Math 168 – Intro to Networks

University of California, Los Angeles

Duc Vu

Spring 2022

This is math 168 – Introduction to Networks taught by Professor Chodrow. We meet weekly on MWF from 10:00 am to 10:50 am for lecture. The required textbook for the class is *Networks* 2^{nd} by *Newman*. Other course notes can be found at my blog site. Please let me know through my email if you spot any typos in the note.

Contents

List of Theorems

List of Definitions

§1 Lec 1: Mar 28, 2022

§1.1 Introduction

Some introduction and logistics stuffs of the class. Nothing mathy is discussed in this lecture.

§2 Lec 2: Mar 30, 2022

§2.1 Networks and Matrices

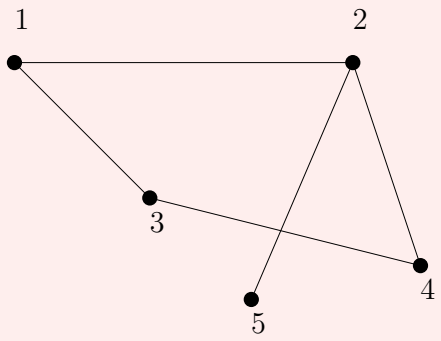
Definition 2.1 (Graph) — A (simple, undirected) graph is $G = (N, E)$, a node set N and an edge set $E \subseteq N \times N$ s.t. $i \neq j \forall (i, j) \in E$.

Definition 2.2 (Adjacency Matrix) — The adjacency matrix $\mathbf{A}$ of a graph $G = (N, E)$ is a matrix in $\mathbb{R}^{n \times n}$ where $n = |N|$ with entries

$$a_{ij} = \begin{cases} 1, & \text{if } (i, j) \in E \\ 0, & \text{otherwise} \end{cases}$$

Example 2.3

Consider the following graph



The adjacency matrix is

$$\mathbf{A} = \begin{bmatrix} 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix}$$

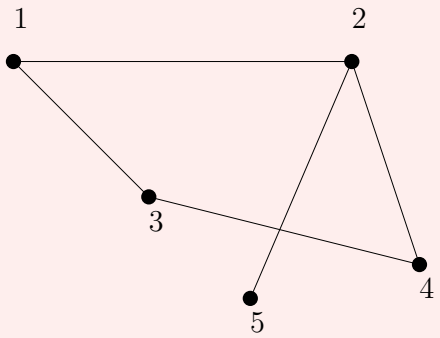
Definition 2.4 (Walk) — A walk in $G = (N, E)$ is a sequence of edges $(i_1, j_1), (i_2, j_2), \ldots, (i_k, j_k)$ where

$$j_1 = i_2, j_2 = i_3, \ldots, j_{k-1} = i_k$$

This is a walk of length k (number of edges) from i_1 to j_k.

Example 2.5

Consider the above example



Walk $3 \to 2$ of length

- $1 : \emptyset$
- $2 : (3,1),(1,2);(3,4),(4,2)$
- $3 : \emptyset$
- $4 : (3,1),(1,2),(2,1),(1,2)$

Fact 2.1. The $ijth$ entry of $\mathbf{A}$ counts the number of walks of length 1 from node i to j.

Conjecture 2.1. The $ijth$ entry of $\mathbf{A}^k$ counts the number of walks of length k from i to j.

Proof. Suppose inductively that $W(k) \triangleq \mathbf{A}^k$ has entries $w_{ij}(k)$ counting k-walks from $i \to j$. Consider $W(k+1) = W(k)A$. Its entries are

$$w_{ij}(k+1) = \sum_{l \in N} w_{il}(k) a_{lj}$$

□

§3 Lec 3: Apr 1, 2022

§3.1 Measures and Metrics

A walk of length 2, $i \leftrightarrow i$, is the number of edges attached to node $i \triangleq$ degree of node i, k_i

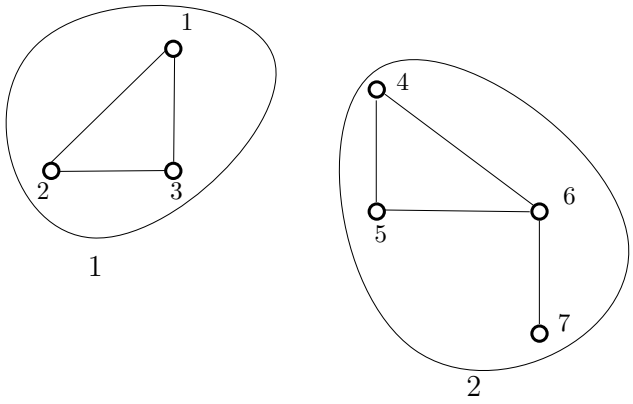
$$k_i = \sum_{j \in N} a_{ij} = \sum_{j \in N} a_{ji} = ii\text{th entry of } \mathbf{A}^2$$

Definition 3.1 (Degree) — The degree k_i of a node i is the number of edges attached to it

$$k_i = |\{j : (i,j) \in E\}|$$

Definition 3.2 (Path-connected) — Nodes i and j are path-connected if $\exists$ a walk $i \leftrightarrow j$ of any length. The connected component of i is the set of notes to which i is path-connected.
G is connected if it has 1 connected component.

Consider a disconnected graph G



Then, the adjacency graph is

$$\mathbf{A} = \begin{bmatrix} \mathbf{A}_1 & 0 \\ 0 & \mathbf{A}_2 \end{bmatrix} \text{ up to permutations of node labels}$$

Question 3.1. How big is a graph?

