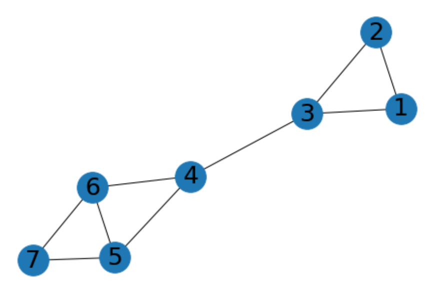
SC435
Introduction to Complex Networks
Mathematics of Networks-1

Representation of networks

- The network consist of entities connected with each other
- The structure of these connections are represented through graphs.
- A graph is represented by two sets
  - A **vertex set** V of the entities participating in the network. The size of the network $n(N)$ is the cardinality of this set.
  - An **edge set** E (also called link or tie set) of the connections between vertices. Usually, $m(L)$ denotes the number of edges.



Graph Terminology

Types of graphs

- Simple graph: don't have loops/multiple edges
- Multigraph: have multiple edges
- Pseudograph: multiple edges and self-loops
- Simple-directed graph: simple but directed
- Directed multigraph: directed with multiple edges and loops
- Mixed graphs: Directed and Undirected (enzymes/metabolic network)

Special Graphs

- complete graph (K_n)
- cycles (C_n)
- Wheel (W_n)
- n-cube (hypercube) (Q_n)
- Complete Bipartite graphs ($K_{n,m}$)

Mathematical Representation

Adjacency Matrix

- Mathematically convenient way of representing a network.
- Adjacency matrix of a simple graph (boolean matrix)

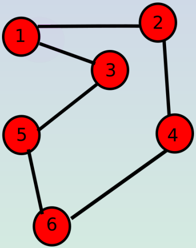
$$\boxed{A_{ij} = \begin{cases} 1 & \text{if there is an edge between vertices } i \text{ and } j \\ 0 & \text{otherwise} \end{cases}}$$

- For directed graph (digraph)

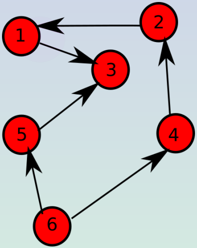
$$\boxed{A_{ij} = \begin{cases} 1 & \text{if there is an edge from } j \text{ and } i \\ 0 & \text{otherwise} \end{cases}}$$

Mathematical Representation

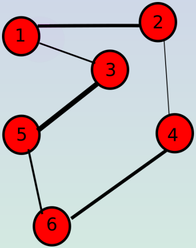
example



$$\begin{pmatrix} 0 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 \end{pmatrix}$$



$$\begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$



$$\begin{pmatrix} 0 & 2 & 1 & 0 & 0 & 0 \\ 2 & 0 & 0 & 0.5 & 0 & 0 \\ 1 & 0 & 0 & 0 & 2 & 0 \\ 0 & 0.5 & 0 & 0 & 0 & 1.5 \\ 0 & 0 & 2 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1.5 & 1 & 0 \end{pmatrix}$$

Mathematical Representation

Adjacency Matrix

- For the case of no self loops, the diagonal of the matrix is all 0.
- For undirected networks, since the existence of edge (i, j) assumes the existence of the edge (j, i), matrix A is symmetric. For undirected graph A is in general not symmetric.
- If node i has a self edge then we set $A_{ii} = 2$ for undirected graph and $A_{ii} = 1$ for directed graphs.
- Multiedges are represented by setting the corresponding entry equal to the number of distinct edges.

Mathematical Representation

Symmetric Adjacency matrix

- The adjacency matrix for simple undirected network is real and symmetric. ($A^T = A$)
- Eigenvalues are all real.
- Eigenvectors are necessarily perpendicular
- Always diagonalizable $A = VDV^T$
- For repeated eigenvalues orthonormal basis of eigenvectors can still be found (Gram-Schmidt Process)
- Principal Axis Theorem: If A is a real symmetric $n \times n$ matrix, there is always an orthonormal basis for $\mathcal{R}^n$ consisting of eigenvectors of A.

