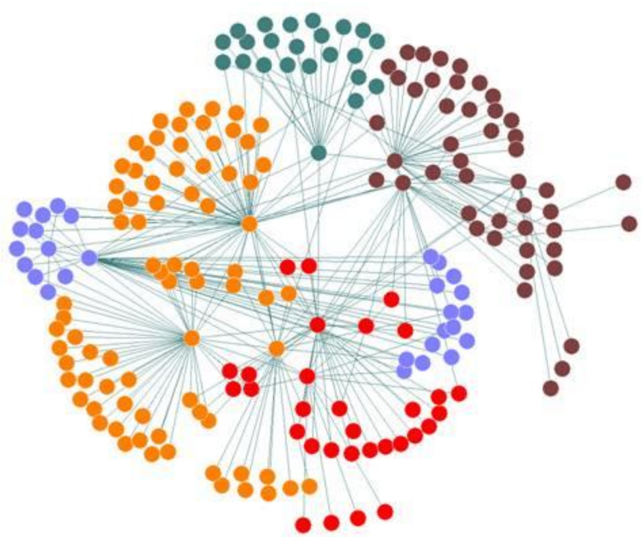
TEL AVIV UNIVERSITY

Algorithms and Applications in Social Networks



2023/2024, Semester A

Slava Novgorodov

Lesson #11

- Network definitions and properties
- Random Graphs, Centrality, Balance
- Communities
- Influence Maximization, Social Learning, Link Prediction
- Large Scale networks, Applications, Riddles

Summary of the course

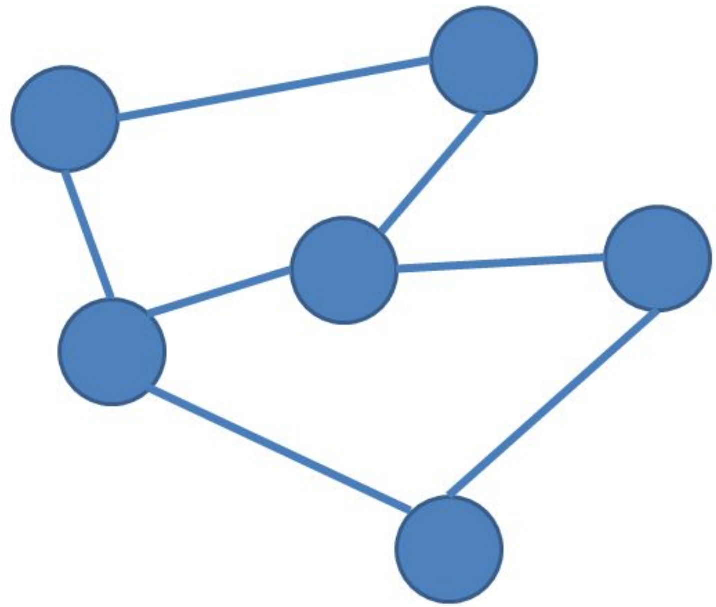
- Course consisted of 8-9 different topics in Social Networks (we will do an overview now)

- We learned both state-of-the-art algorithms and applications of these algorithms in the real world

- In addition we did practical (programming) exercises in these topics using Python and NetworkX library.

Network Definitions and Properties

Components of the Network

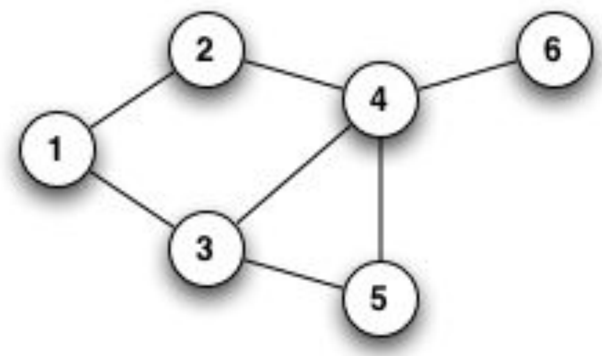


- **Vertices, Nodes** – objects/individuals **[V]**
- **Edges, Links** – interactions/relations **[E]**
- **Graph, Network** – the system **[G(V, E)]**

Directed/Undirected Graphs

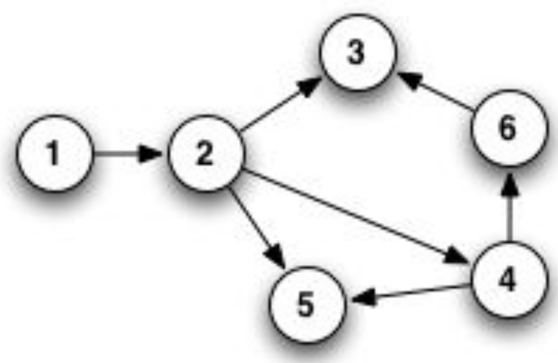
Undirected graph:

- Undirected, symmetrical edges
- Examples:
  - Friends (on Facebook)
  - Classmates

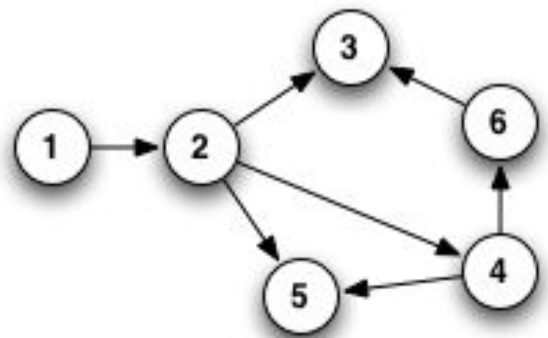


Directed graph:

- Directed edges
- Examples:
  - Followers (Instagram)
  - Phone calls



Representation of Graphs



Adjacency list

- **1:** 2
- **2:** 3, 4, 5
- **3:**
- **4:** 5, 6
- **5:**
- **6:** 3

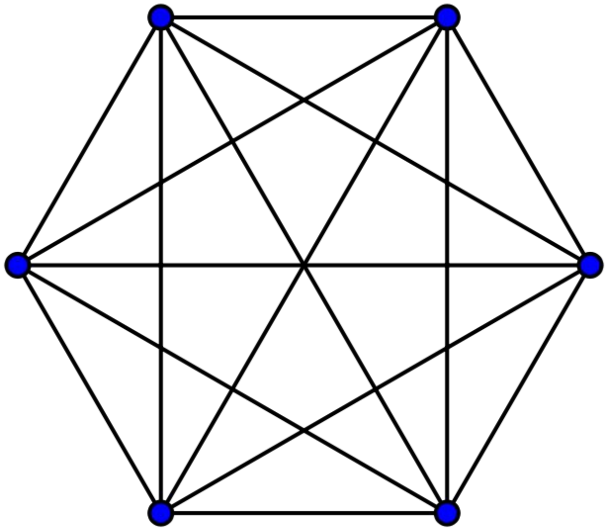
Edges list

- (1, 2)
- (2, 3)
- (2, 4)
- (2, 5)
- (4, 5)
- (4, 6)
- (6, 3)