- Number of nodes
- Number of edges
- Diameter

Definition 3.3 (Geodesic Path) — Geodesic (shortest) path between $i \leftrightarrow j$ is a walk s.t. no walk has shorter length.

Definition 3.4 (Diameter) — Diameter of G is

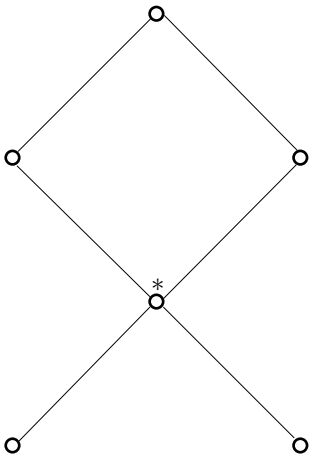
$$\max_{ij} \text{geodesic distance}(i, j)$$

which is undefined if i and j are not connected.

Example 3.5

"6 degrees of separation": in social networks, the diameter is usually about 6.

Node Importance:

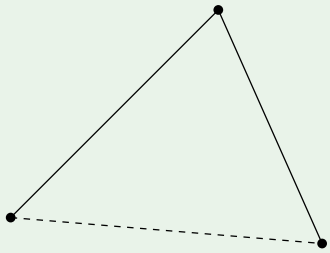


- * has highest degree
- If * were removed, graph would be disconnected
- Short average distance to other nodes
- Betweeness: # of geodesic paths passing through $i \in N$.

§4 | Lec 4: Apr 5, 2022

§4.1 Measures and Metrics (Cont'd)

Definition 4.1 (Triadic Closure) — In networks, the observation that this phenomenon



happens a lot is called triadic closure.

Calculation: # of triangles attached to node i = # of walks of length 3 $i \leftrightarrow i$ (A^3 diagonal) $\cdot\frac{1}{2} = \frac{1}{2}\sum_j \sum_k a_{ij}a_{jk}a_{ki}$.
To compute the # of possible triangles attached to i

1. Calculate k_i
2. $\binom{k_i}{2}$

Exercise 4.1. Express in terms of the adjacency matrix $\mathbf{A}$.

Definition 4.2 (Local Clustering Coefficient) — The local clustering coefficient CC_i at node i is

$$\frac{\#\text{ of triangles at } i}{\#\text{ of possible triangles}}$$

Note: $0 \leq CC_i \leq 1$.

Remark 4.3. On average, CC_i is high (many triangles can be observed) and global measures are high.

Definition 4.4 (Laplacian Matrix) — The (combinatorial) Laplacian matrix of a graph $\mathbf{L} \in \mathbb{R}^{n\times n}$

$$\mathbf{L} = \mathbf{D} - \mathbf{A}$$

where

$$\mathbf{D} = \begin{bmatrix} k_1 & & & \\ & k_2 & & \\ & & \ddots & \\ & & & k_n \end{bmatrix}$$

Definition 4.5 (Clustering/Partition) — A clustering/partition of a graph is a partition of N, $\{C_1, C_2, \ldots, C_l\}$

$$N = \bigcup_{j=1}^{l} C_j, \qquad C_j \cap C_{j'} = \emptyset \text{ if } j \neq j'$$

Let $c_i \triangleq$ cluster of node i.

Definition 4.6 (Cut Value) — The cut value of a partition $\{C_1, \ldots, C_l\}$ is

$$\frac{1}{2} \sum_{i,j \in N} a_{ij} \underbrace{\mathbb{1}[c_i \neq c_j]}_{\substack{=1,\ c_i \neq c_j \\ =0 \text{ otherwise}}}$$

Idea: Good clustering have small cut values.
Setting: 2 clusters $\{C_1, C_2\}$

$$s \in \mathbb{R}^n, \quad s_i = \begin{cases} +1, & c_i = 1 \\ -1, & c_i = 2 \end{cases}$$

Theorem 4.7 (Laplacian Formula for Cuts)
The cut value of $\{C_1, C_2\}$ is $\frac{1}{4} s^\top \mathbf{L} s$.

Proof. Consider

$$\begin{aligned} x^\top \mathbf{L} x &= x^\top (\mathbf{D} - \mathbf{A}) x \\ &= \sum_{i \in N} k_i x_i^2 - \sum_{i,j \in N} a_{ij} x_i x_j \\ &= \sum_{i,j \in N} a_{ij} x_i^2 - \sum_{i,j \in N} a_{ij} x_i x_j \\ &= \frac{1}{2} \left(\sum_{i,j \in N} a_{ij} x_i^2 + \sum_{i,j \in N} a_{ij} x_j^2 - 2 \sum_{i,j \in N} a_{ij} x_i x_j \right) \\ &= \frac{1}{2} \sum_{i,j \in N} a_{ij} (x_i - x_j)^2 \end{aligned}$$

So

$$\begin{aligned} s^\top \mathbf{L} s &= \frac{1}{2} \sum_{i,j \in N} a_{ij} (s_i - s_j)^2 \\ &= \frac{1}{2} \sum_{i,j \in N} a_{ij} \left(4 \mathbb{1}[c_i \neq c_j] \right) \end{aligned}$$

□

§5 Lec 5: Apr 6, 2022

§5.1 Erdos-Renyi Random Graph

Definition 5.1 (Random Graph) — Random graph is a probability distribution over graphs.

Definition 5.2 — An Erdos-Renyi random graph on n nodes with edges probability p is written $G(n,p)$.

To sample, we take each pair of nodes and draw an edge between them i.i.d with probability p.

Question 5.1. How many pairs are there?