Mathematical Representation

Other representations

- **Incidence Matrix** (includes info about the edges): The incidence matrix with respect to ordering V and E is the $n \times m$ matrix $M = \{m_{ij}\}$, where

$$\boxed{m_{ij} = \begin{cases} 1 & \text{when edge } e_i \text{ is incident with vertex } v_i \\ 0 & \text{otherwise} \end{cases}}$$

- **Adjacency list**: specify the vertices that are adjacent to each vertex in the graph.
- **Functional representation**: useful for multigraph and some special graphs such as weighted and signed graphs.
- **Graph Laplacian**: $L_{ij} = \delta_{ij} k_i - A_{ij}$

Nodes and Links

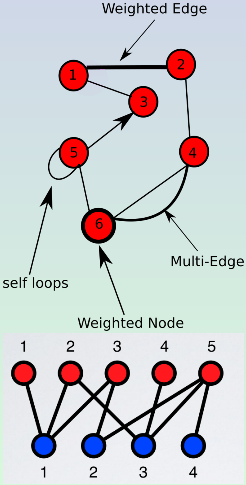
Fundamental question $\Longrightarrow$ Definition of nodes and links

Network	Node	Edge
Internet	Computer or Router	cable or wireless data connection
World Wide Web	Web page	hyperlink
Citation Network	Article Patent	Citation
Friendship Network	Person	Friendship
Twitter	Users	hashtags
Metabolic Network	Metabolite	Metabolic Reaction
Neural Network	Neuron	Synapse
Food web	Species	Predation

Nodes and Links

Possible Networks

- undirected (simple): Friendship
- directed acyclic: citation, food webs, epidemiological
- directed: twitter, neuronal, economic
- bipartite: collaboration, rail, airline



[DOUBT]

Brain Network

- Multiscale Organization (How to define nodes & links)
- Different experiments: Information (How, What??)

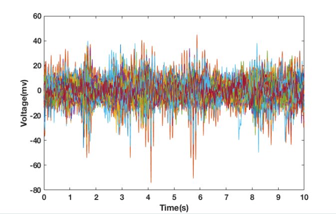
class	Measurement	Type of network
Structural	anatomical connections	directed
Functional	statistical dependence (time domain)	directed/undirected
Effective	causal relationship	directed

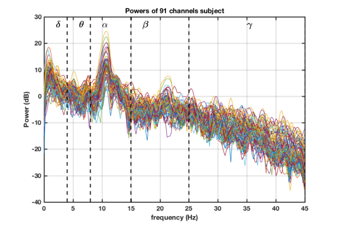
Scale	Structural	Functional
Microscopic	Invasive methods (electron microscopy), (neurons & synapse)	Invasive electrophysiology (neurons & correlations)
Mesoscopic	Invasive tract tracing (parcellation & pathways)	Invasive electrophysiology (electrodes & correlations)
Macroscopic	Diffusion MRI (voxels, sensors)	fMRI, EEG, MEG ((electrodes & correlations))

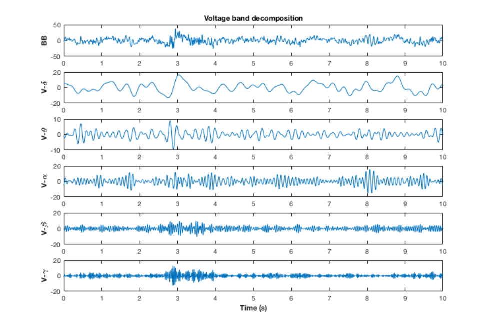
Structural $\Longleftrightarrow$ Functional

micro $\Longleftrightarrow$ macro

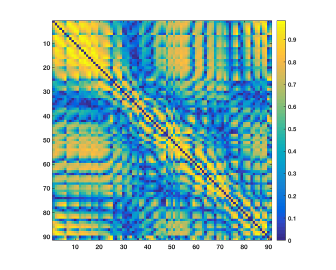
Constructing Networks
Data: Chennu et. al. PlosOne (2016)

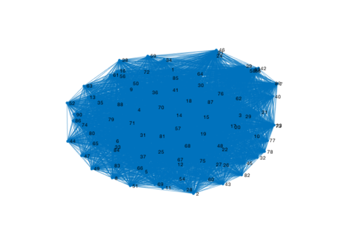


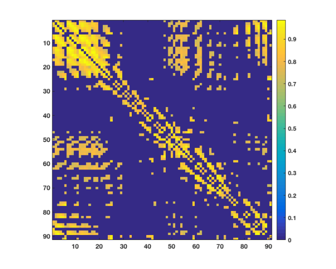


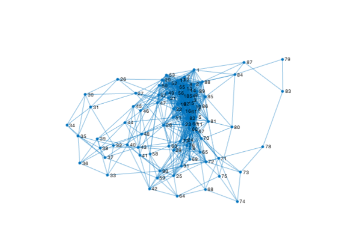


Constructing Networks
Brain Network









Cocitation coupling

- The Cocitation of two vertices i and j in a directed network is the number of outgoing edges pointing to both i and j

$$\boxed{A_{ik}A_{jk} = \begin{cases} 1 & \text{if } i \text{ and } j \text{ are both cited by } k \\ 0 & \text{otherwise} \end{cases}}$$