Adjacency matrix

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

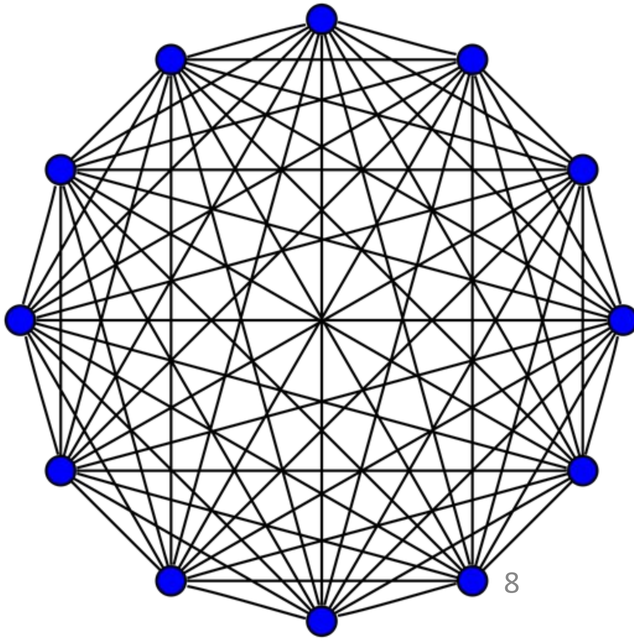
Complete Graph



The maximum number of edges in a graph of N nodes is

N*(N-1)/2

Undirected graph with maximum number of edges called **complete**



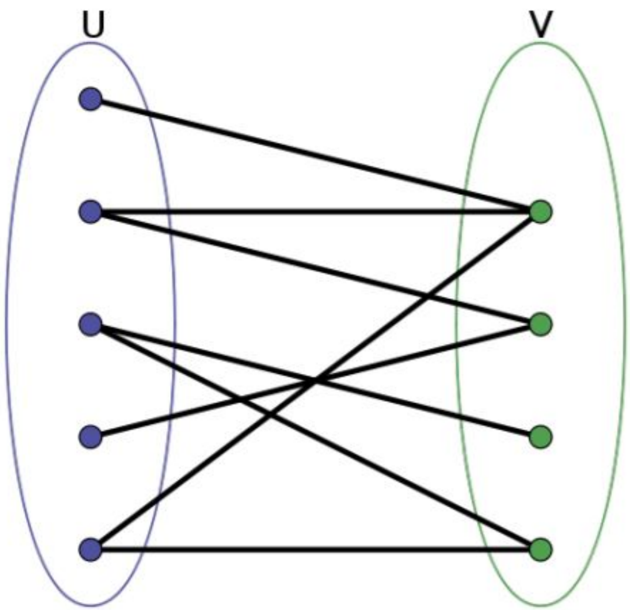
- clique is a complete subgraph
- triangle is a complete graph of size 3

Key Network Properties

• Degree distribution	$P(k)$
• Path length	h
• Clustering coefficient	C

Bipartite Graph

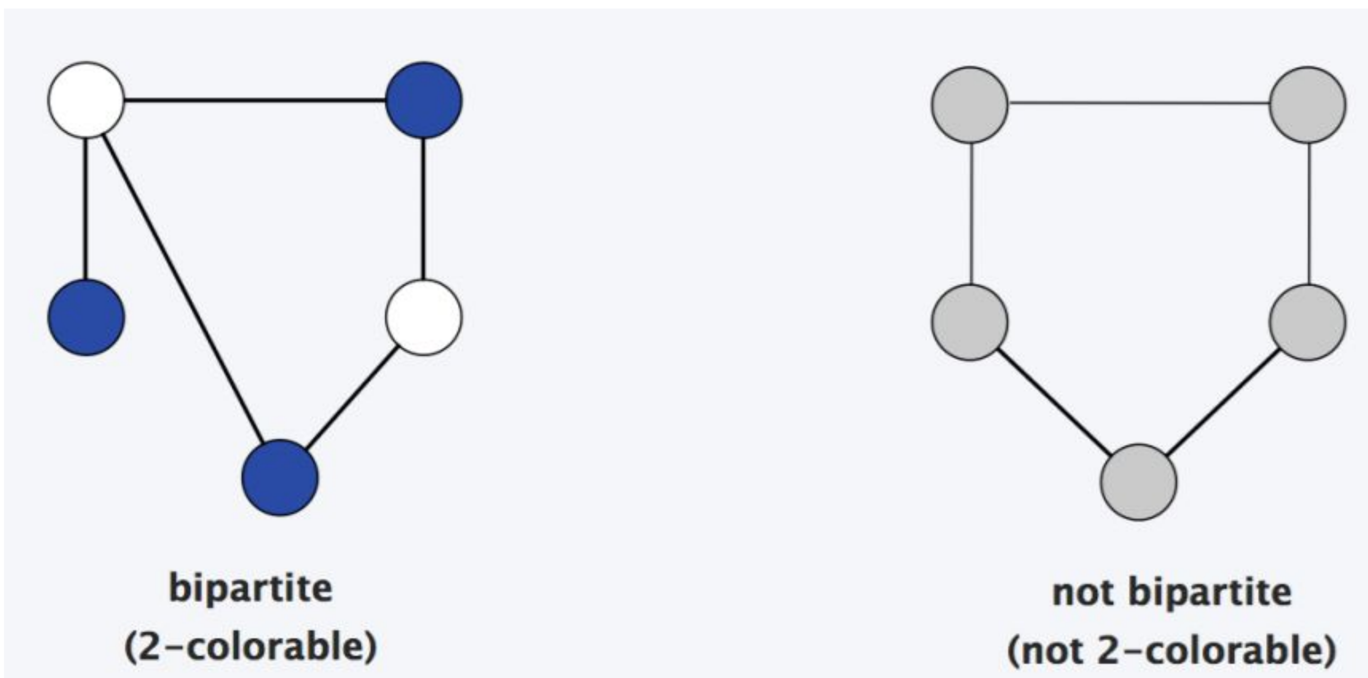
- A graph whose vertices can be divided into two disjoint sets U and V such that every edge connects a vertex in U to one in V



- A bipartite graph does not contain any odd-length cycles
- A bipartite graph can be vertex colored wtih 2 colors

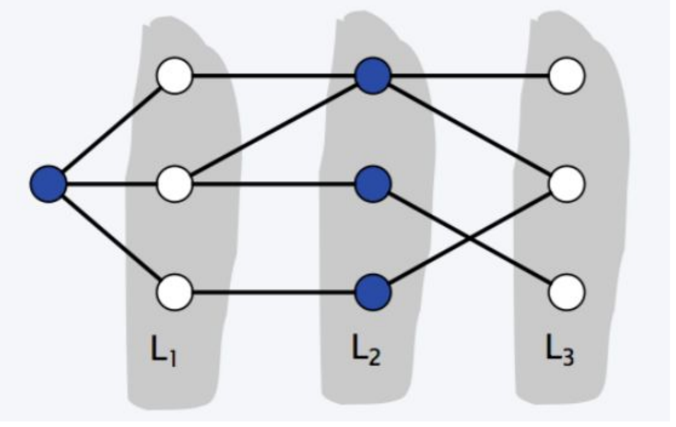
Testing Bipartiteness

- Triangle – not bipatite
- Graph contains an odd cycle – not bipartite



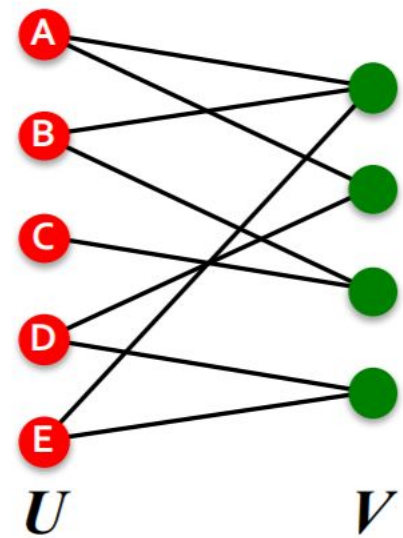
Testing Bipartiteness

- Is given graph bipartite?
- Algorithm:
  – Select and node and perform BFS, color each layer alternate colors
  – Scan all the edges, see if any edge has nodes with the same color (one layer nodes)