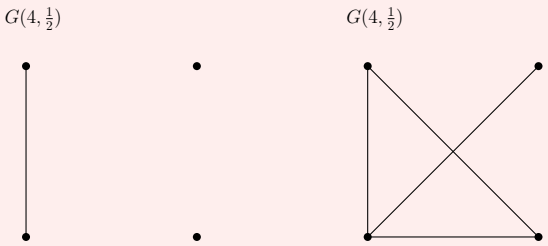
There are $\binom{n}{2}$. Also,

$$\mathbb{E}\left[\#\text{ of edges}\right] = p\binom{n}{2}$$

So # edges $\sim$ Binomial$\left(\binom{n}{2}, p\right)$.

Example 5.3

Consider



Question 5.2. What is the average degree c?

We can see that $c = \frac{1}{2}$ for the left graph and $c = 2$ for the right graph.

Note: The average degree can be calculate as $c = \frac{2m}{n}$ where m is the number of edges and n is the number of nodes.

Expected # of Triangles in ER

$$\mathbb{E}\left[\#\text{ of triangles}\right] = \binom{n}{3}p^3$$

where each edge is independent. Recall the global clustering coefficient is

$$C = \frac{\#\text{ of triangles} \cdot 3}{\#\text{ of wedges}}$$

where a wedge is a graph with 3 nodes and 2 edges. Then,

$$\mathbb{E}\left[\#\text{ of wedges}\right] = 3\binom{n}{3}p^2$$

Note that

$$\mathbb{E}[C] \neq \frac{\mathbb{E}[\triangle] \cdot 3}{\mathbb{E}[\#\text{ of wedges}]} = p$$

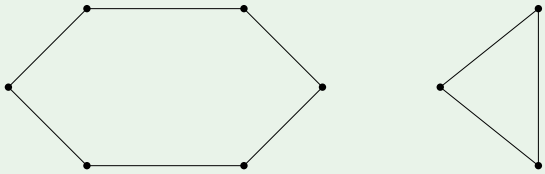
Fact 5.1. As $n \to \infty$, $C \xrightarrow{\text{in dist.}} p$.

Node degree in ER

$$\mathbb{E}[k_i] = (n-1)p = c$$

As p increase, $\mathbb{E}[k_i]$ also increases (linearly). However, this is not always the case. Consider sparse ER with $p = \frac{c}{n-1}$. So, as $n \to \infty$, $p = \frac{c}{n-1} \to 0$, i.e., sparse ER model has very little clustering.

Definition 5.4 (Cycle) — A cycle on node i is a walk from $i \leftrightarrow i$ with no repeated nodes or edges.



We can observe that 3-cycles are rare in large sparse ER ($n \to \infty$).

§6 Lec 6: Apr 8, 2022

§6.1 Paths and Branching Processes in ER Random Graphs

Definition 6.1 (Path) — A path is a walk with no node repetitions.

Path lengths: Pick $i, j \in N$. Define $R_k \triangleq$ # of paths of length k between i and j. Let's compute $r_k = \mathbb{E}[R_k]$.

$$\mathbb{E}[R_k] \approx \mathbb{E}[\#(\text{ path to } l \text{ and } (l,j) \in E)] = \mathbb{E}[\#\text{ of length to } l \text{ of length } k-1]\, p = r_{k-1}p$$

So

$$r_k \approx \underbrace{r_{k-1}p}_{\text{for path through } l} \quad \underbrace{(n-2)}_{\#\text{ of ways to choose } l \neq i,j}$$

Also, notice that

$$r_k \approx r_{k-1}p(n-1) = r_{k-1}c = c^{k-1}r_1 = c^{k-1}p$$

Question 6.1. What length k makes path likely?

We have

$$\log r_k \approx (k-1)\log c + \underbrace{\log p}_{\log c - \log n}$$

$$k \approx \frac{\log r_k + \log n}{\log c}$$

Assume $r_k = 1$. Then, consider the world population of 8 billions with average degree of 1000

$$k \approx \frac{\log n}{\log c} \approx \frac{\log 8 \cdot 10^9}{\log 10^3} \approx 3.4$$

Notice that if $c \leq 1$, the expression above doesn't make any sense.
Galton-Waston Branching Process

Definition 6.2 (Branching Process) — Let p be a probability distribution on $\mathbb{Z}$, called the offspring distribution. A branching process with distribution p is a sequence of random variables $X_0, X_1, X_2, \ldots$ s.t. $X_0 = 1$ and for $t \geq 1$,

$$X_t = \sum_{i=1}^{X_{t-1}} Y_i$$

where each Y_i is distributed i.i.d. according to p.

Branching processes create tree-graphs without cycles, which we can utilize to better understand the behavior of ER random graph.

§7 | Lec 7: Apr 11, 2022

§7.1 Giant Component in Sparse Erdos-Renyi

Fact 7.1. Say we have a Poisson(c) process, then

$$\mathbb{E}[X_k] = c^k$$

Total number of individuals in $\mathbb{E}$

$$\mathbb{E}\left[\sum_{k=0}^{\infty} X_k\right] = \sum_{k=0}^{\infty} \mathbb{E}[X_k] = \sum_{k=0}^{\infty} c^k = \begin{cases} \frac{1}{1-c} & 0 < c < 1 \\ \text{divergent } \text{“}\infty\text{”} & c \geq 1 \end{cases}$$

Let's consider

$$\begin{aligned} P\left(\frac{\text{size of component containing node } i}{n} > a\right) &= P(\text{size} > an) \\ &\leq \frac{\mathbb{E}[\text{size}]}{an} \qquad \text{(Markov's)} \\ &= \frac{1}{an}\frac{1}{1-c} \to 0 \text{ unless } a = 0 \end{aligned}$$

In the case of $a = 0$, $P\left(\frac{\text{size}}{n} > 0\right) = 1$.

