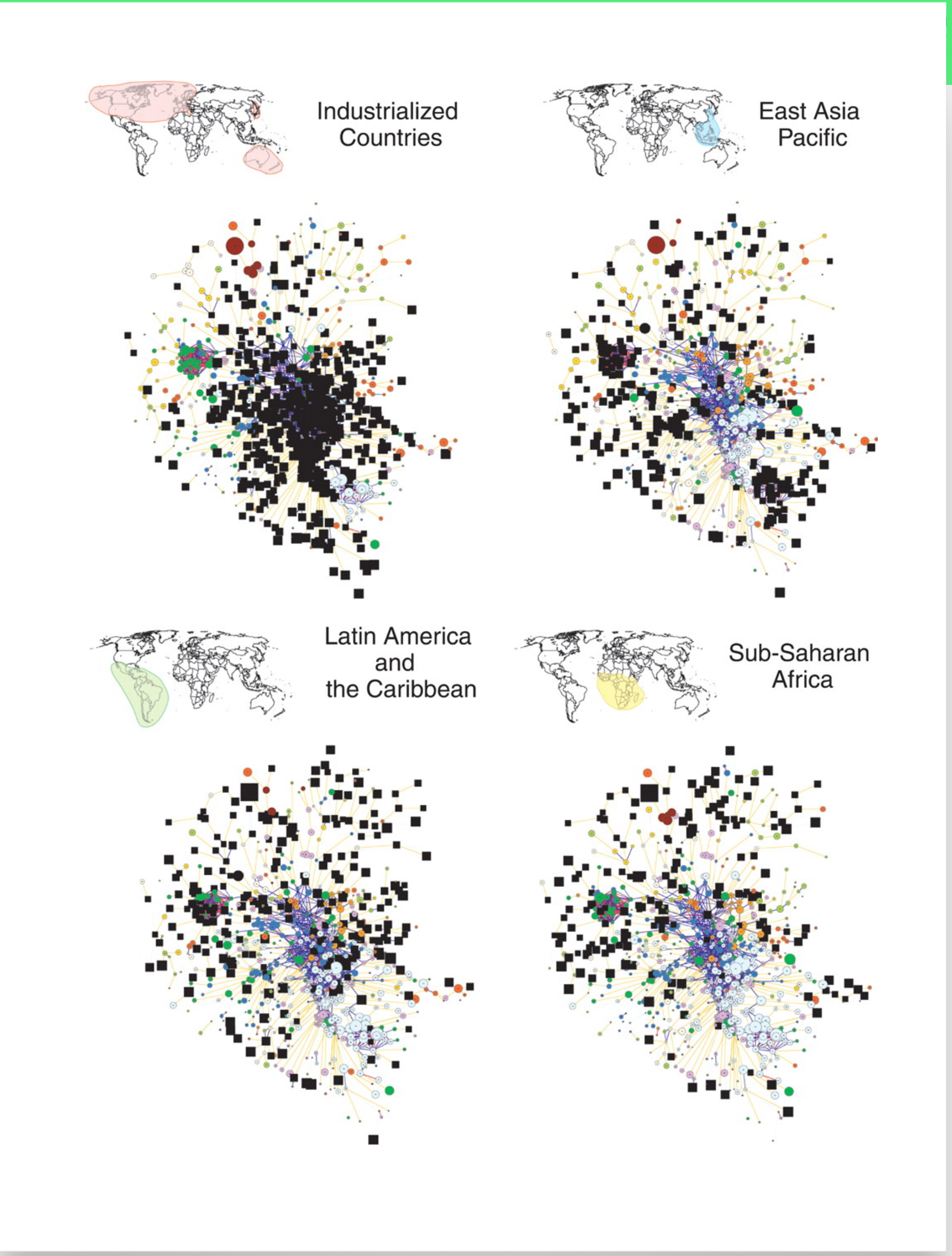


The Product Space Conditions the Development of Nations
C. A. Hidalgo, B. Klinger, A-L. Barabási, and R. Hausmann
Science 27 July 2007: 317 (5837), 482-487

Economy

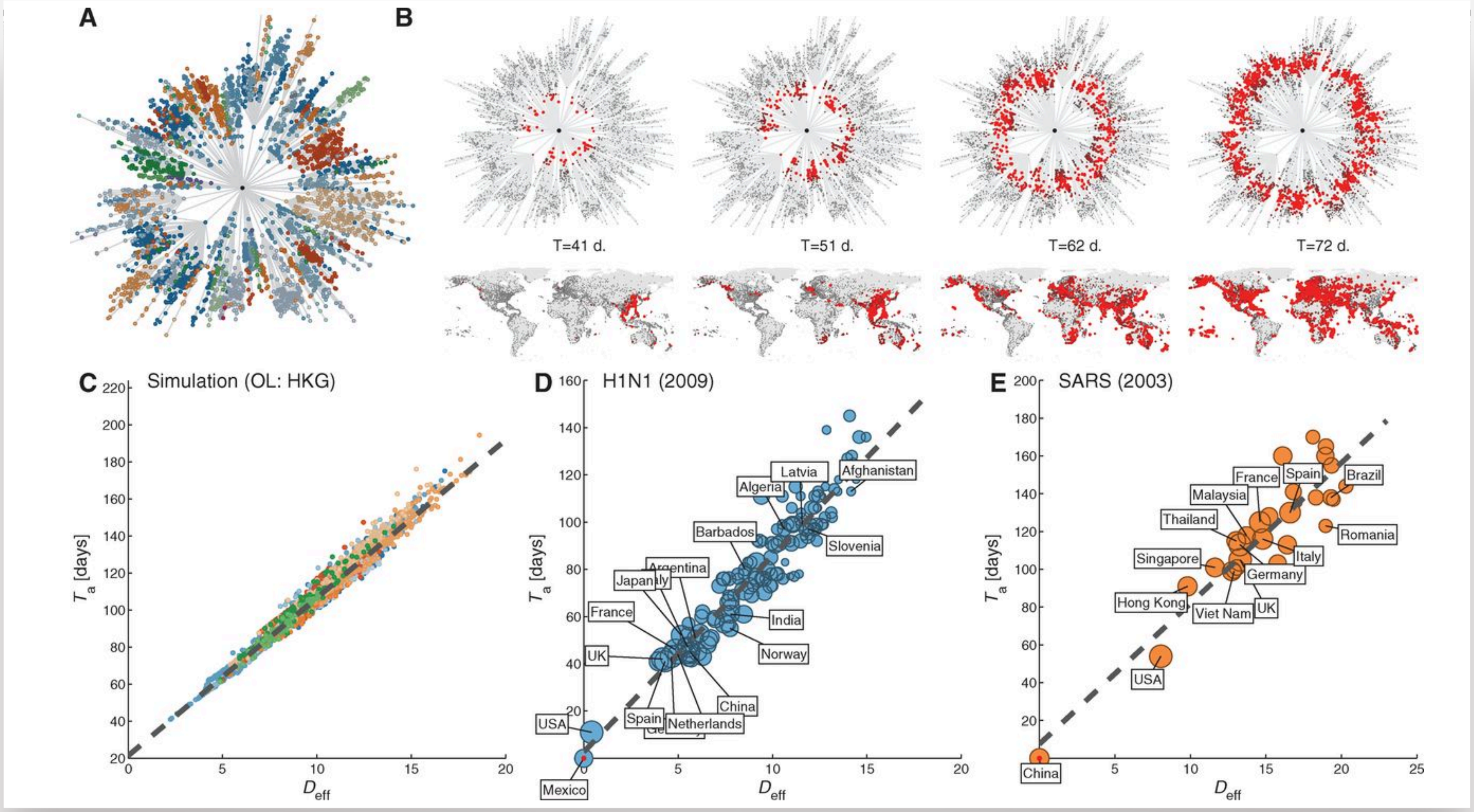
Cesar Hidalgo: "Why Information Grows"

- How do products depend on each other, and how does this network evolve?
- How do countries depend on each other for water, energy, people (immigration), investments?



The Product Space Conditions the Development of Nations
C. A. Hidalgo, B. Klinger, A-L. Barabási, and R. Hausmann
Science 27 July 2007: 317 (5837), 482-487

Diseases Spreading



The Hidden Geometry of Complex, Network-Driven Contagion Phenomena
Dirk Brockmann, and Dirk Helbing
Science 2013

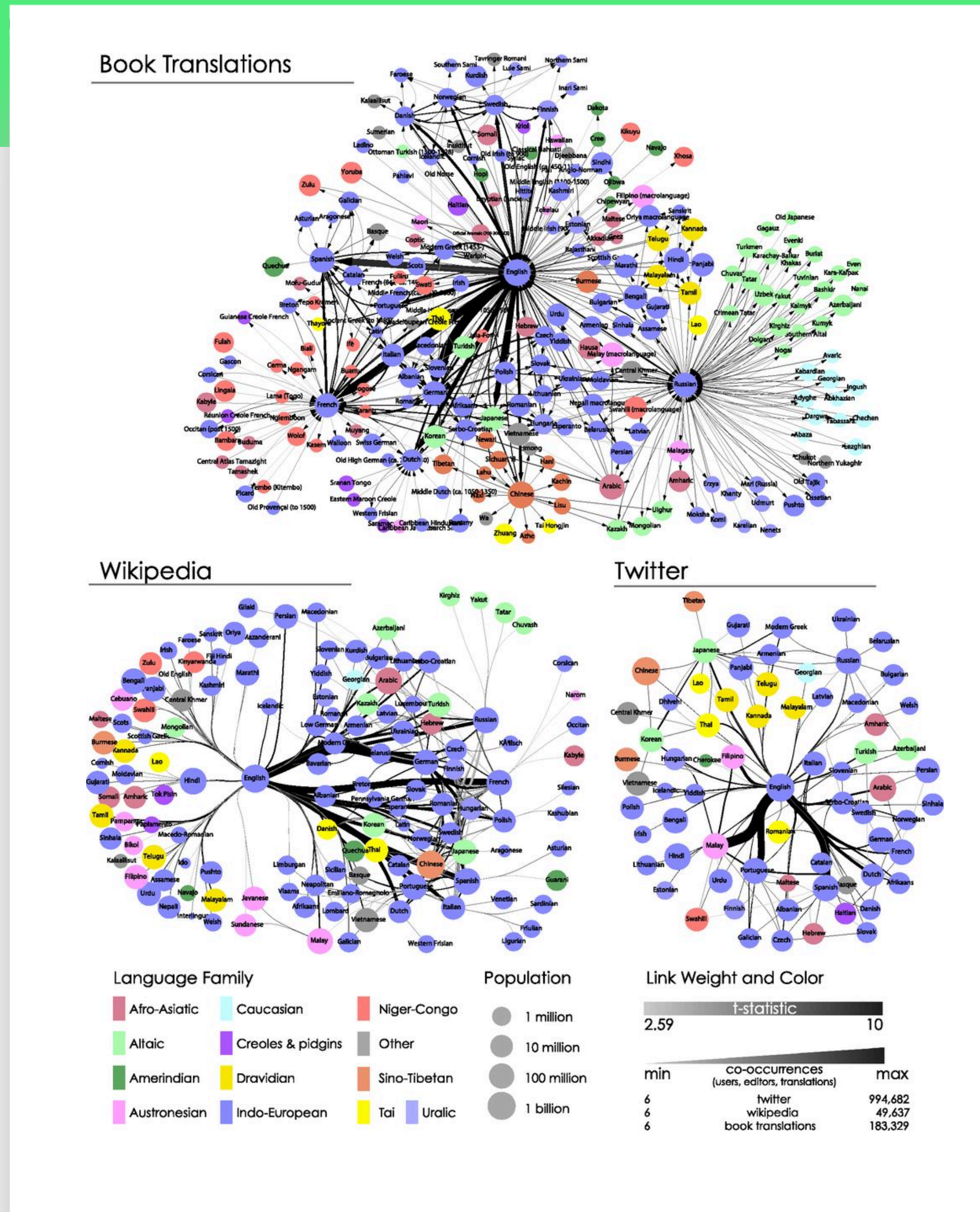
Vertex: Susceptible Population
Edges: Airline, transport network