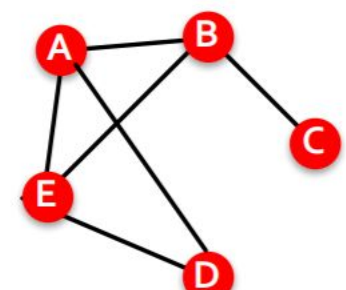


Usage of Bipartite Graph

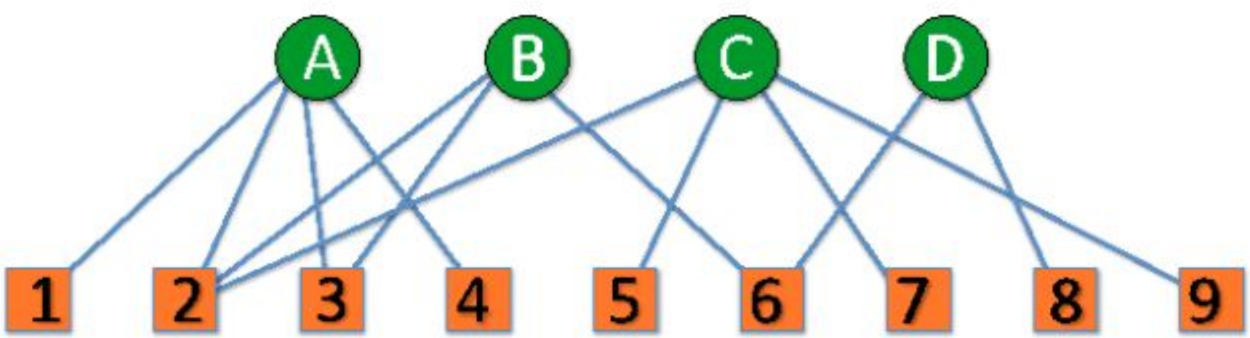
- Different types of nodes:
  - Users/Items ranking
  - Papers/Authors
  - Courses/Students



Folded network □



Folded version of the graph above



Random Graphs, Centrality, Balance

Erdős–Rényi model

- Two variants of the model:
  - G(n, m) – a graph is chosen uniformly from a set of graphs with **n** nodes and **m** edges
  - G(n, p) – a graph is constructed on **n** nodes, with probability of edge equals to **p**
- We will focus on the second variant
- Expected number of edges and average degree:

$$\overline{m} = \frac{n(n-1)}{2} p \qquad \overline{k} = \frac{1}{n} \sum_i k_i = \frac{2\overline{m}}{n} = p(n-1)$$

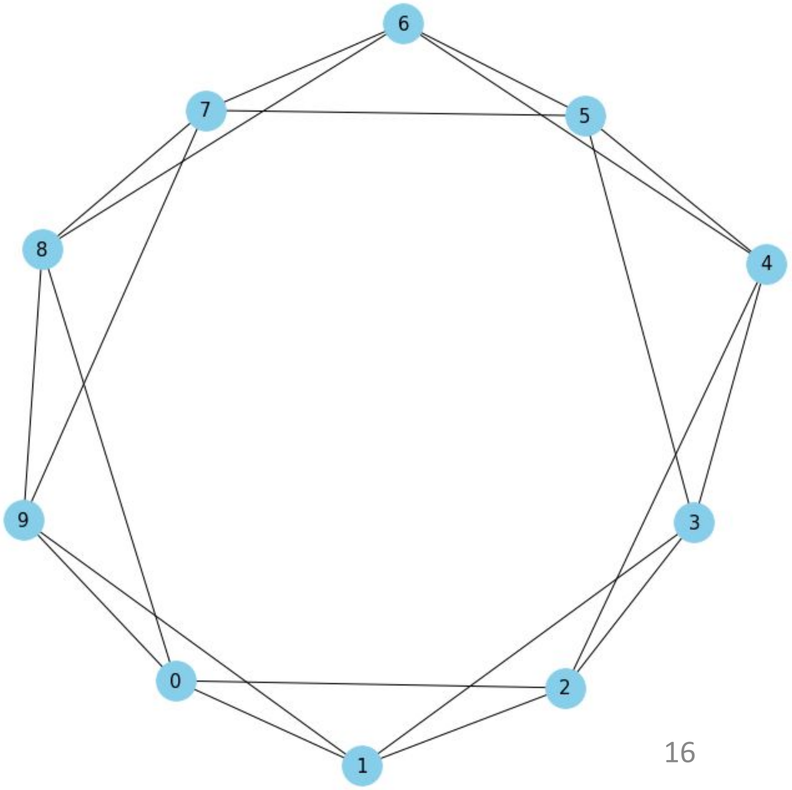
Watts-Strogatz model

- <u>Input</u>: **N** nodes, with average degree **K** and probability **p** of “recreating” the edge.

Step 1:

Create N nodes, connect each node to K/2 neighbors on the left and right (by IDs)

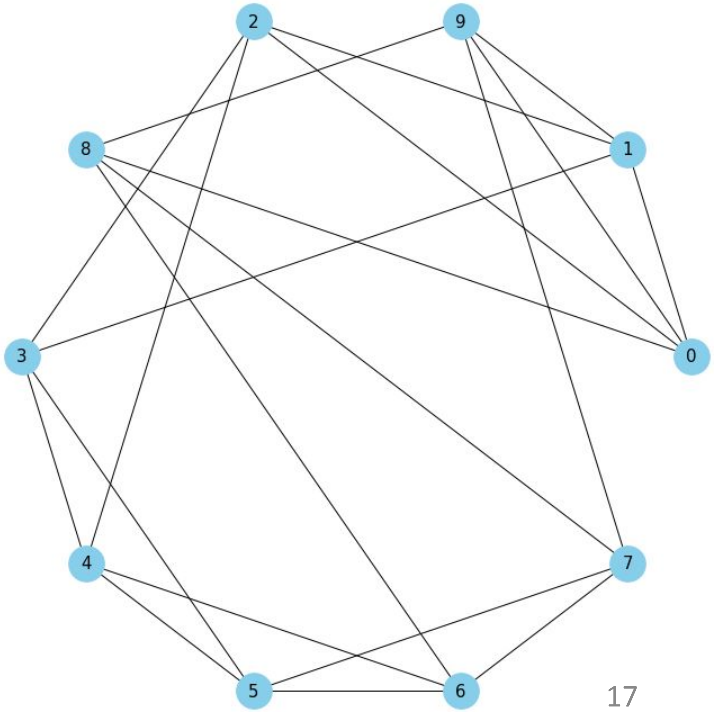
Result: High clustering coefficient, but also big diameter