Definition 7.1 (Giant Component) — A sequence $G\left(n, \frac{c}{n-1}\right)$ as $n \to \infty$ has a giant component(GC) if

$$P\left(\frac{\text{component containing random node } i}{n} > a(>0)\right) \geq b > 0$$

In other words,

$$\mathbb{E}[\text{size of largest component}] = an \text{ for some } 0 < a \leq 1$$

Fact 7.2. Sequence $G\left(n, \frac{c}{n-1}\right)$ has a giant component if and only if $c > 1$.

Let u be the probability P that a node is not in giant component, $s = 1 - u$ is probability that a node is in giant component, sn = size of giant component.

$$u = \left(\underbrace{1-p}_{\text{not connected}} + \underbrace{pu}_{\text{connected not in GC}}\right)^{n-1}$$

Let's simplify the above expression.

$$\begin{aligned} u &= (1 - p(1-u))^{n-1} \\ &= \left(1 - \frac{c(1-u)}{n-1}\right)^{n-1} \\ &= e^{-c(1-u)} \text{ as } n \to \infty \end{aligned}$$

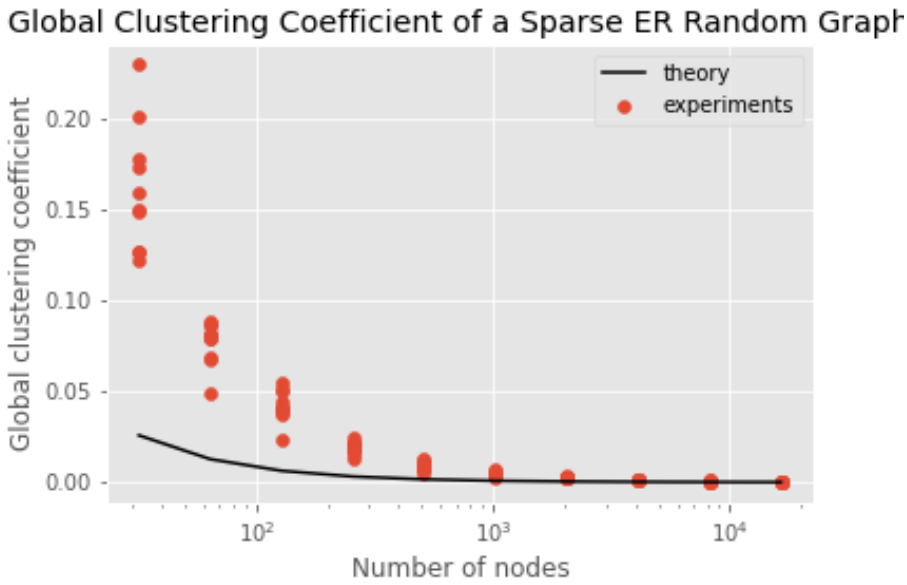
Replace $s = 1 - u$

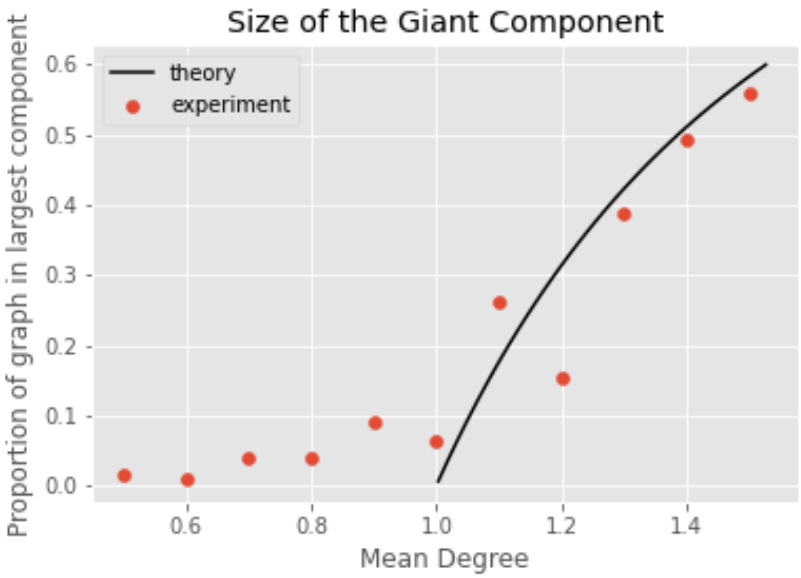
$$s = 1 - e^{-cs}$$

§8 Lec 8: Apr 13, 2022

§8.1 Experimental Lecture on ER Theory

In this lecture, we did some experiments with Python to check whether they agree with the theoretical results that we discussed previously on ER random graph.





§9 Lec 9: Apr 15, 2022

§9.1 Configuration Model

Definition 9.1 (Degree Sequence) — The degree sequence of $G = (N, E)$ with $|N| = n$ is $\vec{k} \in Z^n$ s.t. degree of $i \in N = k_i$.

Definition 9.2 (Configuration Model Random Graph) — The configuration model random graph with degree sequence $\vec{k}$ is a uniformly random graph among all graph with degree sequence $\vec{k}$.

Stub-Matching:

Select uniformly random pairs of half edges and turn them into edges until we run out of edge pair (we then have a graph). However, this method is not perfect as we can have a problem with self-loop or parallel edges, i.e., we only want simple graphs.