- Cocitation of nodes i and j is

$$C_{ij} = \sum_{k=1}^{n} A_{ik}A_{jk}$$

- Cocitation Matrix

$$C = AA^T$$

Bibliographic coupling

The bibliographic coupling of two vertices i and j in a directed network is the number of nodes that they both point to

- In order for node k to contribute to the bibliographic coupling of i and j, the following needs to be true: $A_{ki}A_{kj} = 1$.
- Bibliographic coupling of nodes i and j:

$$B_{ij} = \sum_{k=1}^{n} A_{ki}A_{kj} = \sum_{k=1}^{n} A_{ik}^{T}A_{kj}$$

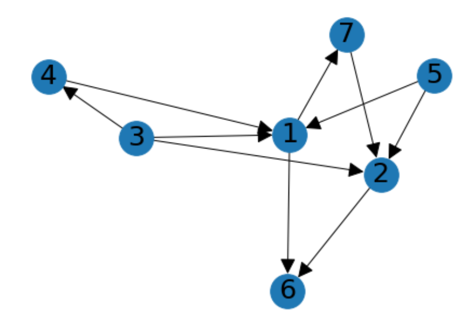
- The $n \times n$ adjacency matrix of the corresponding bibliographic coupling network is:

$$B = A^T A$$

- diagonal elements of B (B_{ii}):

Example

$$A = \begin{pmatrix} 0 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$



Bibliographic vs Cocitation coupling

- While both coupling methods are similar they can give completely different structure: Cocitation is heavily based on incoming edges whereas bibliographic coupling is based on outgoing edges.
- They can be used for vertex similarity, higher the cocitation of two papers in a citation network, the more possible is that these papers deal with similar topic.
- The cocitation and bibliographic coupling matrices are used in search algorithms for directed networks and search engines such as Science citation index, HITS etc.

Acyclic Directed Network

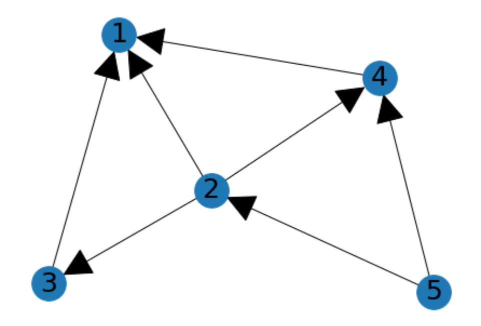
- A *cycle* in a directed network is a closed loop of edges with arrows on each of the edge pointing in the same direction. (real directed networks such as www have cycles)
- Directed Acyclic network/graph (DAG): directed networks that have no cycles are called acyclic networks or DAG. Example, citation network
- self edges also count as cycles.

Determining if a network is acyclic

Algorithm for determining if the network is acyclic

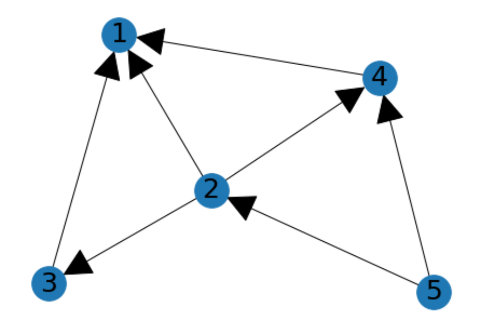
- Find a vertex with no outgoing edges.
- If no such vertex exists, the network is cyclic. Otherwise, if such a vertex does exist remove it and all its incoming edges from the network.
- If all the vertices have been removed, the network is acyclic otherwise go to step 1.

Example



Adjacency matrix for a acyclic graph is strictly triangular.

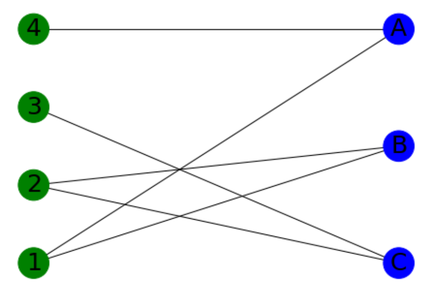
Example



Theorem: All the eigenvalues of an adjacency matrix is zero if and only if the network is acyclic.