Diseases Spreading

Continued

Languages

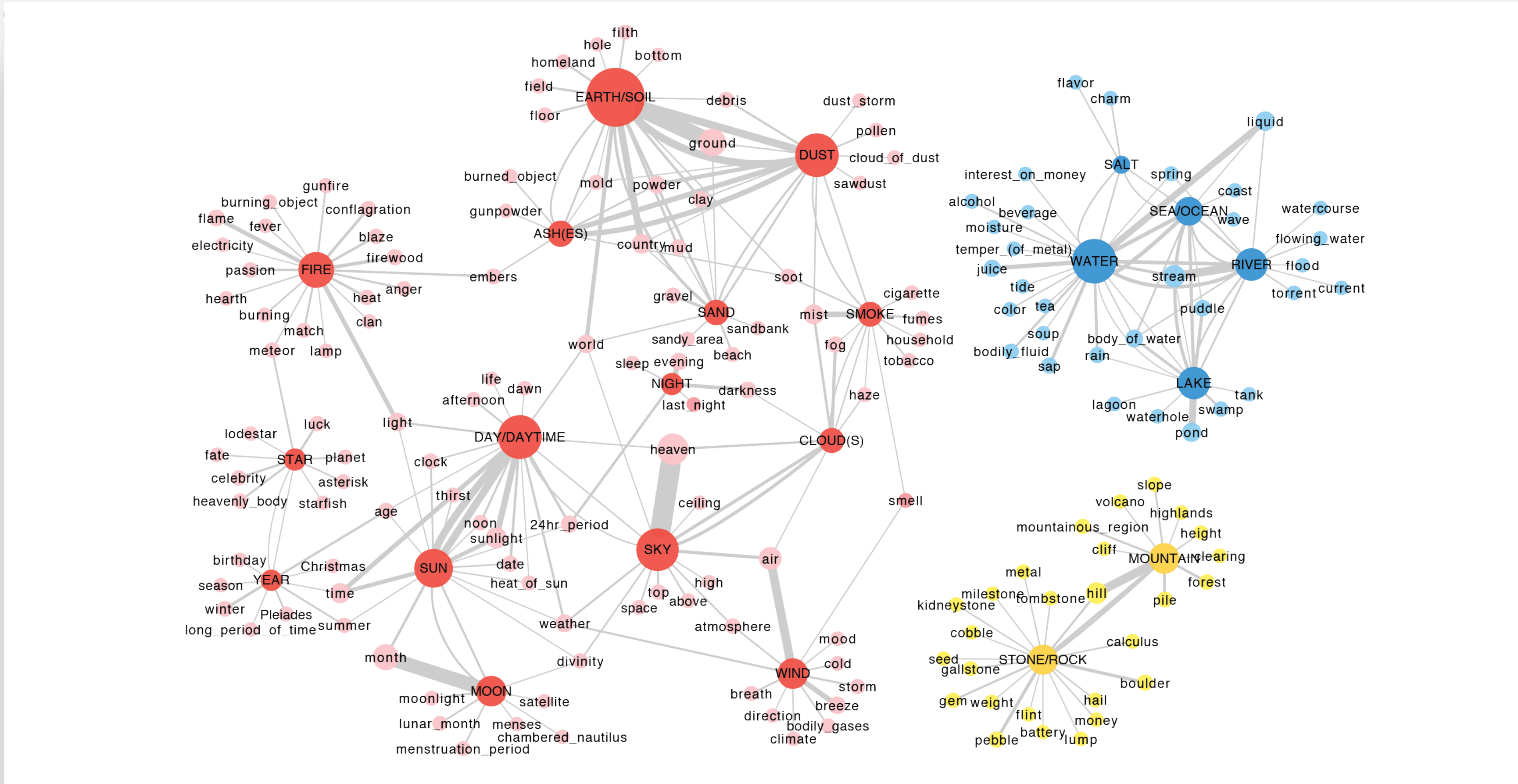
vertex: Langage
edges: translator, Wikipedia author, Twitter user



📘 Links that speak: the global language network and its association with global fame
Ronen et al
PNAS 2014.

Languages

human lexical semantics



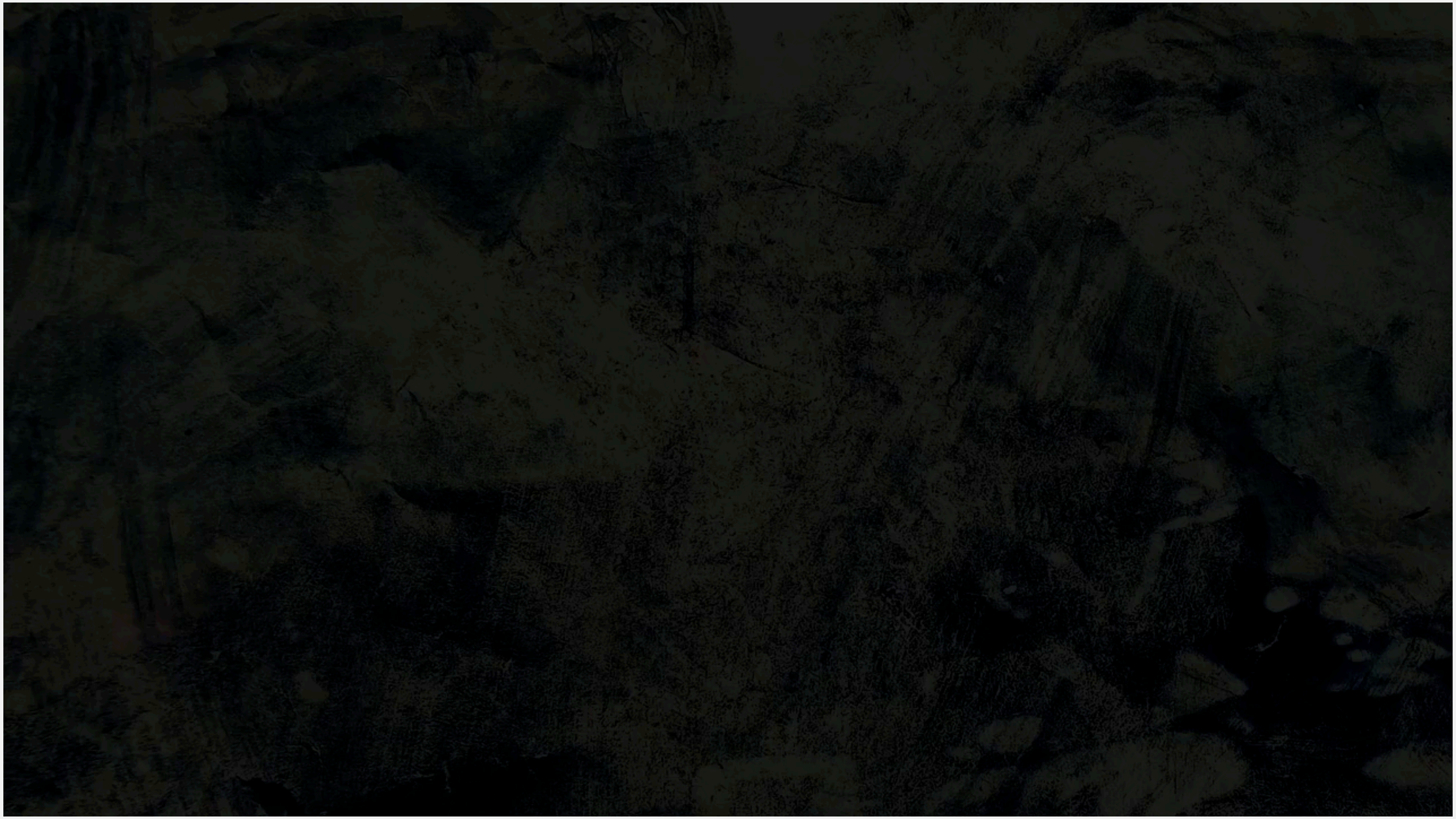
vertex: words

edges: semantics relationship

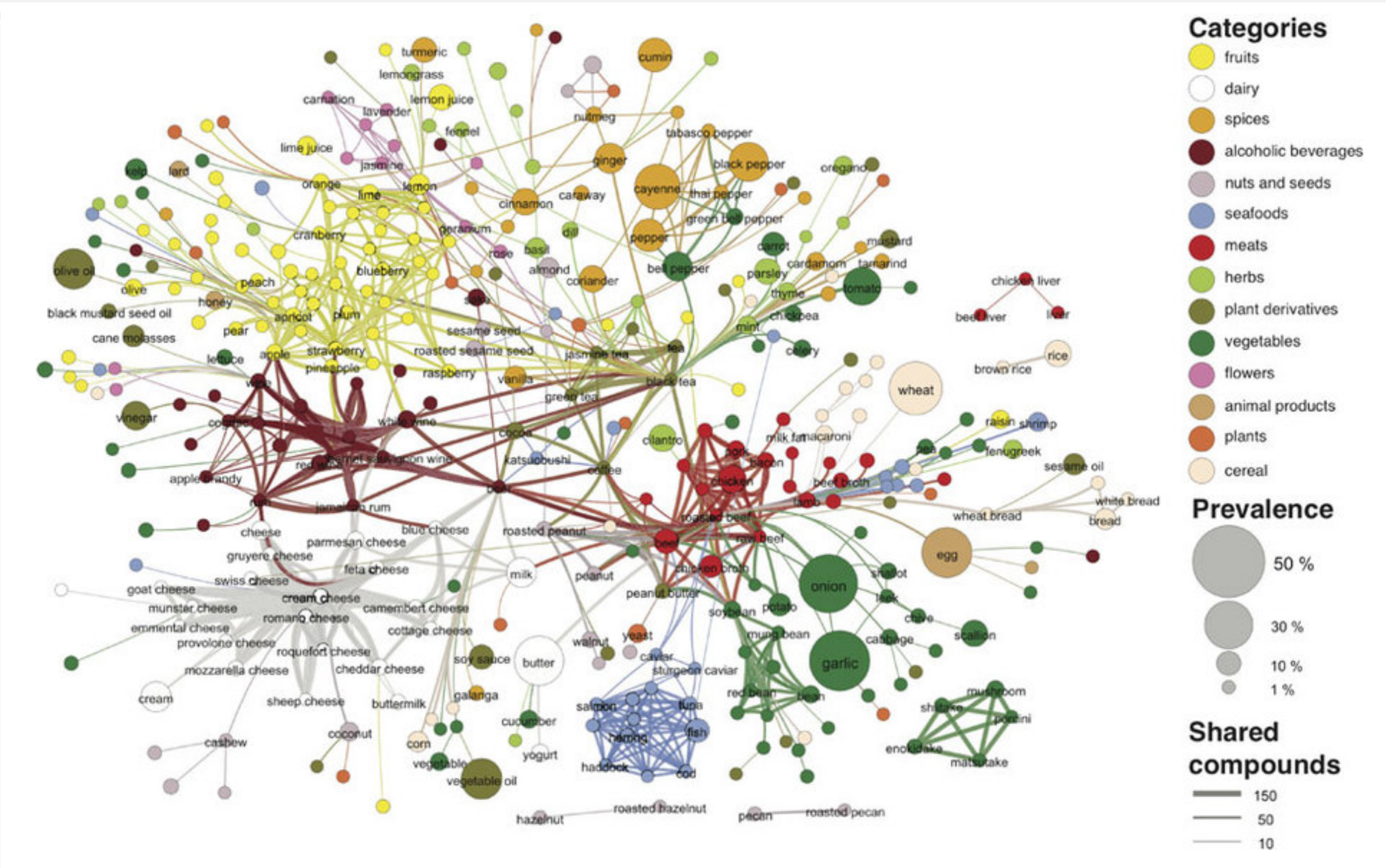
On the universal structure of human lexical semantics
Youn, et al
PNAS 2016.