Watts-Strogatz model

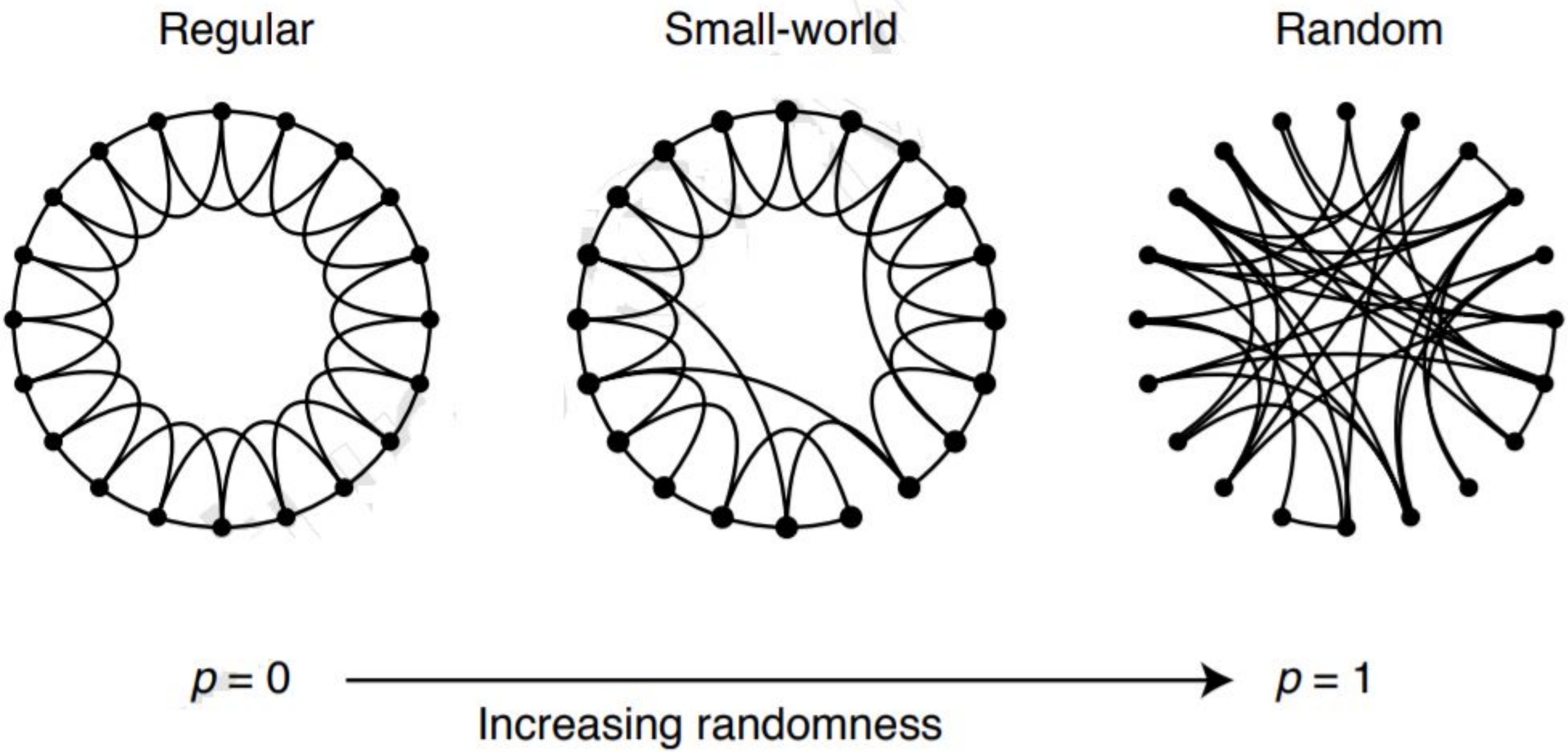
Step 2:

For each edge (i, j), decide if it should be recreated with probability p



Result: High clustering coefficient, and smaller diameter

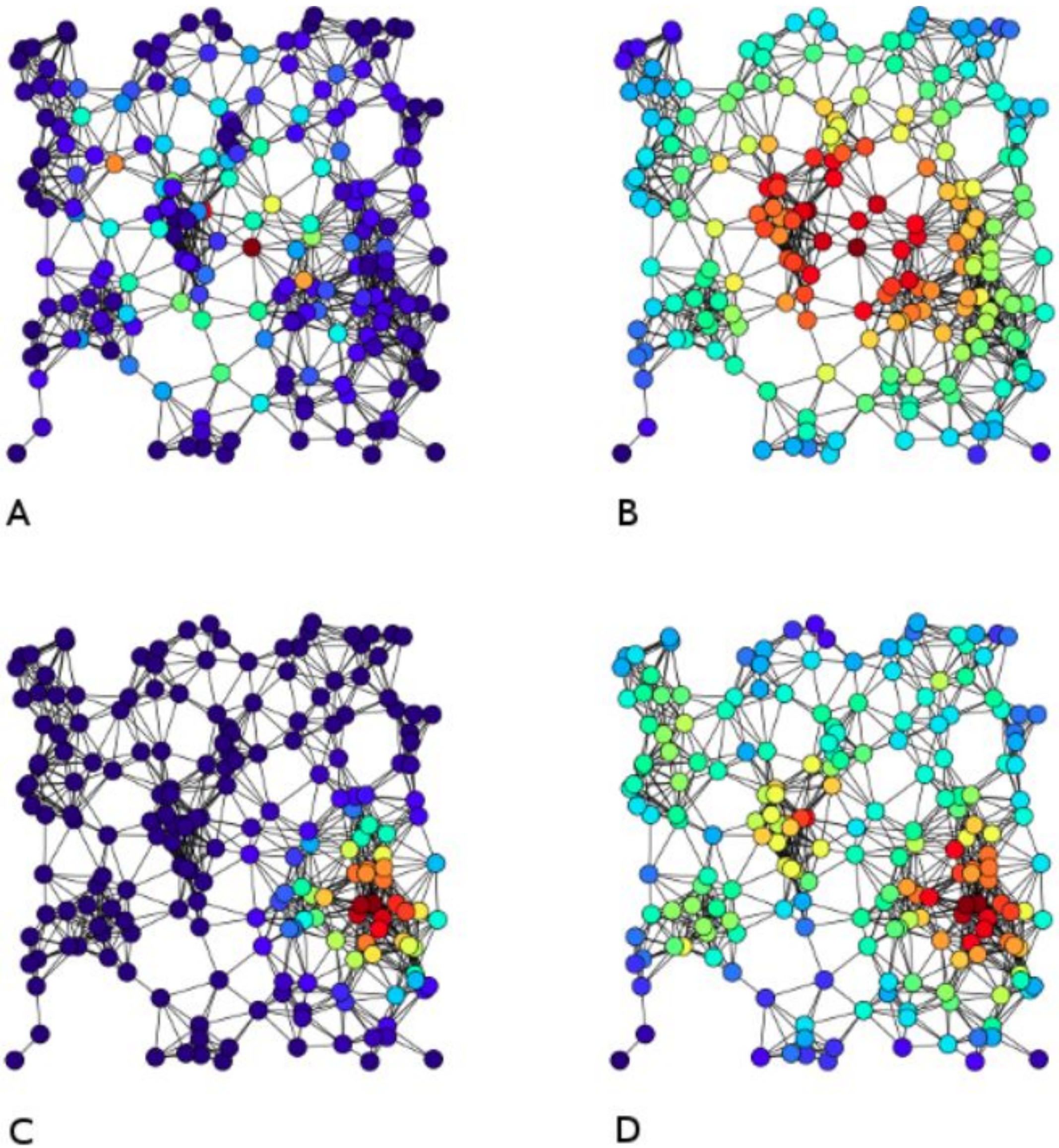
Watts-Strogatz model



Things to measure

- Degree Centrality:
  - Connectedness
- Closeness Centrality:
  - Ease of reaching other nodes
- Betweenness Centrality:
  - Role as an intermediary, connector
- Eigenvector Centrality
  - “Whom you know…”

Centralities



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

Networks with Signed Edges

- Also called: “Signed Network”
- Basic unit of investigation: **Signed triangles**
- Can be undirected or directed:

Signed Networks

- Network with **positive** or **negative** relationships

- Consider a complete signed undirected graph
  - **Positive** edges:
    - Friendship, positive sentiment, …
  - **Negative** edges:
    - Enemy, negative sentiment