Fact 9.1. For $n \to \infty$, if the degree sequence doesn't grow in its entries (sparsity), then $P(\text{simple graph}) > \varepsilon > 0$.

Fact 9.2. Stub matching (conditioned on getting a simple graph) samples from configuration model.

Moment of the degree sequence:

Definition 9.3 — Degree distribution $p_k = P(\text{random node has degree } k) = \frac{\#\text{ nodes of degree } k}{n}$.

The l^{th} moment is defined as

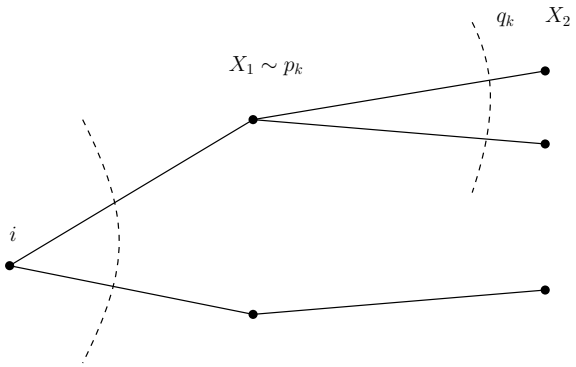
$$\langle k^l \rangle \triangleq \sum_k p_k k^l$$

So

$$\langle k^0 \rangle = \sum_k p_k = 1$$

$$\langle k^1 \rangle = \sum_k p_k k$$

Branching Process:



$$\mathbb{E}[X_1] = \langle k \rangle$$

and

$$\begin{aligned} q_k &= P\left(\text{if I follow an edge to node j, the number of additional edges on j} = k\right) \\ q_k &= \frac{(k+1)p_{k+1}n}{\#\text{ of half-edges=2m}} \\ &= \frac{(k+1)p_{k+1}}{\langle k \rangle} \end{aligned}$$

Then,

$$\begin{aligned} \mathbb{E}\left[\#\text{ offspring in 2nd gen from single parent}\right] &= \sum_{k=0} k q_k = \sum_{k=0} \frac{k(k+1)p_{k+1}}{\langle k \rangle} \\ &= \sum_{k'=1} \frac{(k'-1)k' p_{k'}}{\langle k \rangle} \\ &= \frac{1}{\langle k \rangle} \sum_{k'=1} \left(k'^2 - k'\right) p_{k'} \\ &= \frac{1}{\langle k \rangle} \left(\langle k^2 \rangle - \langle k \rangle\right) \end{aligned}$$

Branching heuristic for giant component: Giant component iff $\frac{\langle k^2 \rangle - \langle k \rangle}{\langle k \rangle} > 1$.

§10 | Lec 10: Apr 18, 2022

§10.1 Configuration Model (Cont'd)

From Cauchy-Schwarz, $\langle k^2 \rangle \geq \langle k \rangle^2$. In real world social networks, $\langle k^2 \rangle \gg \langle k^2 \rangle$ (heterogeneous degree). For example, say we have vaccine for 1% of population, and we want to vaccinate high degree individuals but in reality we don't know their degree. The problem here is we don't have network data/structure, and our assumption is $\langle k^2 \rangle \gg \langle k \rangle^2$.
Instead of randomly picking people to get vaccine, we encourage people to nominate a friend to get the vaccine (walk along social network).

§11 Lec 11: Apr 20, 2022

§11.1 Modularity Maximization

Definition 11.1 (Modularity) — The modularity of graph G and cluster labels z_i for each node, $\vec{z} \in \mathbb{R}^n$ with respect to random graph model M

$$Q(G, z) = \frac{1}{2m} \sum_{i,j \in N} \left[a_{ij} - \mathbb{E}_m \left[A_{ij}\right]\right] \delta(z_i, z_j)$$

where

$$\delta(x, y) = \begin{cases} 1, & x = y \\ 0, & x \neq y \end{cases}$$

Consider an ER random graph $G(n, p)$

$$\mathbb{E}\left[A_{ij}\right] = P\left(A_{ij} = 1\right) = p$$

Now, consider the configuration model, we know the degrees $k_1, k_2, \ldots, k_n$

$$\sum_{i=1}^{n} k_i = 2m$$

Then,

$$\mathbb{E}\left[A_{ij}\right] = \left(\frac{k_i}{2m} \frac{k_j}{2m-1}\right) 2m \approx \frac{k_i k_j}{2m}$$

So we can substitute the expression above into the modularity formula

$$Q = \frac{1}{2m} \sum_{i,j} \left(a_{ij} - \frac{k_i k_j}{2m}\right) \delta(z_i, z_j)$$

which is known as the standard modularity.
Now, let's dig into how to maximize the modularity. We need to find cluster/communities in G by solving

$$\hat{z} = \underset{z}{\operatorname{argmax}}\, Q(G, z)$$

Assume we have n nodes and 2 groups, then there are 2^n candidate solutions. However, this is NP-hard problem. We must use heuristics. First, let's turn $\delta(z_i, z_j)$ into some expression that involves linear algebra.

$$s_i \triangleq \begin{cases} +1, & z_i = 1 \\ -1, & z_i = 2 \end{cases}$$

$$\begin{aligned} \delta(z_i, z_j) &= \begin{cases} 1, & z_i = z_j \\ 0, & \text{otherwise} \end{cases} \\ &= \frac{1}{2}\left(s_i s_j + 1\right) \end{aligned}$$