Languages

human lexical semantics (continued)



Cooking



vertex:

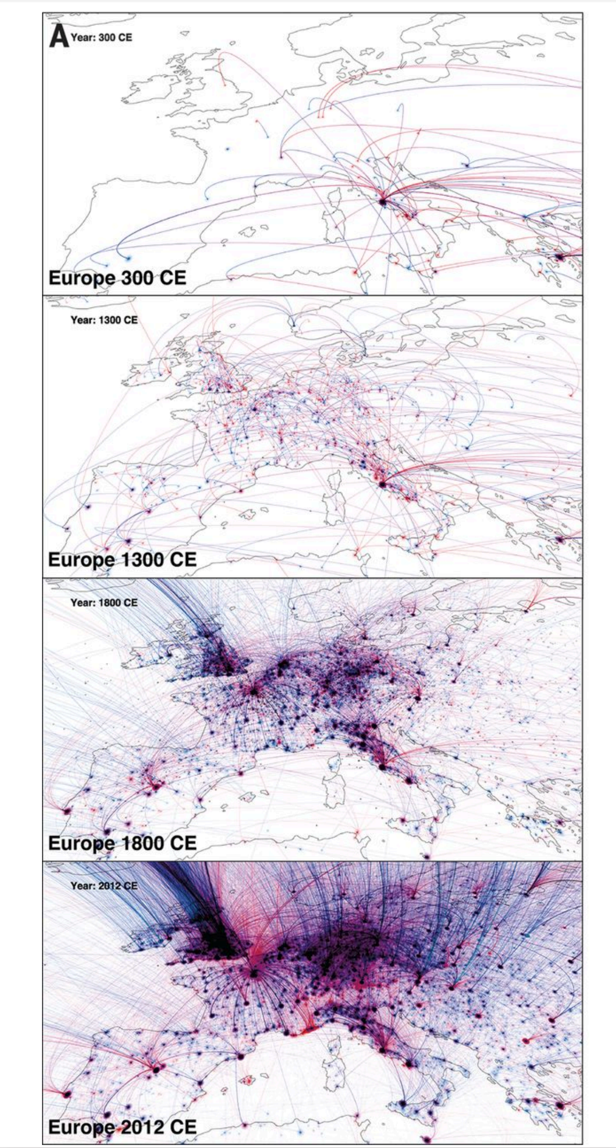
Two ingredients are connected if they share a significant number of flavor

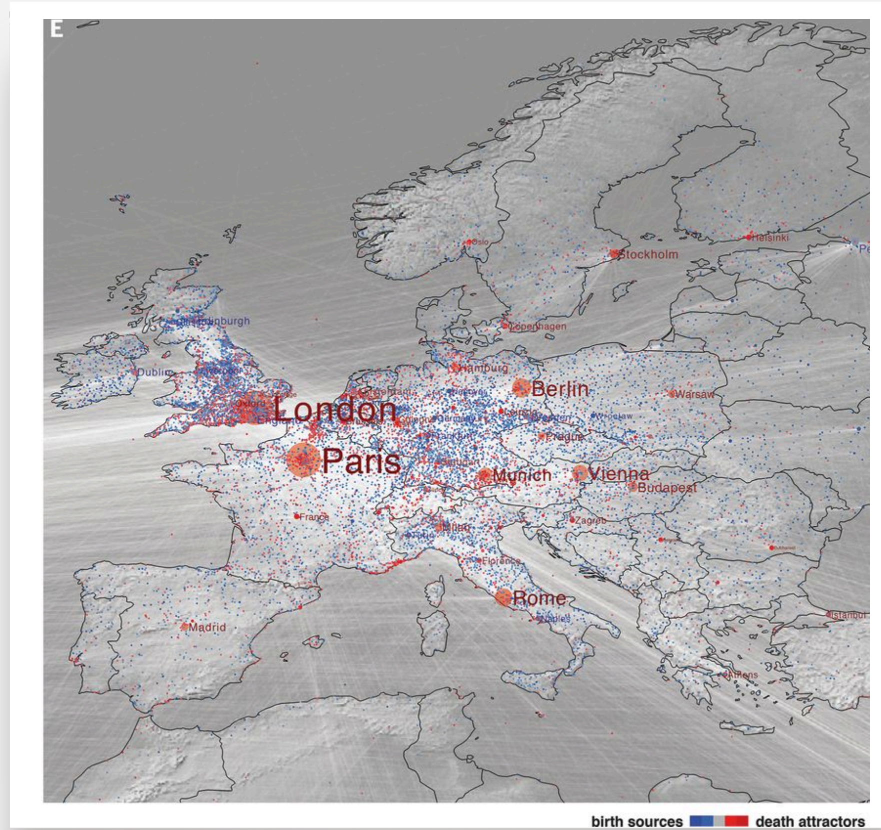
edges:

denotes an ingredient

Flavor network and the principles of food pairing
Yong-Yeol Ahn, Sebastian E. Ahnert, James P. Bagrow & Albert-László Barabási
Scientific Reports, 196 (2011)

Cultural Spreading





vertex: country

edges: immigration, person

A network framework of cultural history
Maximilian Schich et al
Science (2014) Vol. 345, Issue 6196, pp. 558-562

Cultural Spreading

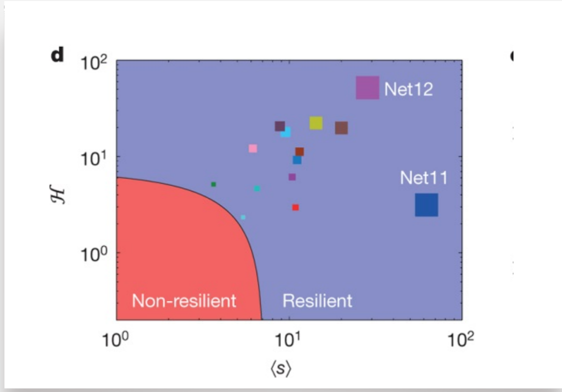
Continued

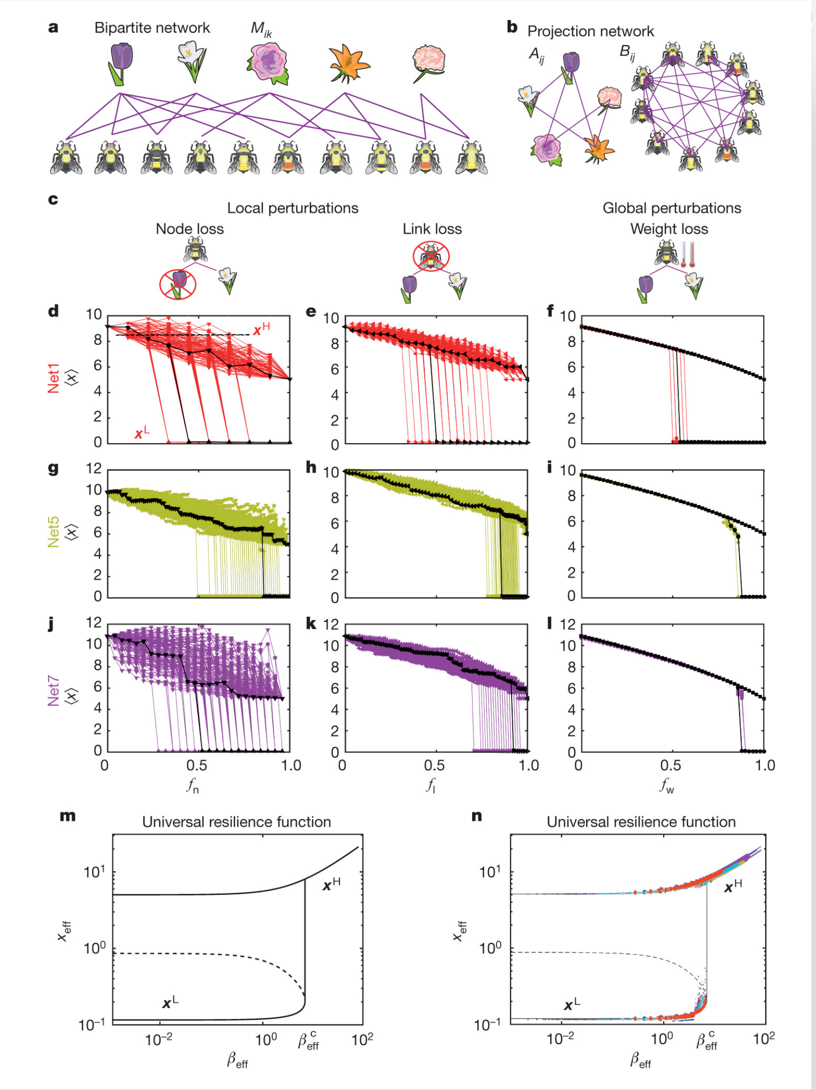
Resilience in ecological networks

vertex: species

edges: dependency relationship, symbiotic relationship:

- food
- shelter





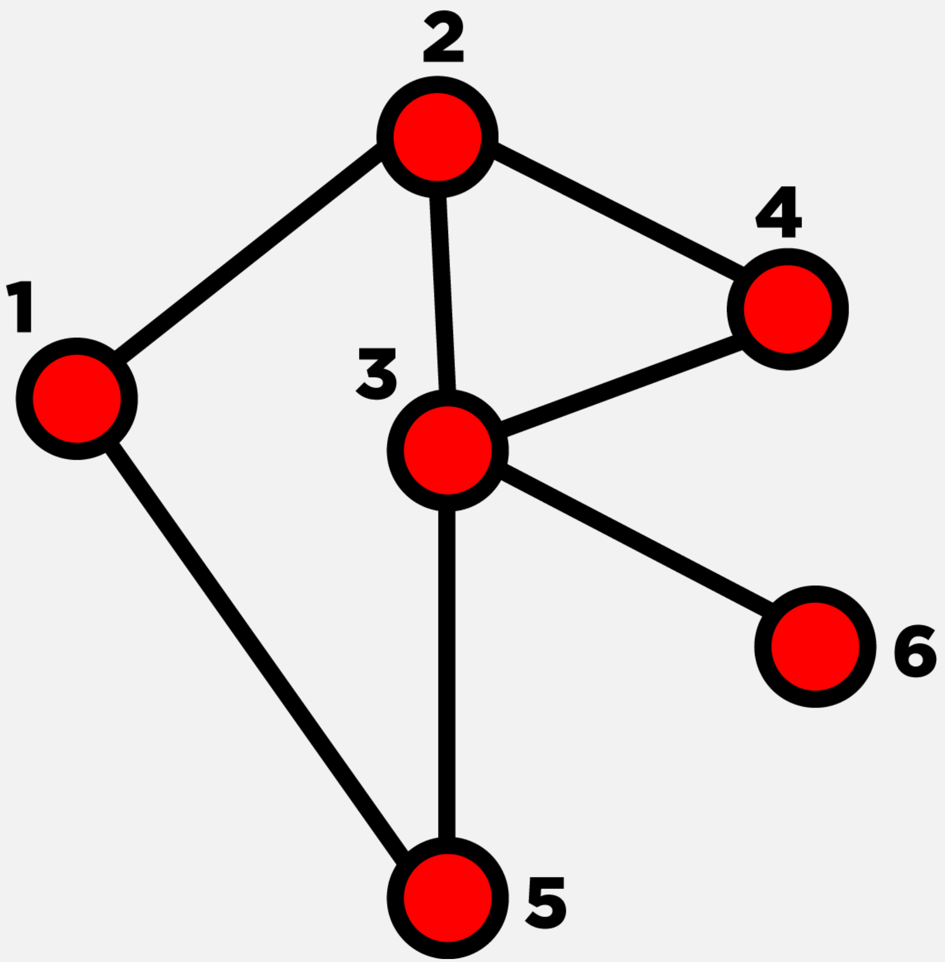
Universal resilience patterns in complex networks
Jianxi Gao, Baruch Barzel & Albert-László Barabási
Nature 530, 307–312, 2015