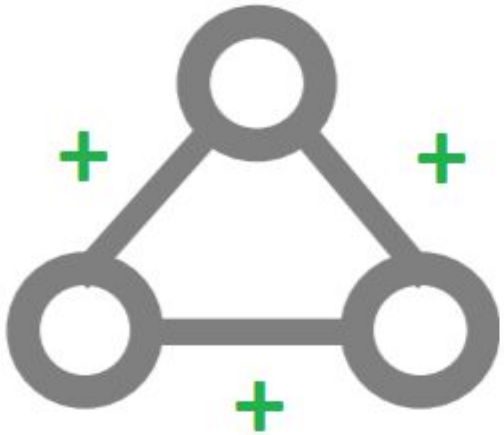
- Let’s focus on three connected nodes A, B, C

Theory of Structural Balance

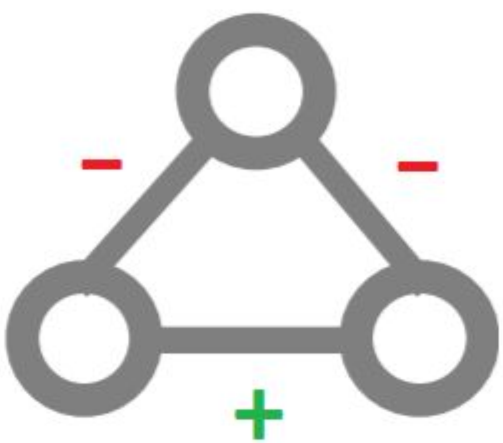
- Intuition (theory by Fritz Heider 1946):
  - **Friend** of a **friend** is a **friend**
  - **Enemy** of an **enemy** is a **friend**
  - **Enemy** of a **friend** is an **enemy**

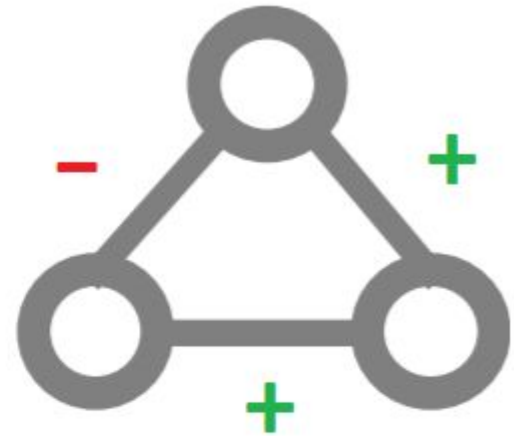
- Let's have a look on a triangle in a graph

Balanced/Unbalanced Triangles

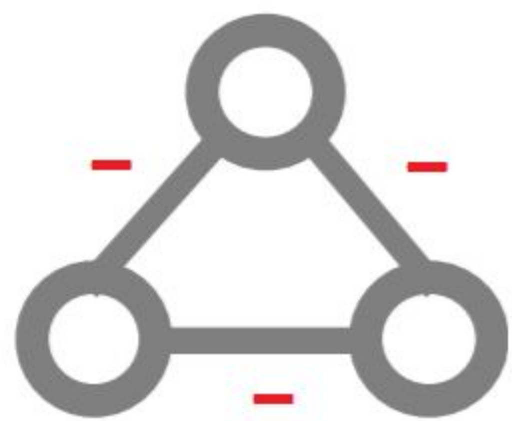


Balanced



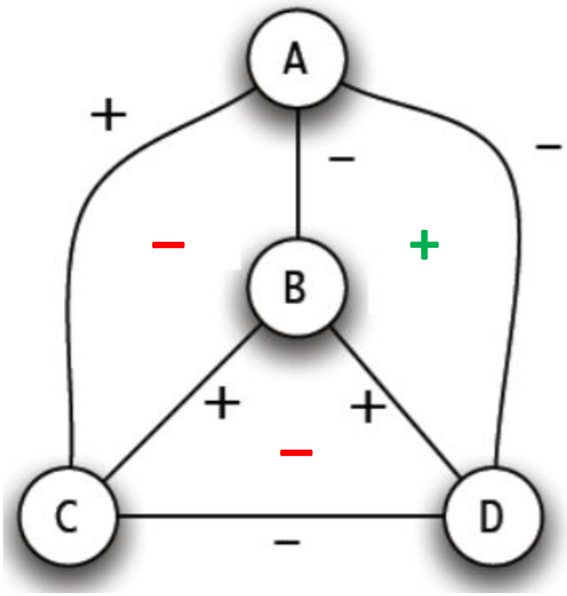


Unbalanced

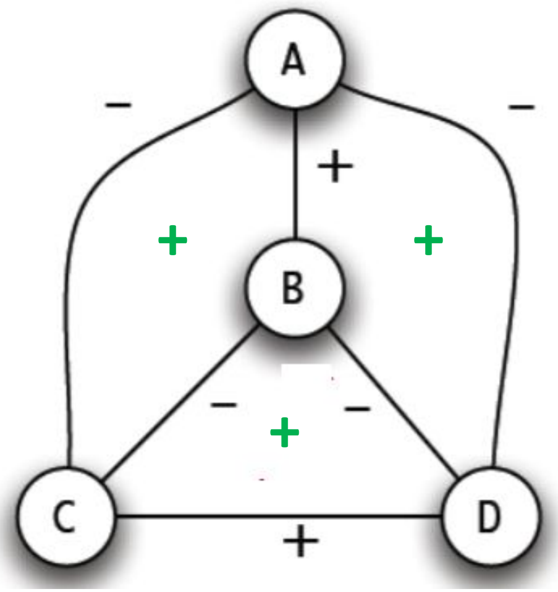


Balanced/Unbalanced Network

- Network is balanced if every triangle in the network is balanced.