Bipartite Networks

- A bipartite (or bigraph) is a network whose nodes can be divided into two disjoint sets U and V such that each edge connects a U node to a V node.



- A bipartite network is defined by the incidence matrix B.
- Incidence matrix (B)

$$\boxed{B_{ij} = \begin{cases} 1 & \text{if vertex } j \text{ belongs to group } i \\ 0 & \text{otherwise} \end{cases}}$$

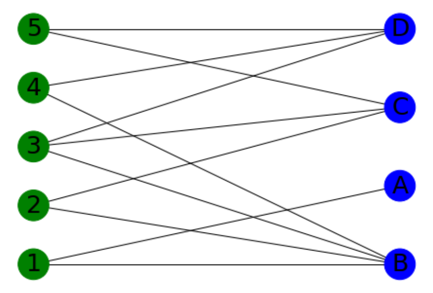
B is of size (g x n).

Bipartite Networks
One mode projection

- It is often convenient to work with direct connections between vertices of one type.
- Two projections possible
- Total number of groups that i and j belong to

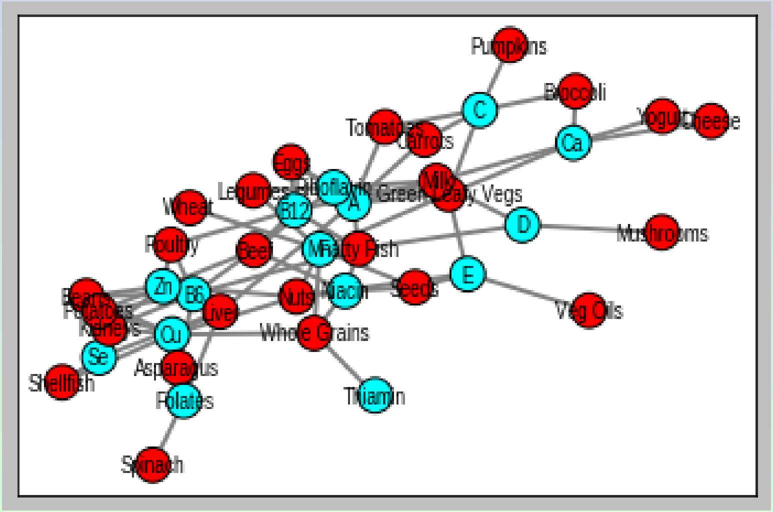
$$P_{ij} = \sum_{k=1}^{g} B_{ki} B_{kj}$$

entries



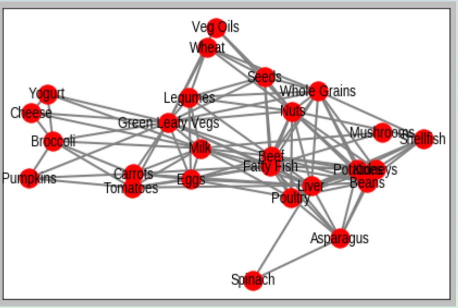
Find projection matrix for projection on groups