Resilience in ecological networks

continued

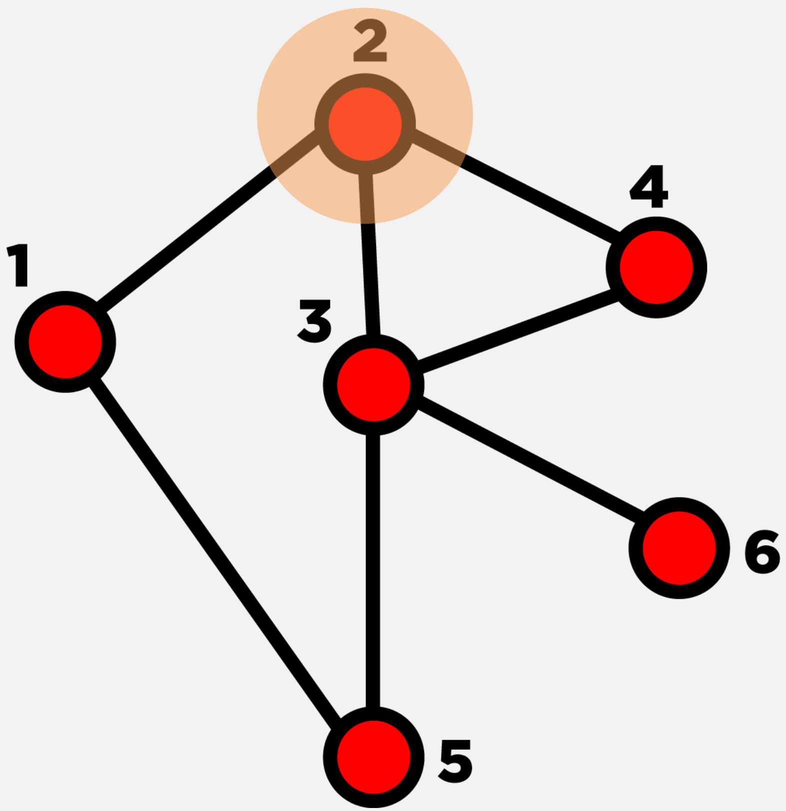
BASICS OF GRAPH THEORY

Network representation



Undirected
Unweighted
No self-loop

Simple network



Undirected
Unweighted
No self-loop

Adjacency Matrix

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

Adjacency List

A

$1 \rightarrow \{2, 5\}$

$2 \rightarrow \{1, 3, 4\}$

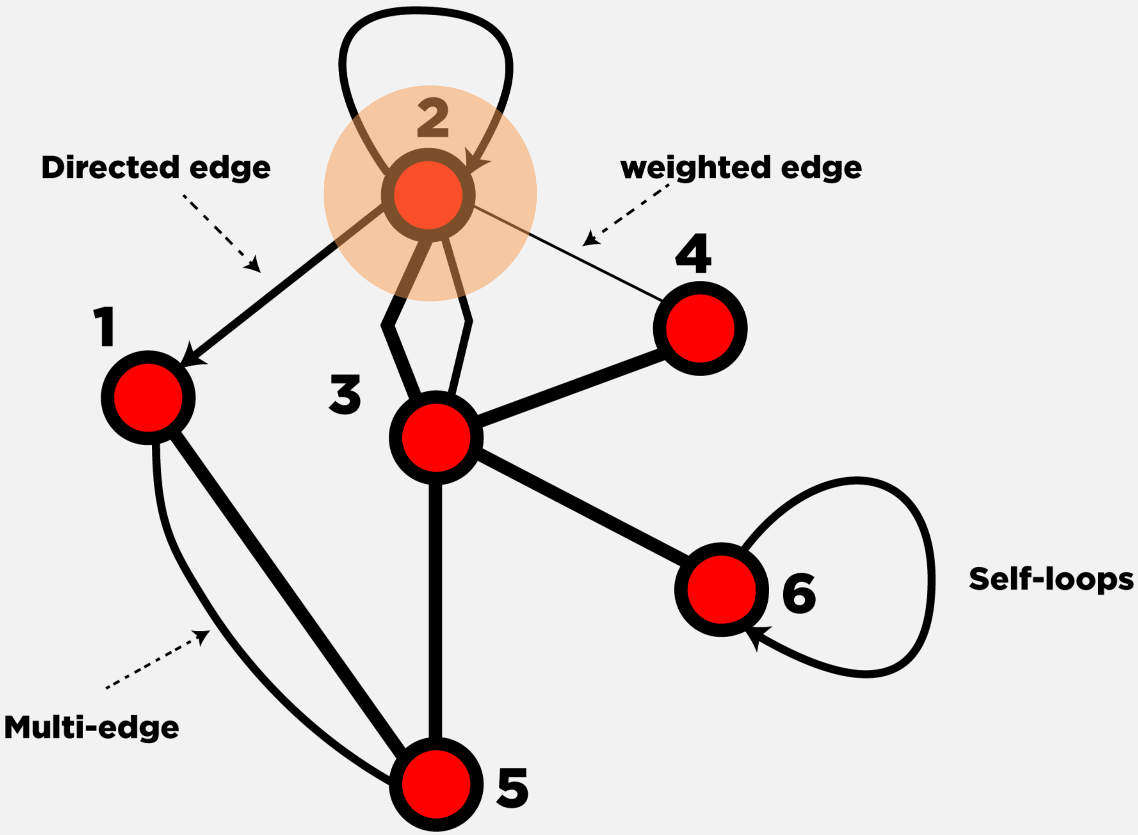
$3 \rightarrow \{2, 4, 5, 6\}$

$4 \rightarrow \{2, 3\}$

$5 \rightarrow \{1, 3\}$

$6 \rightarrow \{3\}$

More complex network



~~Undirected~~
~~Unweighted~~
~~No self-loop~~

Adjacency Matrix

A	1	2	3	4	5	6
1	0	0	0	0	$\{1,2\}$	0
2	1	$\frac{1}{2}$	$\{2,1\}$	1	0	0
3	0	$\{2,1\}$	0	4	4	4
4	0	1	4	0	0	0
5	$\{2,1\}$	0	4	0	0	0
6	0	0	4	0	0	1

Adjacency List

A

$1 \to \{(5,2),(5,1)\}$

$2 \to \{(1,1),(2,\frac{1}{2}),(3,2),(3,1),(4,1)\}$

$3 \to \{(2,1),(2,2),(4,4),(5,4),(6,4)\}$

$4 \to \{(2,1),(3,4)\}$

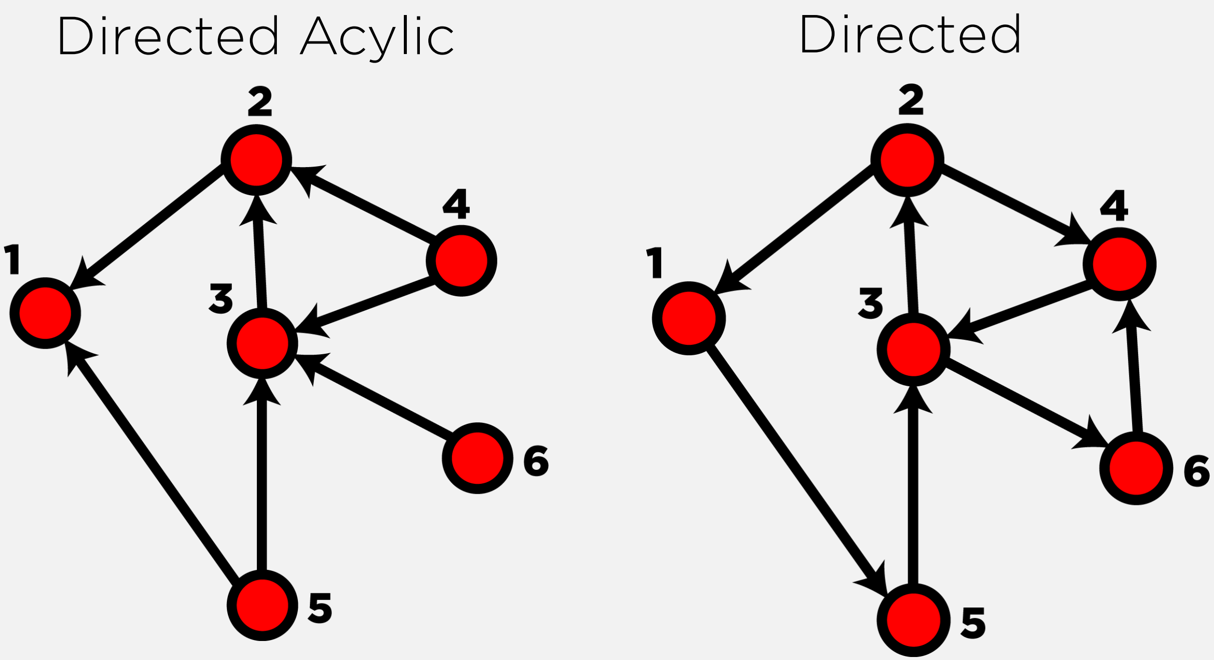
$5 \to \{(1,4),(1,1),(3,4)\}$

$6 \to \{(3,4),(6,1)\}$

Directed Network

citations

foodweb



www

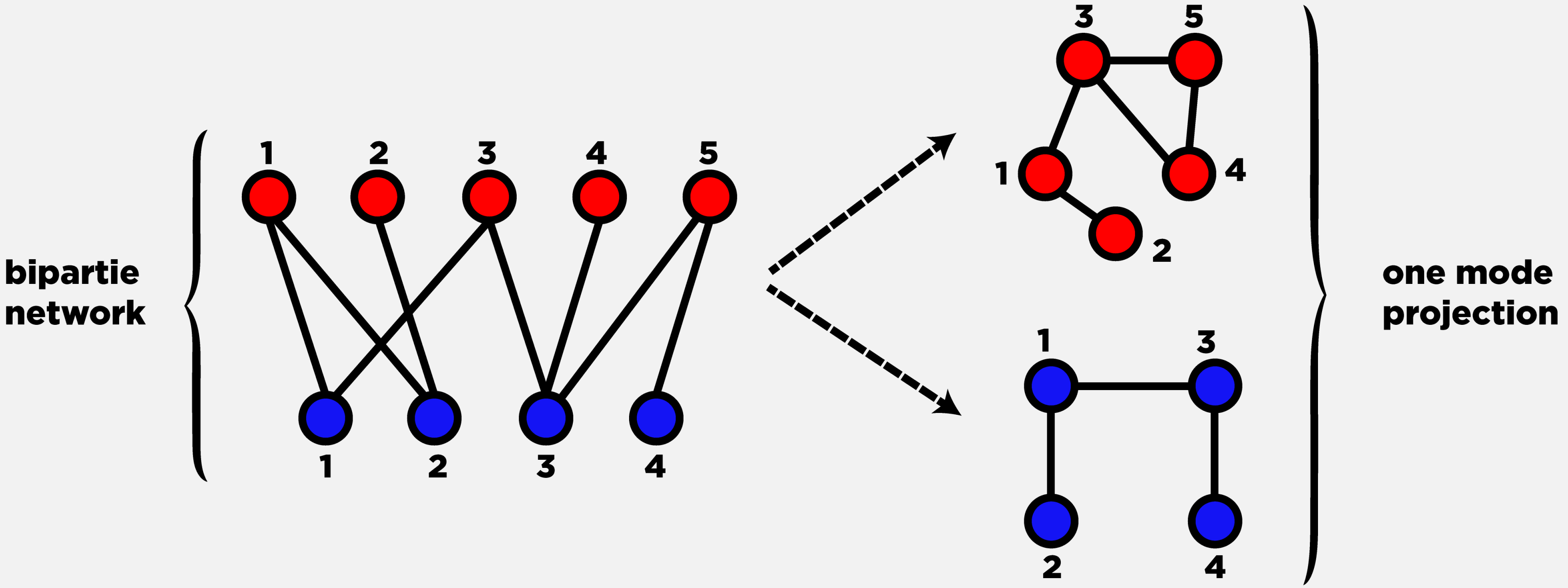
economic exchange

Information

Friendship

$$\mathbf{A_{ij}} \neq \mathbf{A_{ji}}$$

Bipartie Network

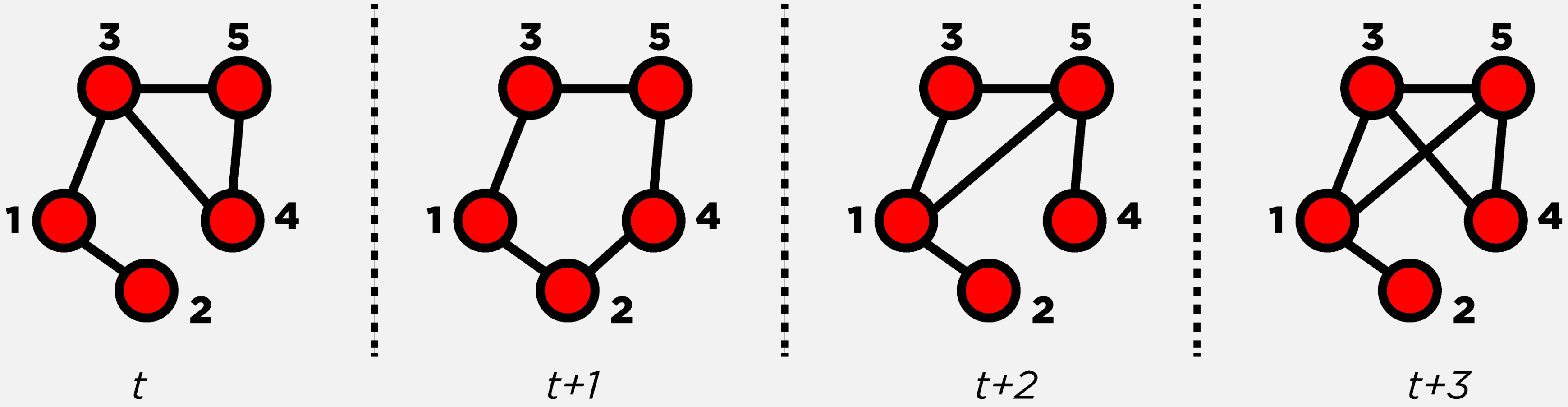


author network: co-citation network

actor Network: Kavin Bacon network

customer relationship: Amazon

Temporal Network



Discrete time (snapshot), edges (i, j, t)