Unbalanced

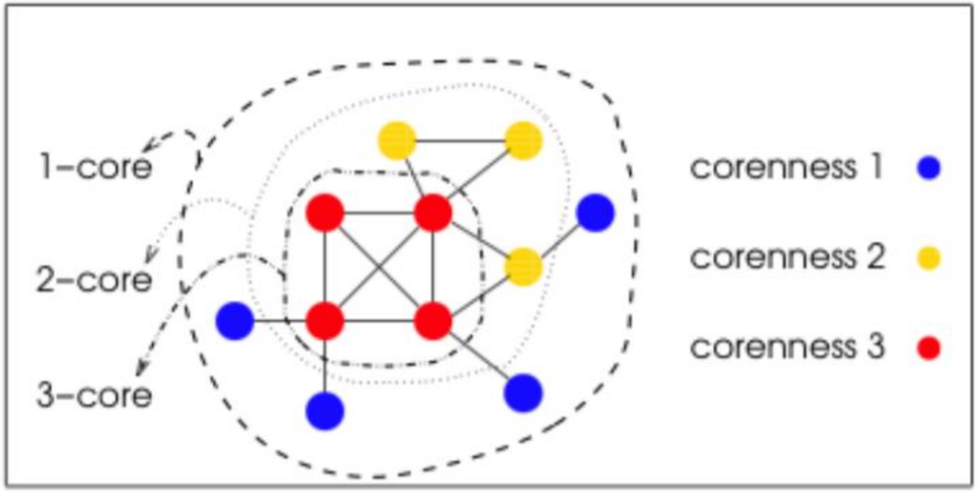


Balanced

Communities

Graph Core

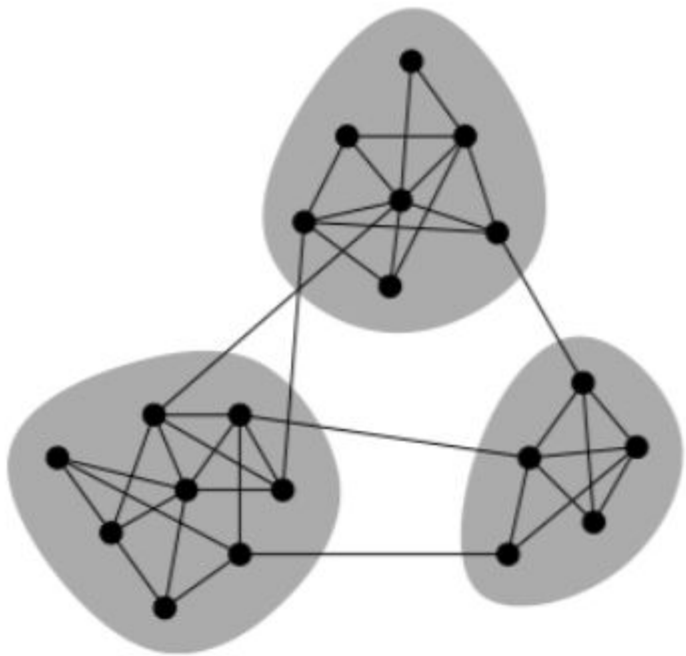
- A **k-core** is the largest subgraph S such as each node is connected to at least k nodes in S



- Every node in k-core has degree >= k
- (k+1)-core is always a subgraph of k-core
- Core number of node is the highest “k” of the k-core that contains this node

Community

Network Communities are group of vertices such that vertices inside the group connected with many more edges than between groups



Community Types

Detection algorithms:

- Non-Overlapping
  - Newman-Girvan algorithm
  - Label propagation
- Overlapping
  - K-clique percolation method
  - CONGO

Newman-Girvan algorithm

Algorithm: Newman-Girvan, 2004

Input: graph G(V,E)

Output: Dendrogram

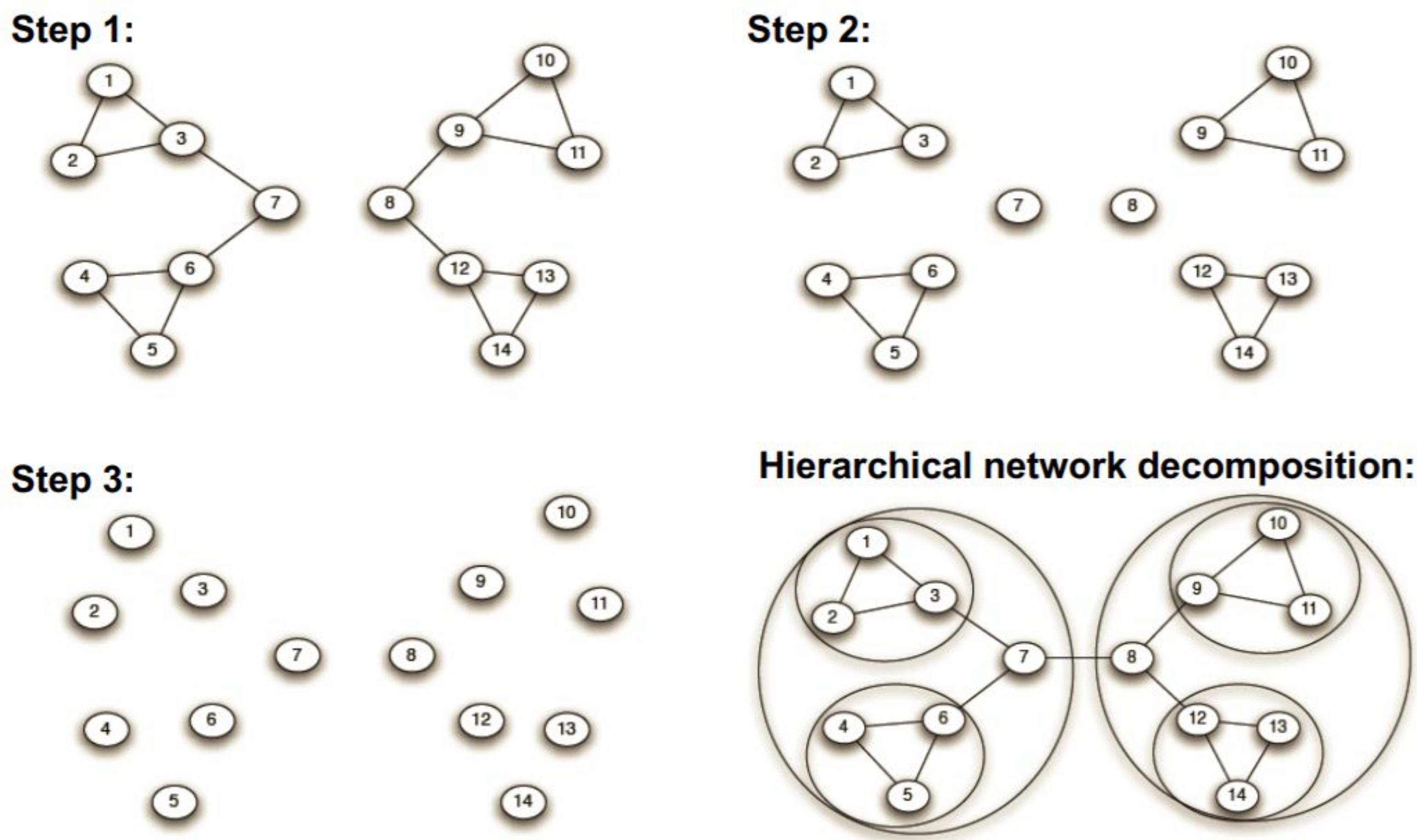
repeat

  For all $e \in E$ compute edge betweenness $C_B(e)$;

  remove edge e_i with largest $C_B(e_i)$;

until *edges left*;

NG – Step-by-step

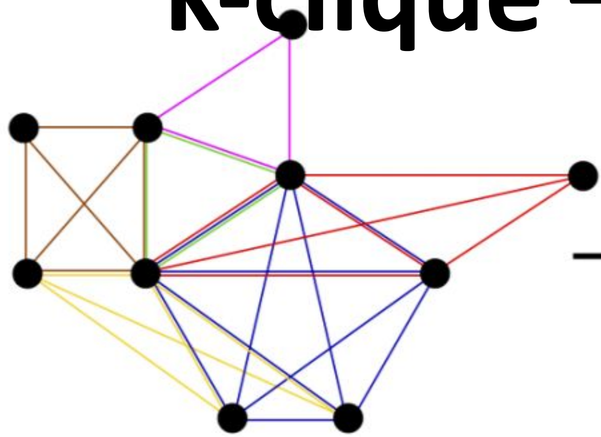


k-clique percolation method

By Palla et al. 2005:

- Find all maximal cliques
- Create clique overlap matrix
- Threshold matrix with k-1
- Communities are connected components

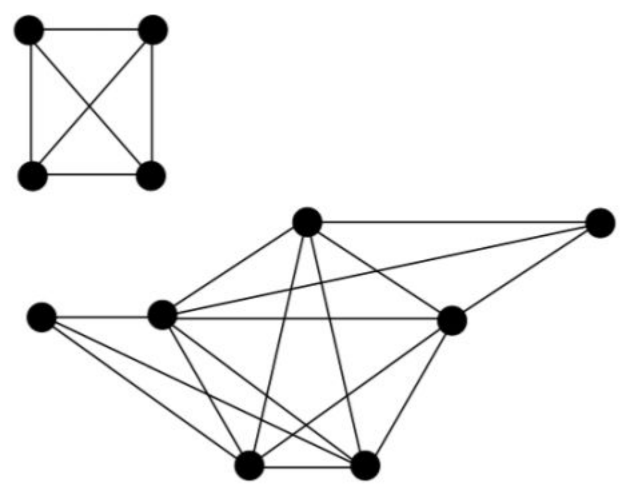
k-clique – Step-by-step



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

k=4

	1	1	0	0	1	0
	1	1	0	0	0	0
	0	0	0	0	0	0
	0	0	0	0	0	0
	1	0	0	0	1	0
	0	0	0	0	0	1



Influence Maximization,
Social Learning, Link Prediction

Models of influence

- Two basic models:
  - Linear Threshold Model
  - Independent Cascade Model

- Setup:
  - A social network is represented as a directed weighted graph, with each person as a node
  - Nodes start either active or inactive
  - An active node may trigger activation of neighboring nodes
  - Monotonicity assumption: active nodes never deactivate