Bipartite Networks
One mode projection

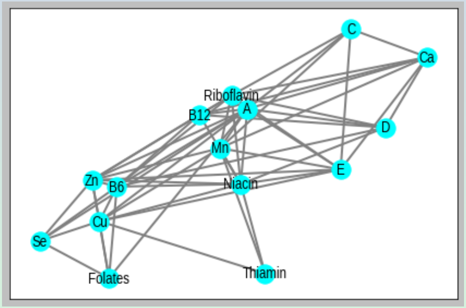


Bipartite Networks

One mode projection

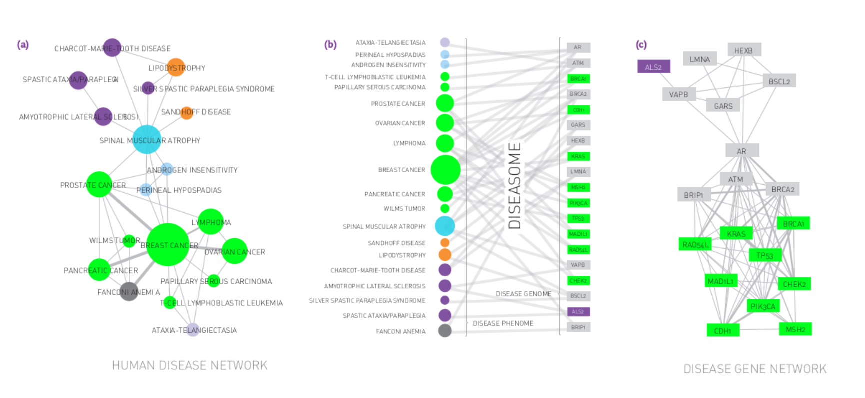


Projection on Food



Projection on nutrients

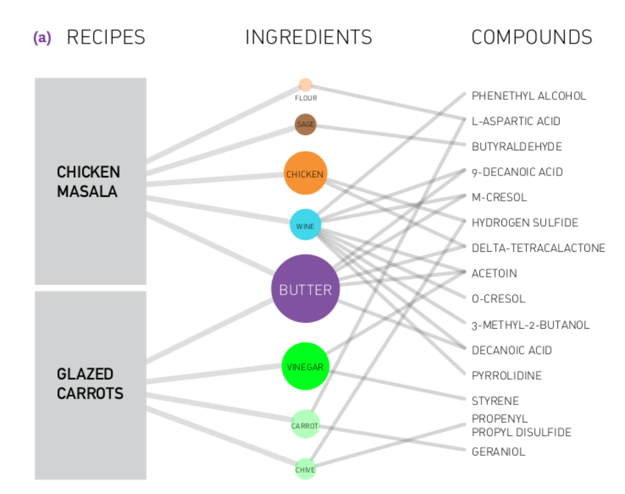
Bipartite Networks
One mode projection



Source: Network Science (Barabasi)

Bipartite Networks

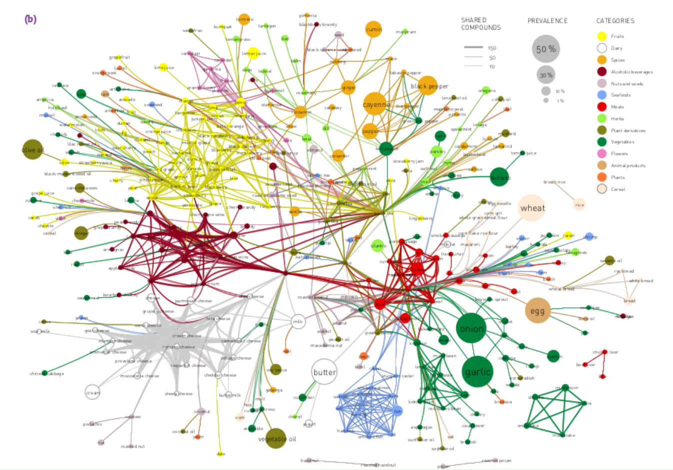
One mode projection



Source: Network Science (Barabasi)

Bipartite Networks

One mode projection



Source: Network Science (Barabasi)

Network Measures

Degree

Undirected

- Degree $k_i = \sum_j A_{ij}$
- Number of edges

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

- Mean degree $c = \frac{1}{n}\sum_i k_i = \frac{2m}{n}$
- Maximum number of possible edges in a simple graph $m_{max} = \binom{n}{2}$
- Connectance or density of a graph

$$\rho = \frac{m}{m_{max}} = \frac{2m}{n(n-1)} = \frac{c}{n-1}$$

Directed

- In-degree $k_i^{in} = \sum_j A_{ij}$
- out-degree $k_i^{out} = \sum_j A_{ji}$
- Total number of edges in a directed graph

$$m = \sum_i k_i^{in} = \sum_i k_i^{out} = \sum_{ij} A_{ij}$$

- Mean in and out degree $c_{in} = c_{out} = \frac{m}{n}$
- Density

$$\rho = \frac{m}{m_{max}} = \frac{m}{n(n-1)}$$

Degree to degree distribution

Important quantities to characterize a sample of N values $x_1, \cdots, x_n$

- Mean or average

$$\langle x \rangle = \frac{1}{N} \sum_{i=1}^{N} x_i$$

- p^{th} moment:

$$\langle x^p \rangle = \frac{1}{N} \sum_{i=1}^{N} x_i^p$$

- Standard Deviation

$$\sigma = \sqrt{\frac{1}{N} \left(x_i - \langle x \rangle\right)^2}$$

- Distribution of x

$$p_x = \frac{1}{N} \sum_i \delta_{x,x_i} \qquad \left(\sum_x p_x = 1 \quad \int_x dx \ p_x = 1 \right)$$