Continus time, edges $(i, j, t_s, \Delta t)$

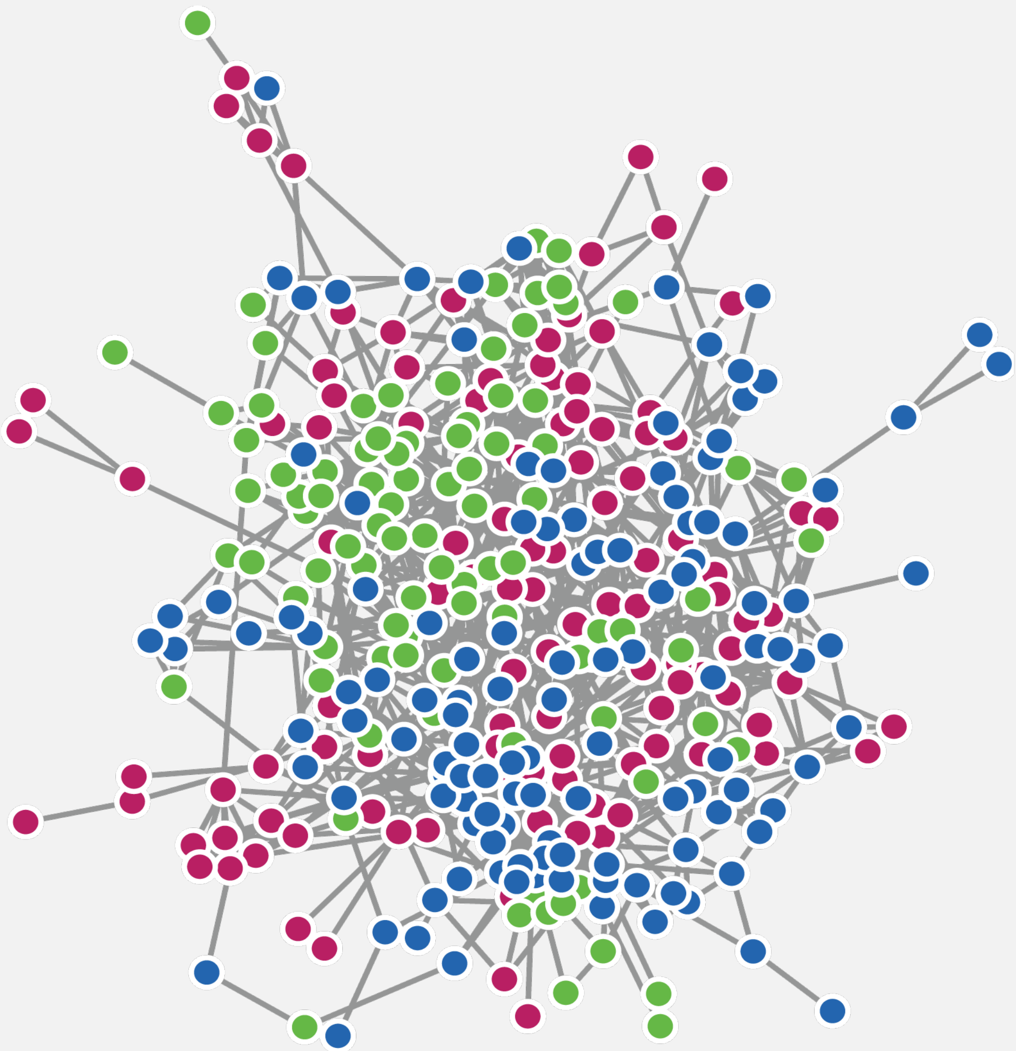
HOW TO DESCRIBE A NETWORK

Describing a Network

What network look like

questions:
- **How edges are organized?**
- **how do vertices differ?**
  does network location matter?
- **are there underlying patterns?**

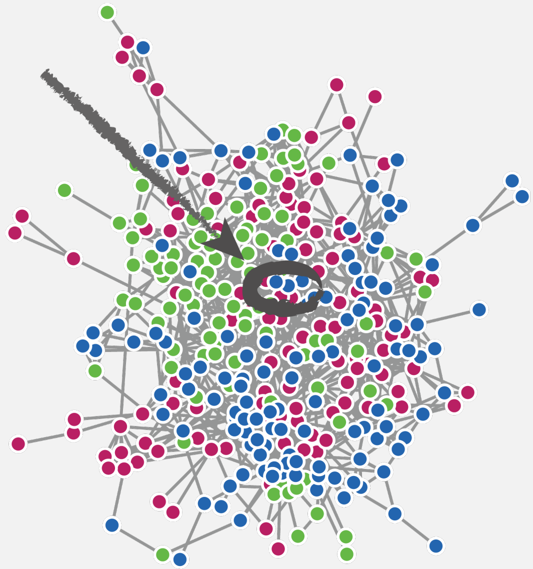
what we want to know:
- **what processes shape these networks?**
- **how can we tell?**

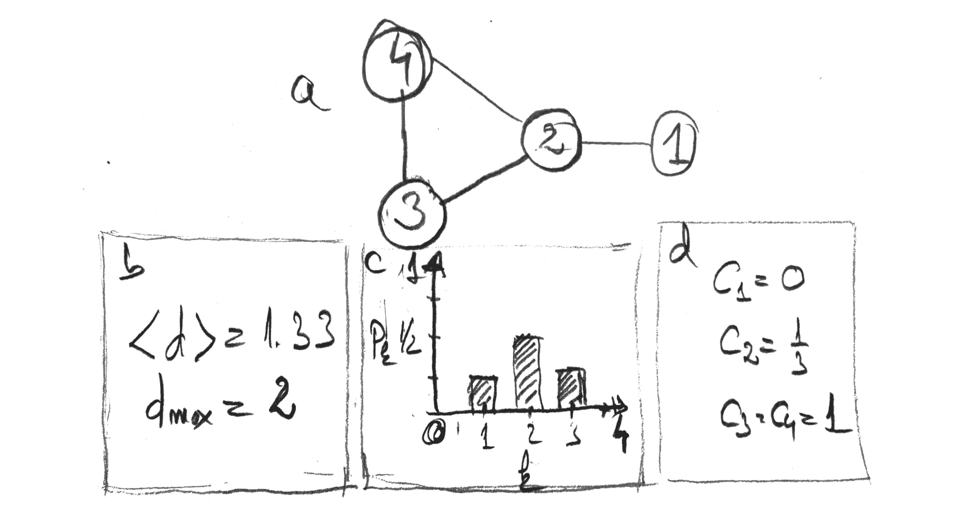


Describing a Network

Four central quantities:

1. **degree distributions**
2. **short-loop density (triangles, etc.)**
   - **Clustering**
3. **shortest paths (diameter, etc.)**
4. **vertex positions,**
5. **correlations between these**

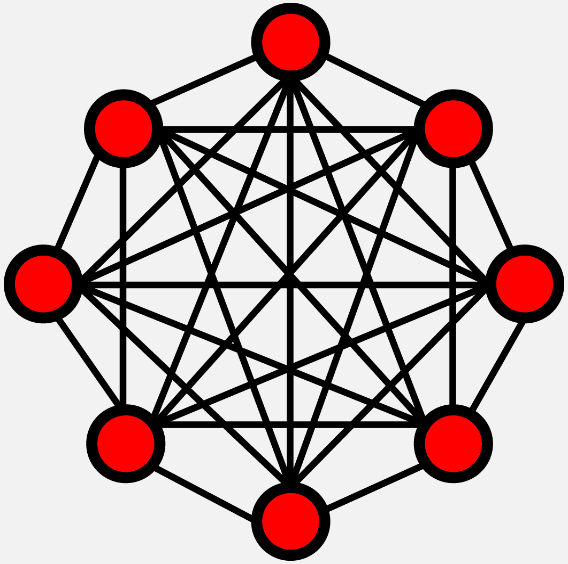




Complete Graph

The maximum number of links a network of N nodes can have is:

$$L_{max} = \binom{N}{2} = \frac{N(N-1)}{2}$$



A graph with degree $L=L_{max}$ is called a complete graph, and its average degree is $\langle k \rangle = N-1$

Real networks are sparse

Most networks observed in real systems are sparse:

$$L \ll L_{max}$$

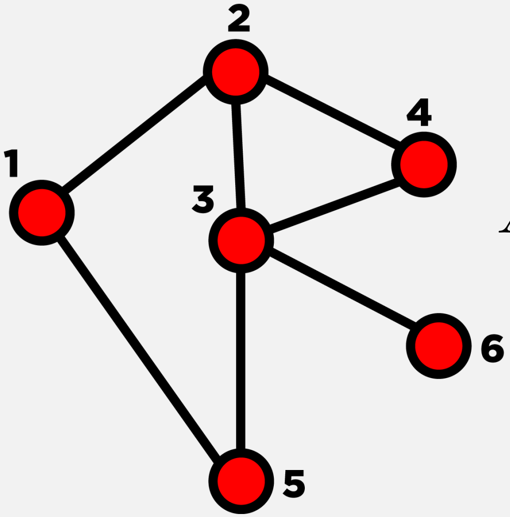
or

$$\langle k \rangle \ll N-1$$

WWW	**N=325, 729**	**L=1.4 10^6**	**Lmax=10^{12}**	**$\langle k \rangle$=4.51**
Internet, router	**N=192 224**	**L=636 643**	**Lmax=18 10^9**	**$\langle k \rangle$=6.34**
Coauthorship (Math)	**N= 70, 975**	**L=2 10^5**	**Lmax=3 10^{10}**	**$\langle k \rangle$=3.9**
Movie Actors	**N=212 ,250**	**L=6 10^6**	**Lmax=1.8 10^{13}**	**$\langle k \rangle$=28.78**

(Source: Albert, Barabasi, 2002)

Node Degree



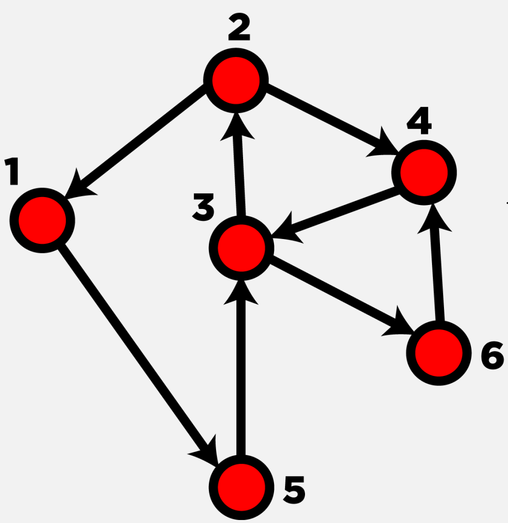
$$A_{ij} = \begin{pmatrix} 0 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{pmatrix}$$

$$A_{ij} = A_{ji}$$

$$A_{ii} = 0$$

$$k_i = \sum_{j=1}^{N} A_{ij}$$

$$k_j = \sum_{i=1}^{N} A_{ij}$$



$$A_{ij} = \begin{pmatrix} 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix}$$