Linear Threshold Model

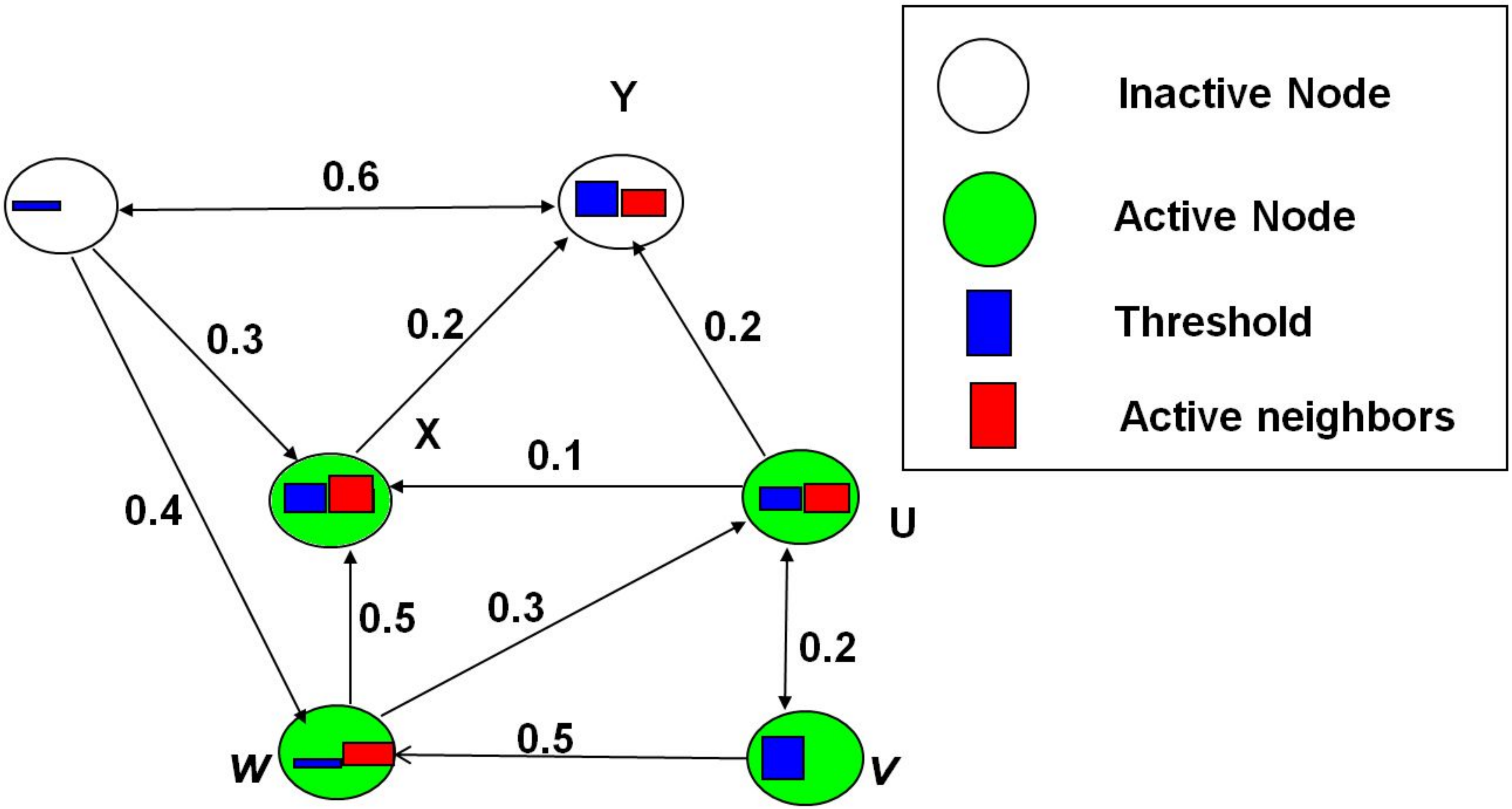
- A node *v* has random threshold $\theta_v \sim U[0,1]$
- A node *v* is influenced by each neighbor *w* according to a *weight* b_{vw} such that

$$\sum_{w \text{ neighbor of } v} b_{v,w} \leq 1$$

- A node *v* becomes active when at least (weighted) θ_v fraction of its neighbors are active