So

$$\begin{aligned} Q &= \frac{1}{2m}\sum_{i,j}\left(a_{ij} - \frac{k_i k_j}{2m}\right)\frac{1}{2}\left(s_i s_j + 1\right) \\ &= \frac{1}{4m}\sum_{i,j}\underbrace{\left(a_{ij} - \frac{k_i k_j}{2m}\right)}_{b_{ij}} s_i s_j + \underbrace{\frac{1}{4m}\sum_{i,j}\left(a_{ij} - \frac{k_i k_j}{2m}\right)}_{=0} \\ &= \frac{1}{4m}\sum_{i,j} b_{ij} s_i s_j \end{aligned}$$

Definition 11.2 (Modularity Matrix) — The modularity matrix $\mathbf{B} \in \mathbb{R}^{n\times n}$ has entries

$$b_{ij} = a_{ij} - \frac{k_i k_j}{2m}$$

This allows to write

$$Q = \frac{1}{4m}\vec{s}^\top \mathbf{B}\vec{s}$$

Let s have any entries, solve $\max_s \vec{s}^\top \mathbf{B}\vec{s}$, set

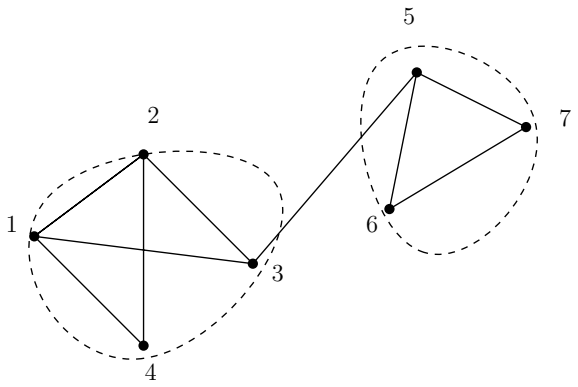
$$z_i = \begin{cases} 1, & s_i < 0 \\ 2, & s_i \geq 0 \end{cases}$$

So from homework 0, we know that $s =$ 1st eigenvector of $\mathbf{B}$.

§12 | Lec 12: Apr 22, 2022

§12.1 Modularity Maximization (Cont'd)

Consider a community detection problem



Say we find a label vector $\vec{z} = (1, 1, 1, 1, 2, 2, 2)$. In modularity maximization, our goal is to pick $\vec{z}$ to maximize

$$Q = \frac{1}{2m} \sum_{i,j \in N} \left[a_{ij} - \frac{k_i k_j}{2m} \right] \delta(z_i, z_j)$$

Let l be a label. Let's define

$$e_l \triangleq \frac{1}{2m} \sum_{i,j \in N} a_{ij} \delta(z_i, l) \delta(z_j, l) = \% \text{ of all edges w/ both ends in community } l$$

and

$$f_l \triangleq \frac{1}{2m} \sum_{i \in N} k_i \delta(z_i, l) = \% \text{ of edges that end in cluster } l$$

So from the above figure, we can see that

$$e_1 = \frac{1 \cdot 10}{2 \cdot 9} = \frac{5}{9}$$
$$f_2 = \frac{1 \cdot 7}{18} = \frac{7}{18}$$

So we can rewrite Q as follows

$$\begin{aligned} Q &= \frac{1}{2m} \sum_{i,j} \left[a_{ij} - \frac{k_i k_j}{2m} \right] \delta(z_i, z_j) \\ &= \frac{1}{2m} \left[\sum_l \sum_{i,j} a_{ij} \delta(z_i, l) \delta(z_j, l) \right] \\ &= \frac{1}{2m} \left[\sum_l \sum_{i,j} \frac{k_i \delta(z_i, l) k_j \delta(z_j, l)}{2m} \right] \\ &= \sum_l \left[e_l - f_l^2 \right] \end{aligned}$$

Consider $\max -\sum_l f_l^2$ or $\min \sum_l f_l^2$ s.t. $\sum_l f_l = 1$ and $f_l \geq 0$. Then, by using Lagrange multiplier, we have

$$\begin{aligned} \nabla \sum_l f_l^2 &= \lambda \nabla \sum_l f_l \\ 2 \begin{pmatrix} f_1 \\ f_2 \\ \vdots \end{pmatrix} &= \lambda \begin{pmatrix} 1 \\ 1 \\ \vdots \end{pmatrix} \\ \implies f_l = f_{l'} &\quad \text{for } l \neq l' \end{aligned}$$

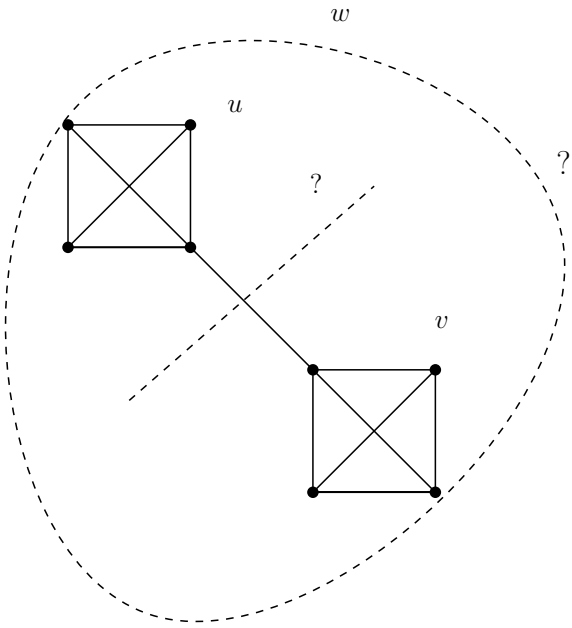
Thus, modularity maximization says

1. Try to make lots of in-cluster edge ($\max e_l$)
2. Try to make the cluster similar sizes (f_l)

§13 Lec 13: Apr 25, 2022

§13.1 Resolution Limit

Modularity maximization can't find communities that are "too small" relative to graph size.