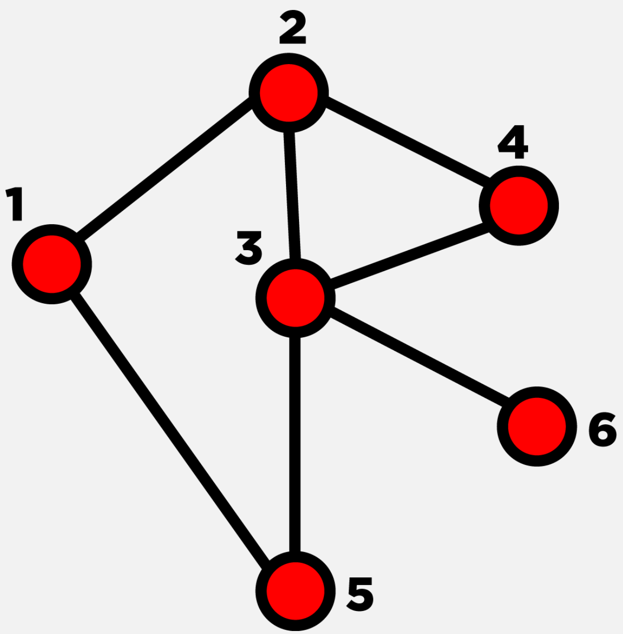
$$A_{ij} \neq A_{ji}$$

$$A_{ii} = 0$$

$$k_i^{in} = \sum_{j=1}^{N} A_{ij}$$

$$k_j^{out} = \sum_{i=1}^{N} A_{ij}$$

Mean Degree



Degree:

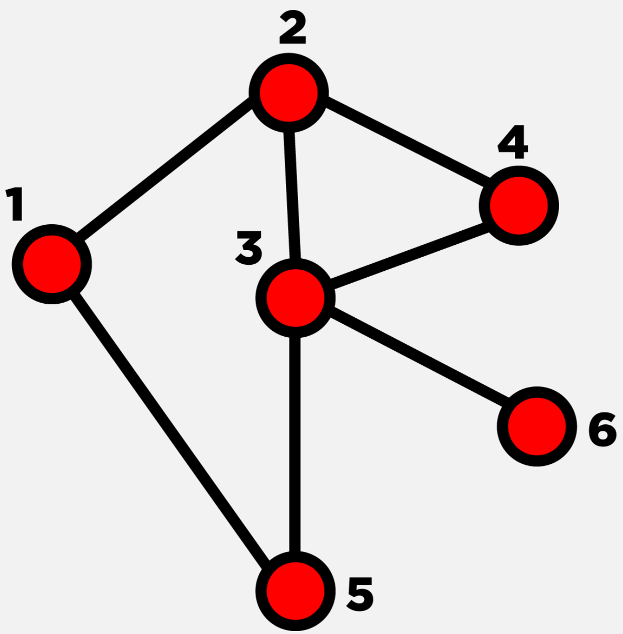
number of connexions k

$$k_i = \sum_j A_{ij}$$

number of edges $$m = \frac{1}{2}\sum_i k_i = \frac{1}{2}\sum_i\sum_j A_{ij}$$

mean degree $$\langle k \rangle = \frac{1}{N}\sum_i k_i = \frac{2m}{N}$$

Degree Distribution



Degree:

number of connexions k

$$k_i = \sum_j A_{ij}$$

degree sequence $\{2, 3, 4, 2, 2, 1\}$

degree distribution $P(k) = \left[(1, \frac{1}{6}), (2, \frac{3}{6}), (3, \frac{1}{6}), (4, \frac{1}{6})\right]$

Degree distribution

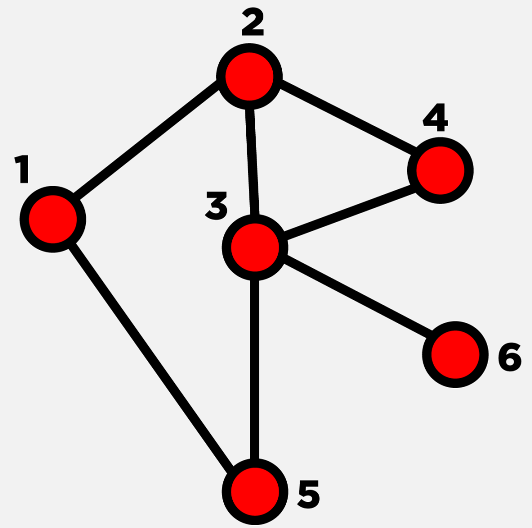
Degree distribution P_k

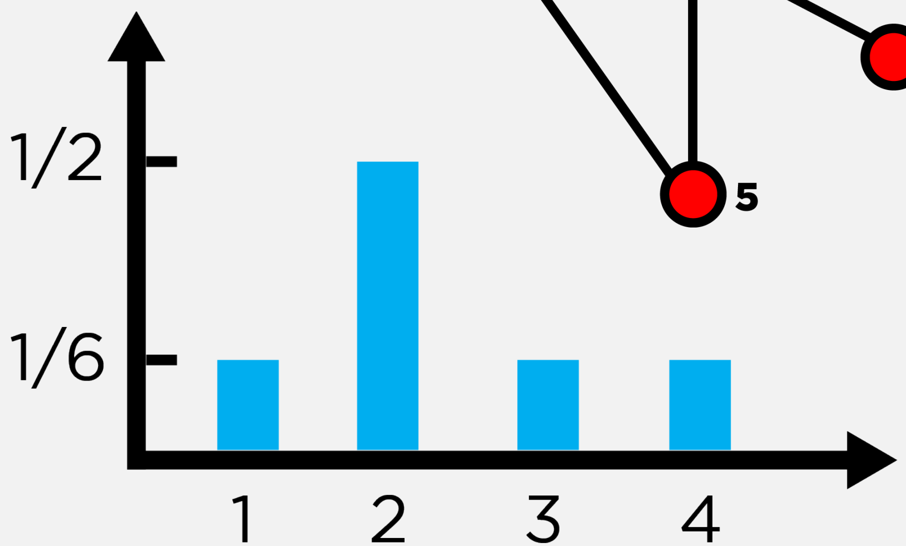
Probability that a randomly chosen vertex has degree k.

N_k : # of node with degree k

P_k : N_k / N

k	N_k	P_k
1	1	1/6
2	3	1/2
3	1	1/6
4	1	1/6



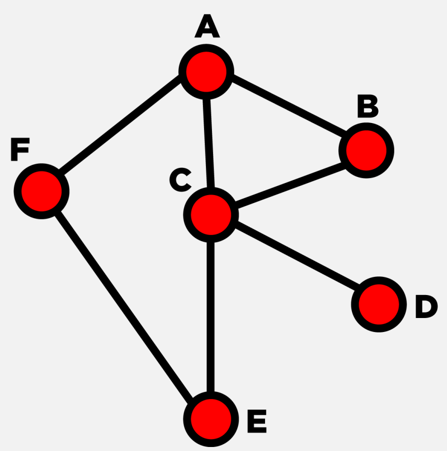


Paths

A path *is* a sequence of nodes in which each node is adjacent to the next one.

$P_{A,B}$ of length n between nodes A and B is an ordered collection of $n+1$ nodes and n links.

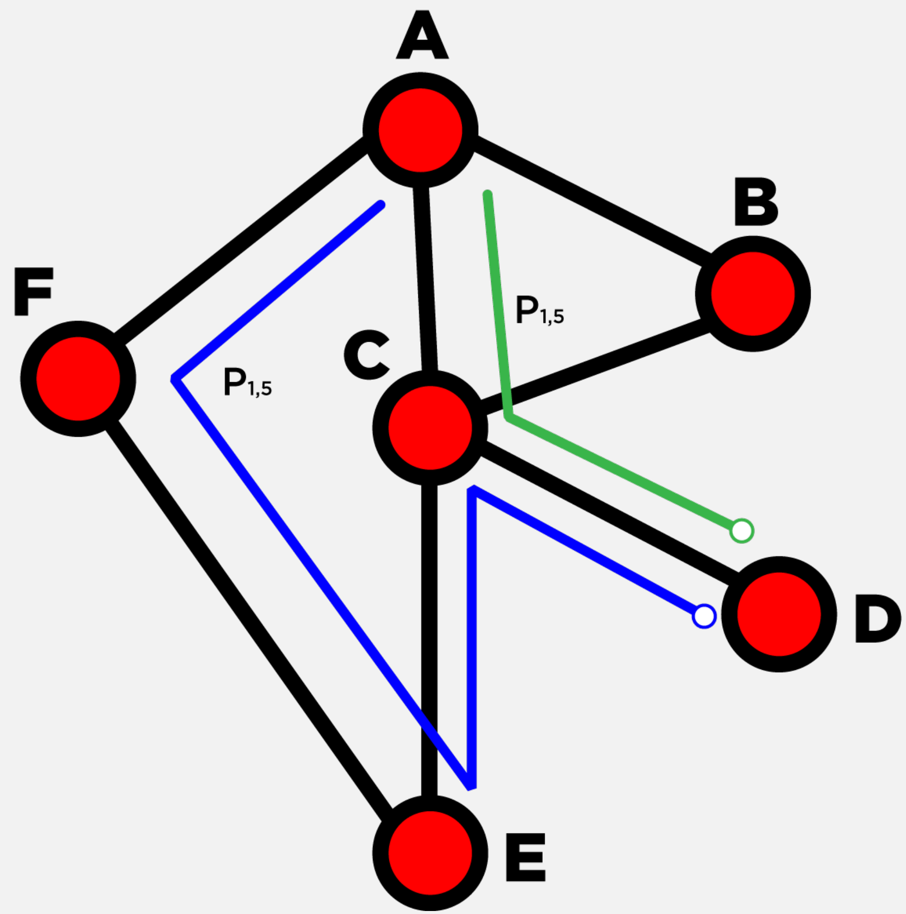
$$G = (V, E)$$
$$where:$$
$$V = \{A, B, C, D, E, F\}$$
$$E = \{(A,B),(A,C),(C,B),(C,D),(C,E),(E,F),(F,A)\}$$
$$P_{A,B} = \{(A,B),(B,C),(C,A),(A,B)\}$$
$$P_{A,B} = \{A, B, C, A, B\}$$



1. A path can intersect itself and pass through the same link repeatedly. Each time a link is crossed, it is counted separately
2. A legitimate path on the graph: ABCBCADCEFA
3. In a directed network, the path can follow only the direction of an arrow.

Shortest Path (Geodesic Path)

The path with the shortest length between two nodes (distance).