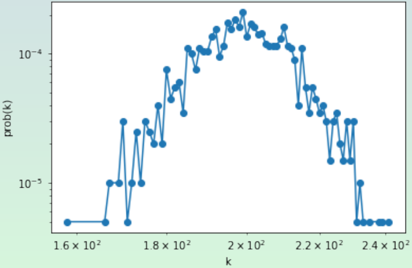
Degree to degree distribution

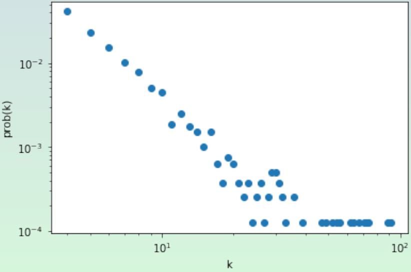
Random Graph

- Number of nodes: 4039
- Number of edges: 88234
- Average degree: 43.6910

Scale free Network

- Number of nodes: 1000
- Number of edges: 3984
- Average degree: 7.9680



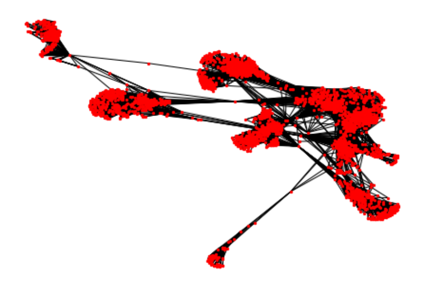


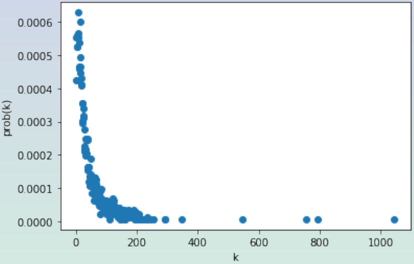
plot on linear scale

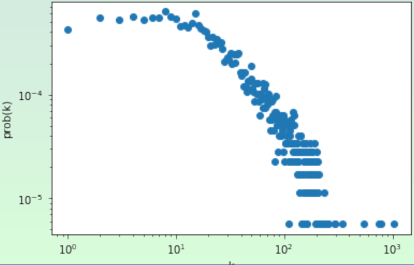
Towards real networks

Facebook Network

- Number of nodes: 4039
- Number of edges: 88234
- Average degree: 43.6910



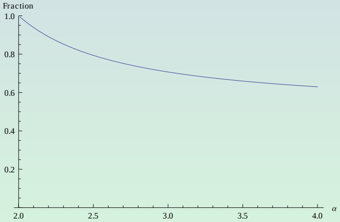




Degree distribution
power law

Properties of power law distribution

- $p(x) = Cx^{-\alpha} \quad \text{for x} > \text{x}_{\text{min}}$
- $C = (\alpha - 1)x_{min}^{\alpha-1}$
- $p(x) = \frac{\alpha-1}{x_{min}}\left(\frac{x}{x_{min}}\right)^{-\alpha}$
- Mean
$$\langle x \rangle = C\left[\frac{x_{max}^{2-\alpha} - x_{min}^{2-\alpha}}{2-\alpha}\right]$$
- r^{th} moment
$$\langle x^r \rangle = C\frac{x_{max}^{r-\alpha+1} - x_{min}^{r-\alpha+1}}{r-\alpha+1}$$
- scaling of x_{max}
$$x_{max} \sim N^{1/(\alpha-1)}$$
- Top heavy distribution(fat tailed)



Paths

Path in a network

- *Path*: A sequence of vertices such that every consecutive pair of vertices in the sequence is connected by an edge in the network.
  - Paths that do not intersect with themselves are called *self-avoiding* paths.
  - *Length* of a path is the number of edges traversed along the path.
- $A_{ij} = 1$ if there is an edge between nodes i and j.
- Total number of paths of length 2 between nodes i and j

$$\mathcal{N}_{ij}^{(2)} = \sum_{k=1}^{n} A_{ik} A_{kj} = [A^2]_{ij}$$

- Total number of paths of length r between nodes i and j

$$\mathcal{N}_{ij}^{(r)} = [A^r]_{ij}$$

Paths

Path in a network

- Total number of paths that start and end at the same vertex i (loops)

$$\mathcal{N}_{ii}^{(r)} = [A^r]_{ii}$$

- Total number of loops of length r

$$L_r \sum_{i}^{n} = \mathrm{Tr}\ [\mathrm{A}^{\mathrm{r}}]$$

[IMP]