Let ΔQ be the change in Q due to merging u and v into w.

$$\Delta Q = \underbrace{e_w - (e_u + e_v)}_{\frac{1}{2m}} - \left(f_w^2 - \left(f_u^2 + f_v^2\right)\right)$$

Note that

$$f_u = \frac{1}{2m}(\text{sum of node degrees in cluster } u) = \frac{1}{2m}\left[(k-1)k+1\right] = f_v \triangleq \frac{s}{2m}$$

and

$$f_w = \frac{1}{2m} 2s \,(= f_u + f_v)$$

So when is $\Delta Q > 0$?

$$\frac{1}{2m} - \frac{(2s)^2}{(2m)^2} + 2\frac{s^2}{(2m)^2} > 0$$
$$2m > s^2$$

Example 13.1

Consider a graph with $n = 5 \times 10^6$ and $c = 20$

$$\implies 2m = nc = 10^8$$

We need $s > \sqrt{2m} = 10^4$ where $s = k^2 - k + 1$. So roughly speaking, $k > 100$ to detect k-clique communities in this graph.

§14 | Lec 14: Apr 27, 2022

§14.1 Random Walks on Graphs

Simple Random Walk:

Start at node i. Pick a neighbor of i uniformly at random and move there. Repeat this process infinitely.

Definition 14.1 (Simple Random Walk) — A simple random walk on graph G is a countable sequence of random variables $X_1, \dots, X_t, \dots$ with values in N ($X_t = i$ implies we are at node i at time t). The distribution of X_{t+1}

$$P(X_{t+1} = i | X_t = j_t, X_{t-1} = j_{t-1}, \dots, X_0 = j_0) = P(X_{t+1} = i | X_t = j_t) = \begin{cases} \frac{1}{k_{j_t}} & (j_t, i) \in E \\ 0 & (j_t, i) \notin E \end{cases}$$
$$= \frac{a_{ij_t}}{k_{j_t}}$$

Definition 14.2 (Transition Matrix) — The transition matrix of a simple random walk is $\mathbf{P} = \mathbf{A}\mathbf{K}^{-1}$ where

$$\mathbf{K} = \begin{pmatrix} k_1 & & \\ & \ddots & \\ & & k_n \end{pmatrix}$$

and $p_{ij} = \frac{a_{ij}}{k_j}$.

Consider

$$P(X_{t+1} = i) = \sum_{j \in N} P(X_{t+1} = i | X_t = j) P(X_t = j)$$
$$q_i(t+1) = \sum_{j \in N} p_{ij} q_j(t)$$

So

$$\vec{q}(t+1) = \mathbf{P}\vec{q}(t) = \mathbf{P}^{t+1} q(0)$$

Definition 14.3 (Stationary Distribution) — A simple random walk has a stationary distribution $\vec{\pi} \in \mathbb{R}^n$ if $\lim_{t \to \infty} \vec{q}_i(t) = \vec{\pi}_i$, regardless of the starting point.

Definition 14.4 (Ergodic Graph) — A graph is ergodic if

1. it is connected and
2. it is aperiodic (gcd of cycle length = 1)

Theorem 14.5

A simple random walk on an ergodic graph has a unique stationary distribution $\vec{\pi}$. Furthermore, $\vec{\pi}$ is the unique solution of $\vec{\pi} = \mathbf{P}\vec{\pi}$.

Proof. Use Perron-Frobenius Theorem. □

Structure of $\vec{\pi}$: Recall $\mathbf{P} = \mathbf{A}\mathbf{K}^{-1}$. We want to show

$$\vec{\pi} = \mathbf{A}\mathbf{K}^{-1}\vec{\pi}$$

Guess: $\vec{\rho} = \vec{k}$. Then, let's check.

$$\mathbf{A}\mathbf{K}^{-1}\vec{\rho} = \mathbf{A}\mathbf{K}^{-1}\vec{k} = \mathbf{A}\mathbf{1} = \vec{k} = \vec{\rho}$$

This is not normalized, so we can deduce that $\vec{\pi} = \frac{1}{2m}\vec{k}$.

§15 | Lec 15: Apr 29, 2022

§15.1 PageRank

Definition 15.1 (Directed Adjacency Matrix) — Directed adjacency matrix $\mathbf{A} \in \mathbb{R}^{n \times n}$

$$a_{ij} = \begin{cases} 1 & \text{if } j \to i \\ 0 & \text{otherwise} \end{cases}$$

Consequently, $\mathbf{A}$ is, in general, not symmetric.

Definition 15.2 ((Directed) Degree) — We define

$$k_i^{\text{in}} = \sum_{j \in N} a_{ij}$$
$$k_i^{\text{out}} = \sum_{j \in N} a_{ji}$$

Directed Random Walk: From node j, follow a random outgoing arrow to the next node

$$P\left(X_{t+1} = i | X_t = j\right) = \frac{a_{ij}}{k_j^{\text{out}}} \triangleq p_{ij}$$

Then, we define the transition matrix as follows

$$\mathbf{P} = \mathbf{A}(\mathbf{K}^{\text{out}})^{-1}$$

where

$$\mathbf{K}^{\text{out}} = \begin{bmatrix} k_1^{\text{out}} & & \\ & \ddots & \\ & & k_n^{\text{out}} \end{bmatrix}$$

in which we assume $k_i^{\text{out}} \geq 1 \ \forall i$.