Breadth-first search

Data: Graph: **G**, Vertex: *root*

```
begin
    foreach vertex v in G do
        v.distance ⟵ ∞
        v.parent ⟵ ∞
    end
    create empty queue Q
    root.distance ⟵ 0
    Q.enqueue(root)
    while while Q is not empty do
        current = Q.dequeue()
        foreach node n that is adjacent to current do
            if n.distance = INFINITY then
                n.distance ⟵ current.distance + 1
                n.parent ⟵ current
                Q.enqueue(n)
            end
        end
    end
end
```

Application of BFS:

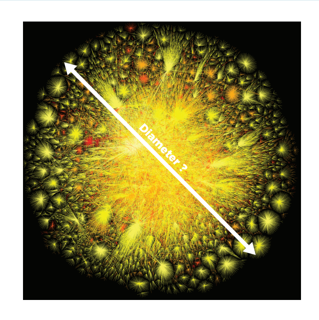
- Find connected Components
- Find all possible paths between two vertices
- Find the shortest path (length measured by number of path edges)

Dijkstra’s Algorithm

```
function Dijkstra(Graph, source):
  dist[source] := 0                 // Distance from source to source
  for each vertex v in Graph:        // Initializations
    if v ≠ source
      dist[v] := infinity          // Unknown distance function from source to v
      previous[v] := undefined     // Previous node in optimal path from source
    end if
    add v to Q                     // All nodes initially in Q
  end for
   while Q is not empty:             // The main loop
     u := vertex in Q with min dist[u]  // Source node in first case
     remove u from Q

     for each neighbor v of u:       // where v has not yet been removed from Q.
       alt := dist[u] + length(u, v)
       if alt < dist[v]:            // A shorter path to v has been found
         dist[v] := alt
         previous[v] := u
       end if
     end for
   end while
   return dist[], previous[]
 end function
```

Network distance



1. *Shortest path* : ***d_{ij}*** *between node i and j*
2. *Diameter* : ***d_{max}*** Maximum shortest path length

   between any two nodes: $\max_{i \neq j} d_{ij}$
3. *Average path length/distance : <d>*

For a connected graph: $$\langle d \rangle = \frac{1}{2L_{max}} \sum_{i \neq j} d_{ij}$$

For undirected graph ***d_{ij}* =*d_{ji}*** , so we only need to count them once: $$\langle d \rangle = \frac{1}{L_{max}} \sum_{i > j} d_{ij}$$

$$L_{max} = \frac{N(N-1)}{2}$$

Paths and Cycles

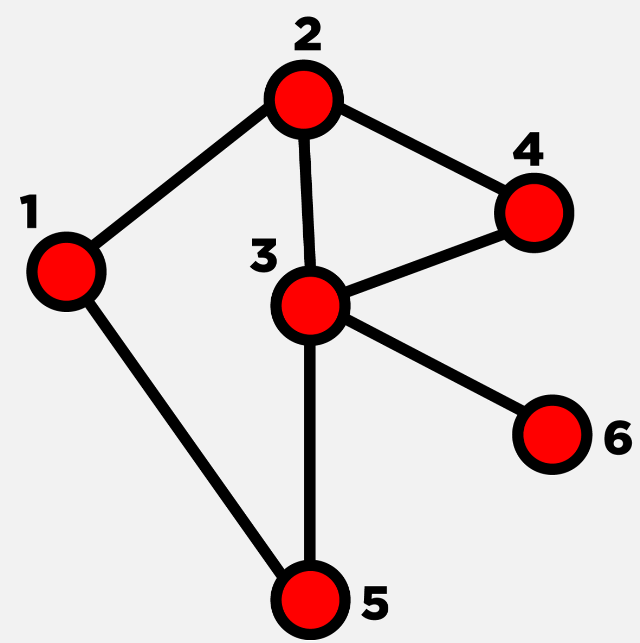
Number of path of length (r) between i and j

$$N_{ij}^{(r)} = [\mathbf{A}^r]_{ij}$$

Number of cycle of length (r)

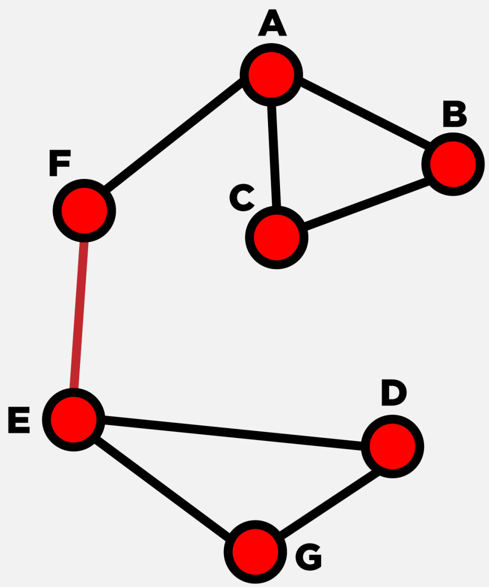
$$L_r = \sum_{i=1}^{N} [\mathbf{A}^r]_{ii} = Tr\ \mathbf{A}^r$$

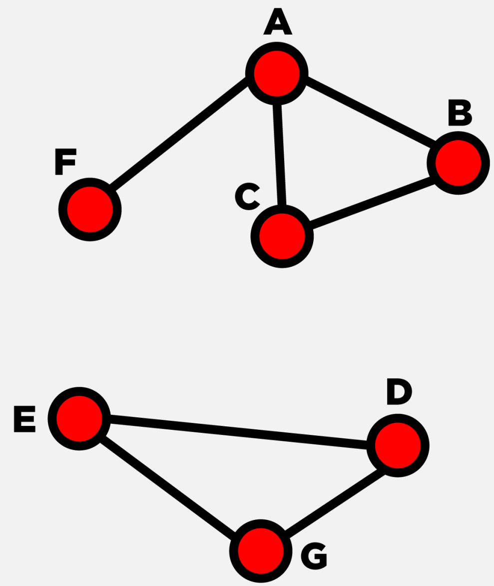
$\mathbf{A}^2$	1	2	3	4	5	6
1	2	0	2	1	0	0
2	0	3	1	1	2	1
3	2	1	4	1	0	0
4	1	1	1	2	1	1
5	0	2	0	1	2	1
6	0	1	0	1	1	1



Connected Component

Connected (undirected) graph: any two vertices can be joined by a path.
A disconnected graph is made up by two or more connected components.





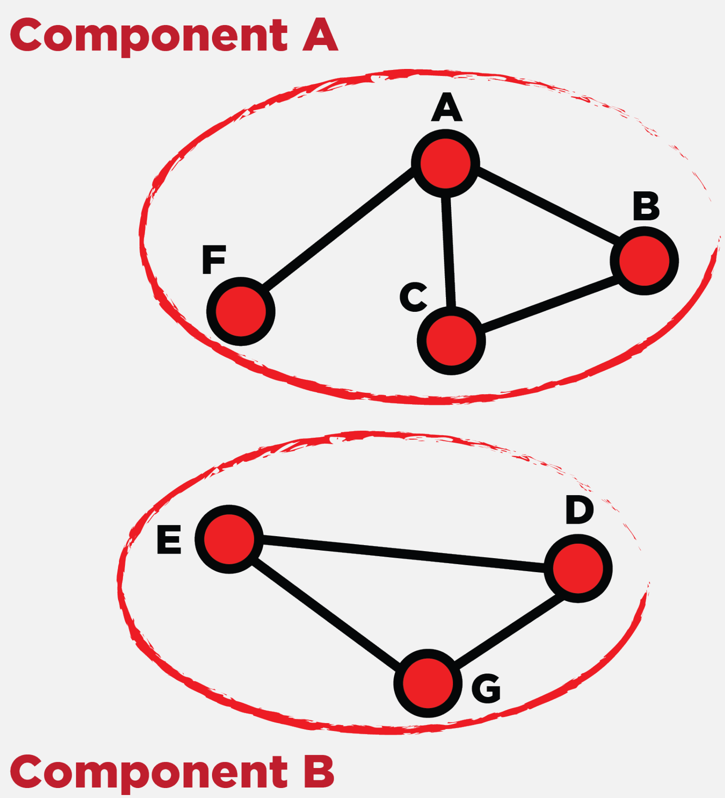
Largest Component:
Giant Component

The rest: Isolates

Bridge edge: if we erase it, the graph becomes disconnected.

Connectivity of Undirected Graphs

The adjacency matrix of a network with several components can be written in a block-diagonal form, so that nonzero elements are confined to squares, with all other elements being zero:



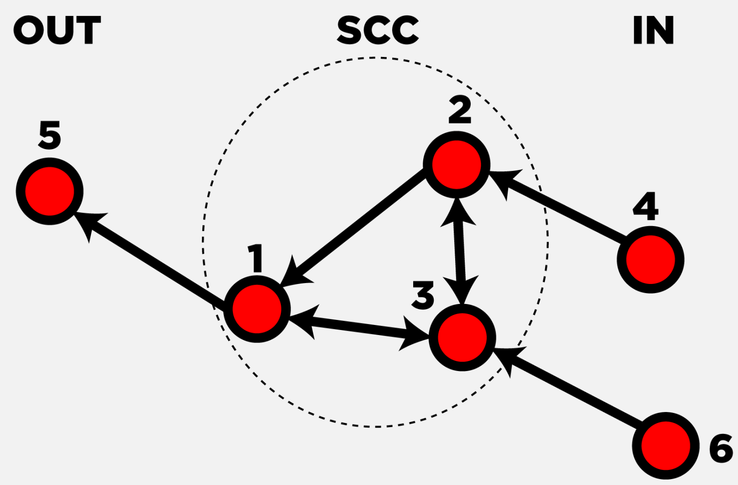
Components

$$\mathbf{A} = \begin{bmatrix} \boxed{\mathbf{A}} & 0 & \cdots \\ 0 & \boxed{\mathbf{B}} & \cdots \\ \vdots & \vdots & \ddots \end{bmatrix}$$

Connectivity of Directed Graphs

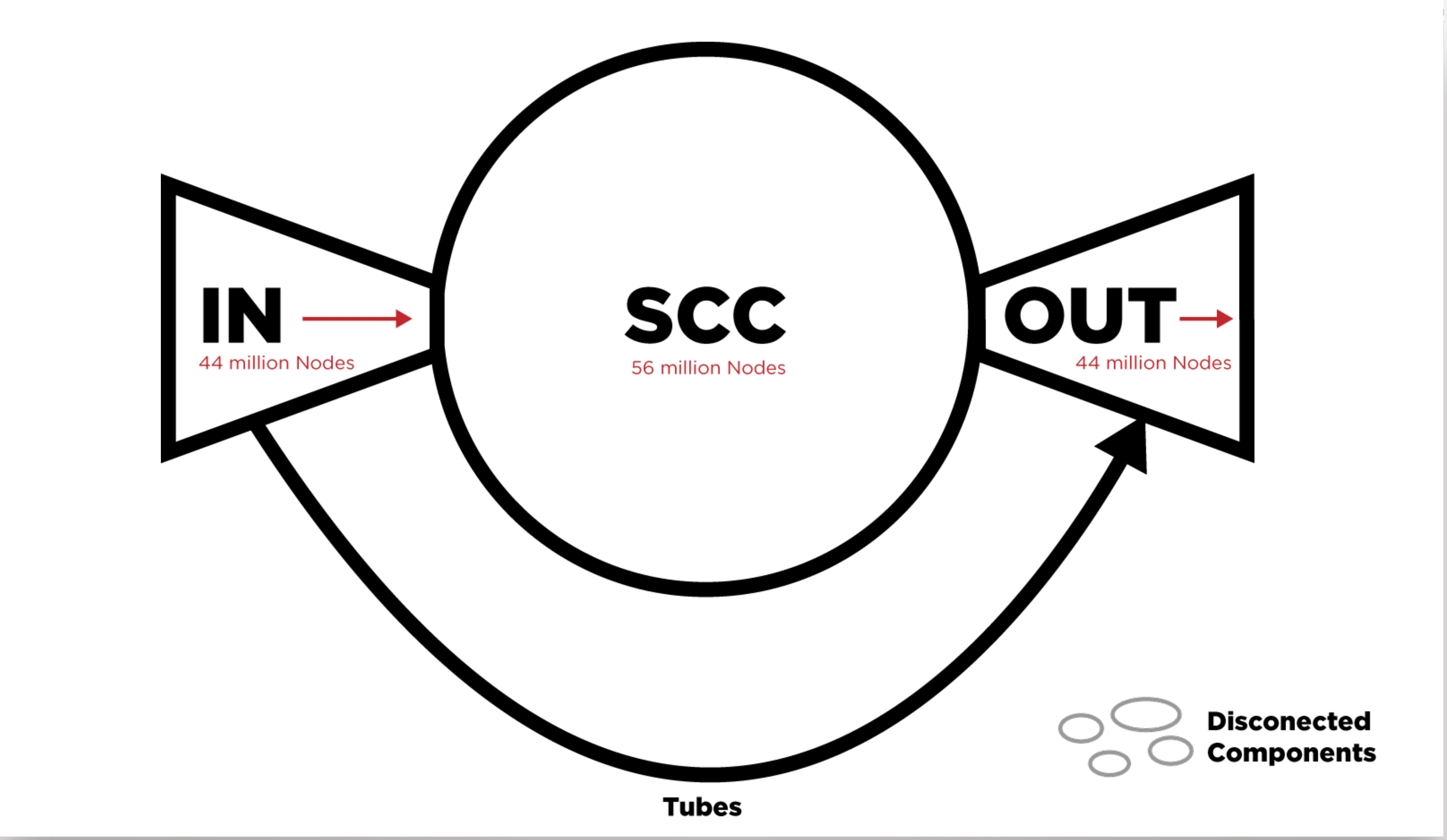
Strongly connected directed graph: has a path from each node to every other node and vice versa (e.g. AB path and BA path).
Weakly connected directed graph: it is connected if we disregard the edge directions.