Decomposition

- If the network is undirected, then A is symmetric. Through eigenvalue decomposition

$$L_r = \mathrm{Tr}\left[\mathrm{A}^{\mathrm{r}}\right] = \mathrm{Tr}\left[\mathrm{U}\Lambda^{\mathrm{r}}\mathrm{U}^{\mathrm{T}}\right] = \mathrm{Tr}\ \left[\mathrm{U}^{\mathrm{T}}\mathrm{U}\lambda^{\mathrm{r}}\right] = \sum_{\mathrm{i}} \lambda_{\mathrm{i}}^{\mathrm{r}}$$

- If the network is undirected via Schur decomposition

$$L_r = \mathrm{Tr}\ \mathrm{A}^{\mathrm{r}} = \mathrm{Tr}\ \mathrm{Q}\mathrm{T}^{\mathrm{r}}\mathrm{Q}^{\mathrm{T}} = \mathrm{Tr}\ \left[\mathrm{Q}^{\mathrm{T}}\mathrm{Q}\mathrm{T}^{\mathrm{r}}\right] = \sum_{\mathrm{i}} \lambda_{\mathrm{i}}^{\mathrm{r}}$$

  - T: upper triangular matrix
  - The eigenvalues of T are the same as that of A
  - Q orthogonal matrix

Average Path Length

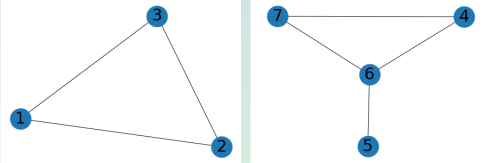
- A *geodesic* path is the *shortest* path between two vertices such that no shorter path exists.
  - The length of this path is called geodesic distance or shortest path.
  - geodesic distance (d_{ij}) between vertex i and j is the smallest value of r such that $[A^r]_{ij} > 0$
  - If two nodes are not connected by any distance their geodesic distance is infinite.
  - shortest path are self-avoiding.
- *Diameter* of a graph: length of the longest geodesic path between any pair of vertices in the network ($\max(d_{ij})$)
- Average (shortest) path length $\langle d \rangle$

$$\langle d \rangle = \frac{1}{n(n-1)} \sum_{i \neq j} d_{ij}$$

  - complete graph: $\langle d \rangle = 1$
  - Random graph: $\langle d \rangle \sim \log n$
- BFS can be used for geodesic distance.

Component and connectedness

- A network for which there exists pairs of vertices that there is no path between them is called *disconnected*.
- Component is a maximal subset of vertices in a network such that there exists at least one path from every vertex of the subgroup to another.
- The largest connected component in many real networks includes a substantial portion of the network and is called the *giant* component.
- Adjacency matrix of more than one network can be written in a *Block diagonal form*
- BFS can be used to check the connectedness of a graph and calculate the number of components.



Component and connectedness

Components in a directed graph

- Weakly connected component: Assuming no direction in the edges we can identify components as in an undirected graph.
- Strongly connected component (GSCC): Maximal subset of vertices such that there is a directed path in both directions between any pair of the vertices.
- Gian in-component: formed by nodes from which it is possible to reach GSCC by means of a directed path.
- Giant out-component: Nodes that can be reached from GSCC by means of a directed path.

The Graph Laplacian

Diffusion

- process by which gas moves from a region of high density to low density.
- model of *spread* of gas, an idea, disease etc on a network
- Suppose we have some commodity on the vertices of a network (ψ_i for vertex i)
- Rate at which the commodity moves along the edges, from a node j to a neighbor node i at a rate $C\left(\psi_j - \psi_i\right)$
- Rate at which ψ_i is changing in time

$$
\begin{aligned}
\frac{d\psi_i}{dt} &= C\sum_j A_{ij}\left(\psi_j - \psi_i\right) = C\sum_j A_{ij}\psi_j - C\psi_i\sum_j A_{ij} \\
&= C\sum_j A_{ij}\psi_j - C\psi_i k_i = C\sum_j \left(A_{ij} - \delta_{ij}k_i\right)\psi_i
\end{aligned}
$$

doubt

The Graph Laplacian

[Diffusion vadu bau clear nathi]