$$\sum_{w \text{ active neighbor of } v} b_{v,w} \geq \theta_v$$

LT - Example

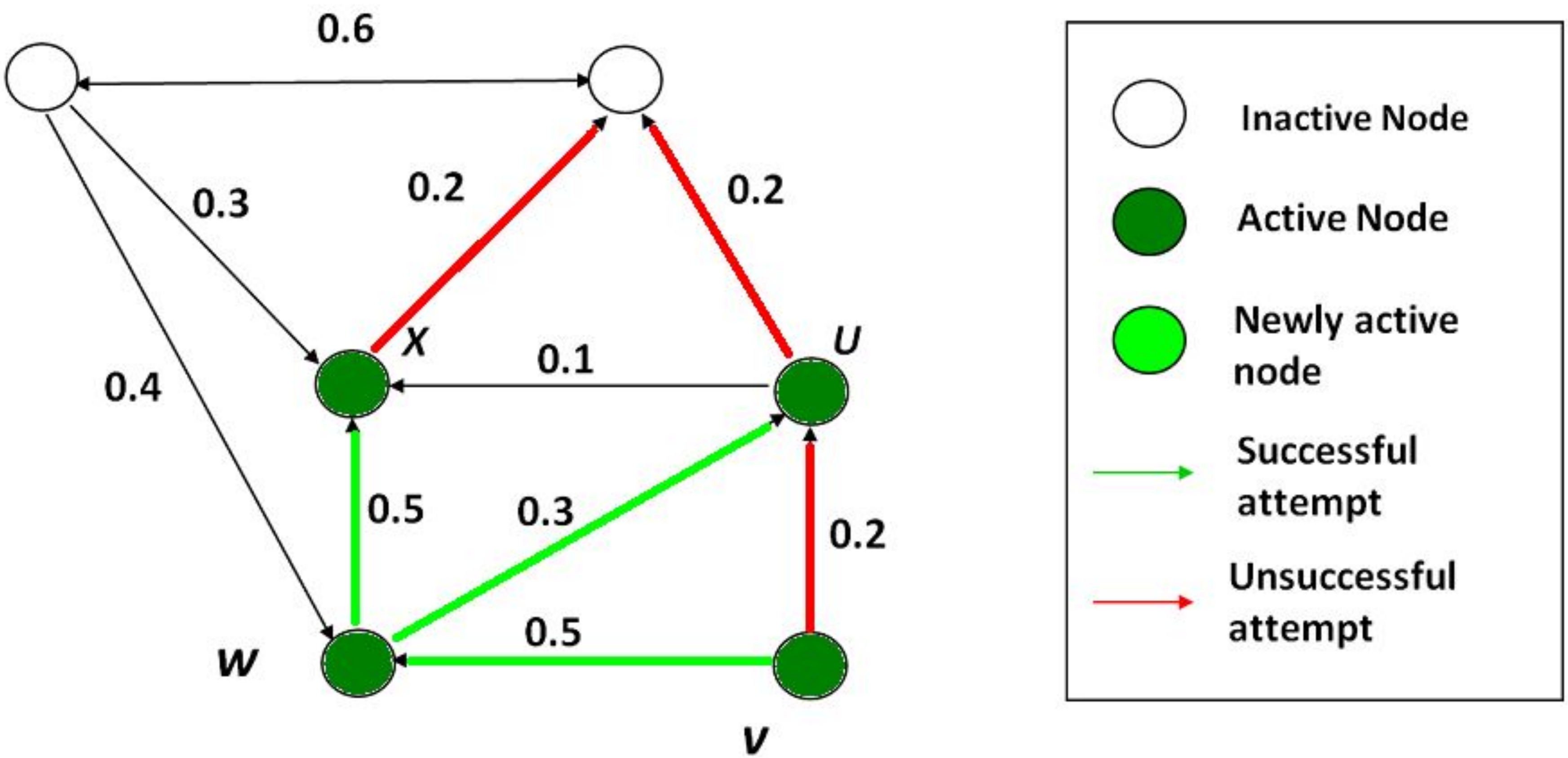

Y
0.6
0.3
0.2
0.2
X
0.1
0.4
U
0.5
0.3
0.2
0.5
W
V
Inactive Node
Active Node
Threshold
Active neighbors

Independent Cascade Model

- When node *v* becomes active, it has a **single** chance of activating each currently inactive neighbor *w.*

- The activation attempt succeeds with probability p_{vw} .

IC - Example



Stop!

Modeling Social Learning

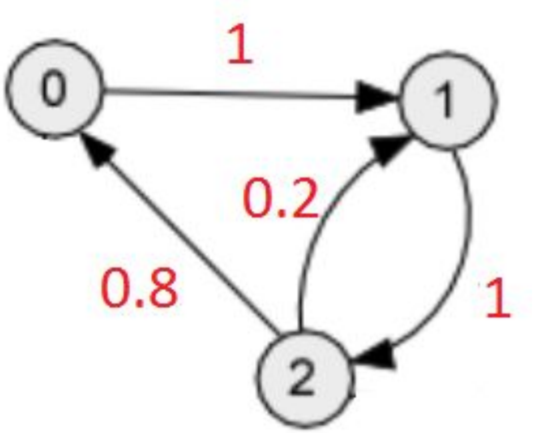
Nodes: Directors

Links: Influence (“listens to”)

Weights: % of influence (sum up to 1)

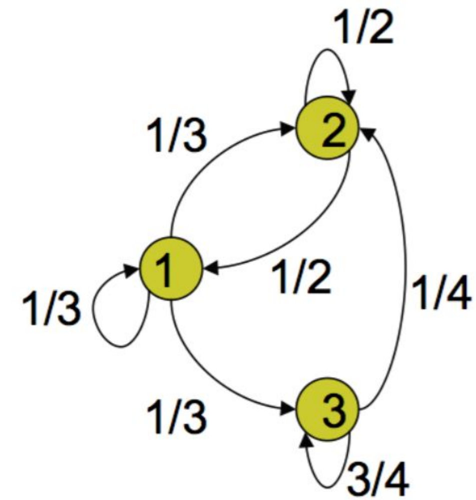
Example:

- “0” listens to “1”
- “1” listens to “2”
- “2” listens to “0” (80%) and “1” (20%)



How to “guess” the final decision?

DeGroot Model – Example



$$p(0) = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}$$

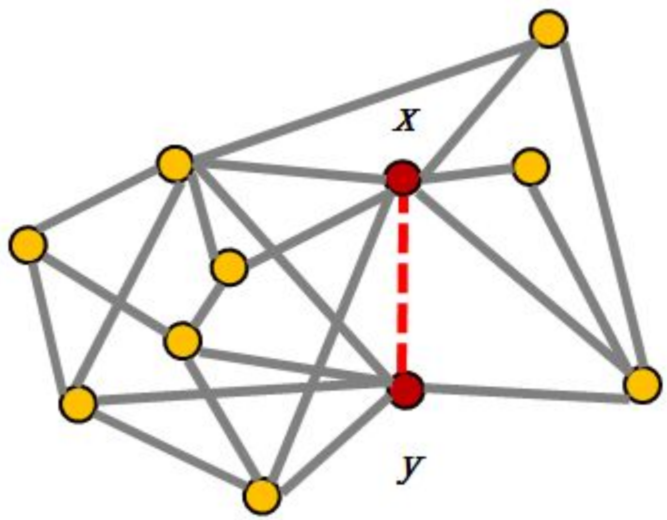
$$p(1) = Tp(0) = \begin{pmatrix} 1/3 & 1/3 & 1/3 \\ 1/2 & 1/2 & 0 \\ 0 & 1/4 & 3/4 \end{pmatrix} \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 1/3 \\ 1/2 \\ 0 \end{pmatrix}$$

$$p(2) = Tp(1) = \begin{pmatrix} 1/3 & 1/3 & 1/3 \\ 1/2 & 1/2 & 0 \\ 0 & 1/4 & 3/4 \end{pmatrix} \begin{pmatrix} 1/3 \\ 1/2 \\ 0 \end{pmatrix} = \begin{pmatrix} 5/18 \\ 5/12 \\ 1/8 \end{pmatrix}$$

$$p(20) = Tp(19) = \begin{pmatrix} 3/11 \\ 3/11 \\ 3/11 \end{pmatrix}$$

$$p(21) = Tp(20) = p(20)$$

Link Prediction



- **Local**
  - (negated) Shortest path (SP)
  - Common neighbors (CN)
  - Jaccard (JC)
  - Adamic-Adar (AA)
  - Preferential attachment (PA)
  - …
- **Global**
  - Katz score
  - Hitting time
  - PageRank
  - …

Notation: Neighbors of x: N(x) = $\Gamma(x)$

Degree of x: d_x = |N(x)| = | $\Gamma(x)$

Link Prediction

- Pick a favorite heuristic method
- Compute over all pairs of nodes
- Sort
- Take the top-k

Evaluation methods (precision, recall)

Large Scale networks, Applications, Riddles

M/R Approach

- Read the data
- **Map**: Extract information from each row
- Shuffle
- **Reduce**: Aggregate, filter, transform…
- Write the results

M/R and Social Networks

- Representation:
  - Adjacency Matrix vs Neighbors list?

- As Map Reduce takes text files and works line by line, better to have each line as a separate node:

A -> B C D
B -> A C D E
C -> A B D E
D -> A B C E
E -> B C D

Applications

- Crime, Fraud, Terrorism detection and prevention:
  - Bi-partite graphs
  - Centrality
  - Communities detection
  - Link prediction
  - …
- Feed generation algorithms
- Advertisement in Social Networks and outside
- Data leakage & its prevention

Riddles

- Short questions related to Social Networks, that can be solved without prior knowledge in SN, but much easier if you did the course.
- Related to possible/non possible network structure, number of edges, nodes, average degree, path length, diameter, balance, communities, etc.
- Sometimes these questions are used as a “logical” quiz in interviews.

Last slide

- I hope you enjoyed the course as much as I enjoyed it!

- Please fill the feedback (“Seker Horaa”) – it’s very important for me for the future courses

- Stay in touch (slavanov@post.tau.ac.il)

GOOD LUCK!

Thank you!

Questions?