Strongly connected components (SCC) can be identified, but not every node is part of a nontrivial strongly connected component.



In-component: nodes that can reach the scc,
Out-component: nodes that can be reached from the scc.

SCC of the Web Graph



Clustering – methode 1

How many of my friends are also firends ?

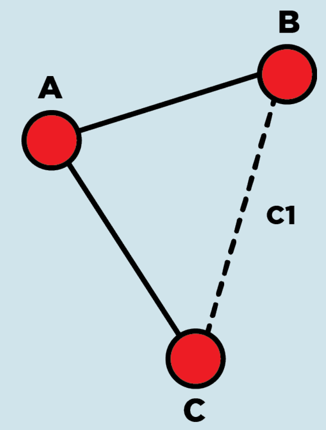
Consider a triangle/triple, the clustering coefficient (C_1) is defined as: a measures the fraction of closed triples.

- The 3 appears because for each triangle, we have 3 closed triples

$$C_1 = \frac{3 \times \#Triangle}{\#Triple}$$

Number of Triangle

$$C_1 = \underbrace{\frac{3 \times \boxed{\frac{1}{6} Tr \mathbf{A}^3}}{\frac{1}{2}\left(\sum_{i,j} \left[\mathbf{A}^2\right]_{ij} - Tr\mathbf{A}^2\right)}}_{\text{Number of Triple}}$$



- C is the probability of two node are also friends
- Or C is the probability that a triple is part of a triangle

Clustering – methode 2

How many of my friends are also firends ?

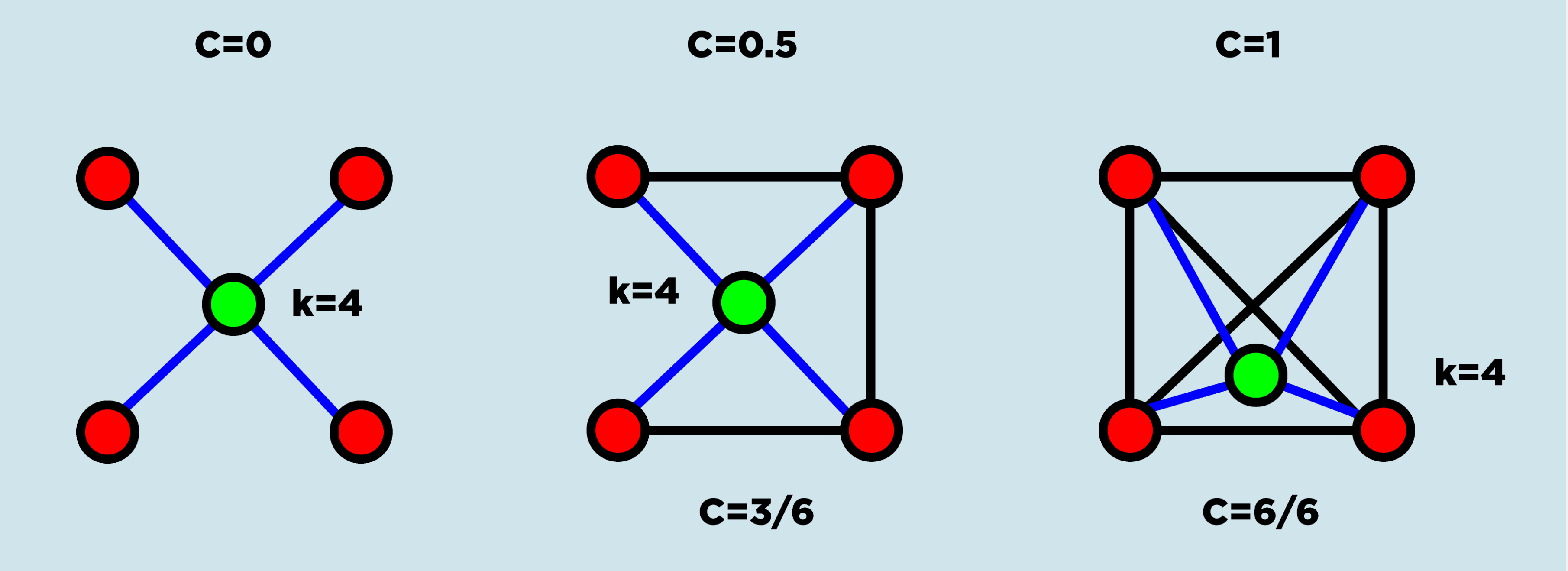
The second method consider a local measure of the clustering coefficient (C_2) is defined as: the fraction of pairs of neighbors who are connected.

Local Clustering Coeficient

$$C_2(i) = \frac{\#of\ links\ between\ neighbors}{\frac{k(k-1)}{2}}$$

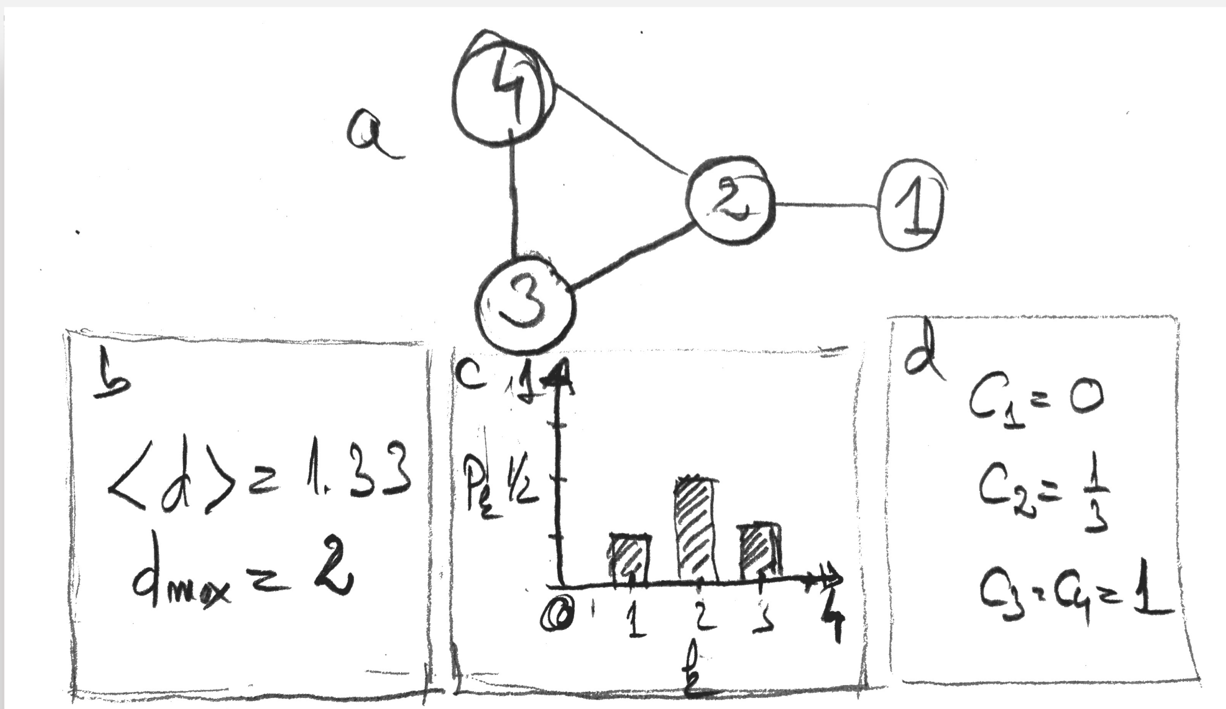
Average Clustering Coeficient

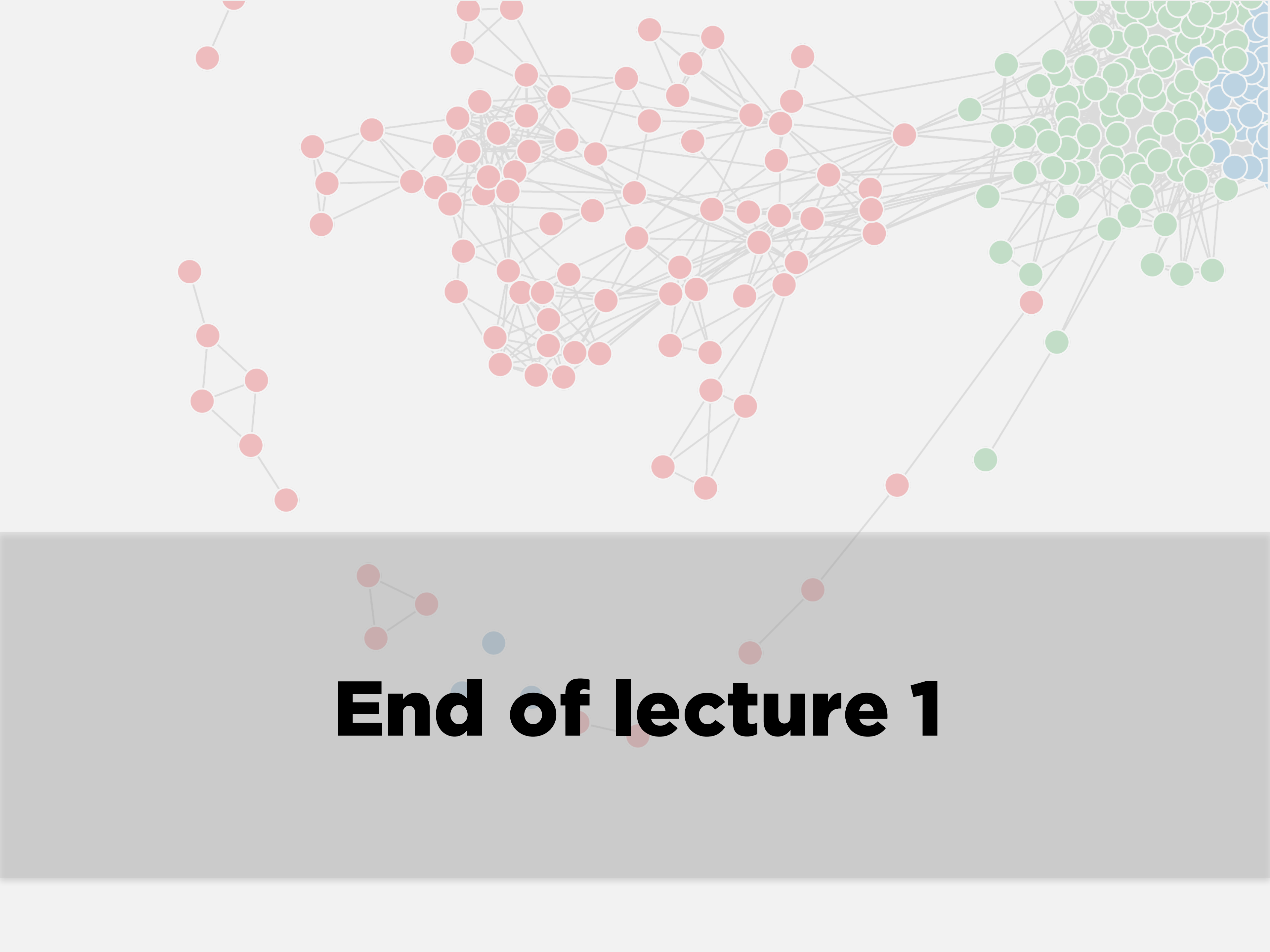
$$C_2 = \frac{1}{N}\sum_{i=1}^{N} C_2(i)$$



Three Central Quantities

1. Degree distribution: P_k
2. Path length: $\langle d \rangle$
3. Clustering coefficient: $C_i = \frac{2e_i}{k_i(k_i - 1)}$





End of lecture 1

Credits

- Complex Network Peter Dodds
- Five Lectures on Networks, Aaron Clauset
- Network Science course, Albert-László Barabási

- Networks An Introduction Mark Newman, 2010.
- Networks, Crowds, and Markets, D. Easley, J. Kleinberg, 2010.
- Network Science, Albert-László Barabási, 2016.