Diffusion

- In matrix notation
$$\frac{d\psi}{dt} = C(A - D)\psi$$
- D is a diagonal matrix with the vertex degrees along the diagonal.
- Define $L = D - A$,
$$\boxed{\frac{d\psi}{dt} + CL\psi = 0} \quad \left(\frac{d\psi}{dt} + D\frac{\partial^2\psi}{\partial x^2} = 0\right)$$
- $L_{ij} = \delta_{ij}k_i - A_{ij}$ is the graph Laplacian
- $$L_{ij} = \begin{cases} k_i & \text{if } i = j \\ -1 & \text{if } i \neq j \text{ and there is an edge } (i,j) \\ 0 & \text{otherwise} \end{cases}$$

The Graph Laplacian

Solving the Diffusion equation

- We can write ψ as the linear combination of the eigenvectors v_i of the Lapacian

$$\psi(t) = \sum_i a_i(t) v_i \qquad a_i(t) \text{time varying constant}$$

- In terms of $a_i(t)$,

$$\sum_i \left(\frac{da_i}{dt} + C\lambda_i a_i \right) v_i = 0$$

- Since v_i is not zero

$$\frac{da_i}{dt} + C\lambda_i a_i = 0$$

- Solution

$$\boxed{\psi(t) = \sum_i a_i(0) e^{-C\lambda_i t} v_i}$$

Given an initial condition we can solve for diffusion provided we know the eigenvalues and eigenvectors of the graph Laplacian

The Graph Laplacian

Eigenvalues of graph Laplacian

- L is symmetric $\Rightarrow$ Eigenvalues are *non-negative* and real.
- Consider an undirected network with n vertices and m edges. For each edge designate on one end of the edge as $+1$ and the other as -1
- Edge incidence matrix

$$B_{ij} = \begin{cases} +1 \text{ if end 1 of edge i is connected to vertex j} \\ -1 \text{ if end 2 of edge i is connected to vertex j} \\ 0 \text{ otherwise} \end{cases}$$

Doubt ?

-

$$B_{ij} = \begin{cases} \sum_k B_{ki}B_{kj} = 1 \;\; \text{if edge k connects nodes i and j } (i \neq j) \\ \sum_k B_{ki}B_{kj} = k_i \;\; \text{if } i = j \\ 0 \text{ otherwise} \end{cases}$$

- Laplacian $L = B^T B$

The Graph Laplacian

Eigenvalues of graph Laplacian

Let v_i be an eigenvector L with eigenvalue λ_i

$$Lv_i = \lambda_i v_i = B^T B v_i$$

$$\lambda_i = \left(v_i^T B^T\right)(Bv_i)$$

$$\boxed{\lambda_i \geq 0}$$

Diffusion equation on any network contains only decaying exponentials or constants $\Rightarrow$ solutions tend to an equilibrium value as $t \to \infty$

The Graph Laplacian

Components and algebraic connectivity

- Laplacian always has atleast one zero eigenvalue

$$\sum_{j} L_{ij} \times 1 = 0$$

- Suppose a network has c components, with size $n_1 \cdots n_c$
- The Laplacian can also be (like adjacency matrix) written in a block diagonal form.
- There are c eigenvectors of L with eigenvalue zero. These eigenvectors have ones in all positions corresponding to vertices in single component and zero elsewhere.
- The number of zero eigenvalues is exactly equal to the number of connected components.

The Graph Laplacian

Random Walk on undirected, connected and unweighted graph

- A *random walk* is a path across a network created by taking repeated random steps
  - Start at a specific vertex
  - Choose uniformly at random one of the edges of this vertex and move along the edge.
  - Arrive at the vertex at the other end of the vertex and repeat the above steps

The Graph Laplacian

Random Walk on undirected, connected and unweighted graph

- Let p_i be the probability that the random walk is at vertex i at time t.
- Probability equation

$$p_i(t) = \sum_j \frac{A_{ij}}{k_j} p_j(t-1)$$

- Matrix form

$$\bar{p}(t) = AD^{-1}\bar{p}(t-1)$$

- For steady state solution: $\lim_{t\to\infty} \bar{p}(t) = \bar{p}$

$$(I - AD^{-1})\bar{p} = LD^{-1}\bar{p} = 0$$

- For random walk on connected graph:

$$D^{-1}\bar{p} = \alpha\mathbf{1} \Rightarrow \bar{p} = \alpha D\mathbf{1} \Rightarrow p_i = \alpha k_i$$

- In a connected graph the probability that a random walk will be found on a vertex is proportional to its degree.