Theorem 15.3

Suppose that there exists integer $t > 0$ s.t. $\mathbf{A}^t$ has all positive entries. Then the directed random walks has a stationary distribution $\vec{\pi}$, and $\mathbf{P}\vec{\pi} = \vec{\pi}$.

Definition 15.4 (PageRank) — With probability $1 - \alpha$, take a directed random walk step with probability α, teleport somewhere else. Note that

- $\alpha \in [0, 1]$ is the teleportation rate
- $\vec{v} \in \mathbb{R}^n$ is the teleportation vector (assume $\vec{v}$ is entry-wise positive, $\sum v_i = 1$)

This walk has the probability transition

$$P\left(X_{t+1} = i | X_t = j\right) = (1 - \alpha)\frac{a_{ij}}{k_j^{\text{out}}} + \alpha v_i = \tilde{p}_{ij}$$

We define

$$\tilde{\mathbf{P}} = (1-\alpha)\mathbf{P} + \alpha\mathbf{V}$$

Traditionally, $\alpha = 0.15$ and $\vec{v} = \frac{1}{n}\begin{pmatrix}1 & 1 & \dots & 1\end{pmatrix}^\top$. If we choose this particular choice, PakeRank has a stationary distribution. Notice that $\pi_i > 0$ where $\vec{\pi} = \tilde{\mathbf{P}}\vec{\pi}$.

§16 Lec 16: May 2, 2022

§16.1 Agent-Based Modeling

Coding session :)

§17 | Lec 17: May 4, 2022

§17.1 Opinion Dynamics

figure here

Each node i has opinion $x_i \in [-1, 1]$. For example,

$$\begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{pmatrix} = \begin{pmatrix} 0.5 \\ -0.7 \\ 0.2 \\ -0.3 \\ 0.0 \end{pmatrix} = \vec{x}(t)$$

Then,

$$\vec{x}(t+1) = F\left(\vec{x}(t)\right)$$

This is discrete-time deterministic (not random) synchronous (all nodes update simultaneously) model, and our function F depends on the graph structure.

$$F\left(\vec{x}(t)\right) = \begin{pmatrix} f_1\left(\vec{x}(t)\right) \\ f_2\left(\vec{x}(t)\right) \\ \vdots \\ f_n\left(\vec{x}(t)\right) \end{pmatrix}$$

Note that $f_i\left(\vec{x}(t)\right)$ is a function only of the neighbors of node i and i itself.

$$x_i(t+1) = f_i\left(\vec{x}(t)\right) = (1-\beta)x_i(t) + \beta \underbrace{\frac{1}{k_i}\sum_{j\sim i} x_j(t)}_{\text{average of neighbor opinion}}$$

Notice that $\sum_{j\sim i} x_j(t) = \sum_{j\in N} a_{ij}x_j(t) = (\mathbf{A}\vec{x})_i$ and $(1-\beta)x_i(t) = ((1-\beta)\mathbf{I}\vec{x})_i$. Then,

$$\vec{x}(t+1) = \left[(1-\beta)\mathbf{I} + \beta\left(\mathbf{K}^{-1}\mathbf{A}\right)\right]\vec{x}$$

§18 | Lec 18: May 6, 2022

§18.1 Opinion Dynamics (Cont'd)

Recall

$$\vec{x}(t+1) = (1-\beta)\left[\mathbf{I}\vec{x}(t)\right] + \beta\left[\mathbf{K}^{-1}\mathbf{A}\vec{x}(t)\right]$$

Then, we have

$$\begin{aligned}\vec{x}(t+1) - \vec{x}(t) &= \beta\left(\mathbf{K}^{-1}\mathbf{A} - \mathbf{I}\right)\vec{x}(t)\\ &= \beta\mathbf{K}^{-1}\left(\mathbf{A} - \mathbf{K}\right)\vec{x}(t)\\ &= \beta\mathbf{K}^{-1}(-\mathbf{L})\vec{x}(t)\\ &= -\beta\overline{\mathbf{L}}\vec{x}(t)\end{aligned}$$

Fact 18.1. $\vec{x}(t) \overset{t\to\infty}{\longrightarrow} \vec{x}^*$ where $\overline{\mathbf{L}}\vec{x}^* = \vec{0}$ and $\vec{x}^*$ is unique if G is connected.

Suppose $\overline{\mathbf{L}}\vec{x} = \vec{0}$, and in particular

$$\begin{aligned}\mathbf{K}^{-1}(\mathbf{K} - \mathbf{A})\vec{x} &= \vec{0}\\ (\mathbf{K} - \mathbf{A})\vec{x} &= \vec{0}\end{aligned}$$

So $\mathbf{L}\vec{x}^* = \vec{0}$. Thus,

$$\vec{x}^* = \gamma\vec{1}$$

Fact 18.2. On connected graphs, linear consensus dynamics converges to consensus, i.e., $x_i^* = x_j^*$ for all $i, j \in N$.

Model Modifications

- β depends on node where some nodes have $\beta_i = 0$
- Interaction depends on x_i and x_j (Hegselmann-Krause)
- Introduce noise

§19 | Lec 19: May 9, 2022

§19.1 Opinion Model Implementation

Coding session :)

§20 | Lec 20: May 11, 2022

§20.1 Midterm

§21 | Lec 21: May 13, 2022

§21.1 Intro to Epidemics on Networks

Coding session :D

§22 Dis 1: Mar 29, 2022

§22.1 Review of Linear Algebra

Networks can be represented as ordinary matrices and the related graph Laplacian so linear algebra will be very important for us.

Example 22.1

Spectral graph theory techniques.

Definition 22.2 (Matrix Kernel) — The kernel of a matrix $\mathbf{A}$ denoted $\ker(\mathbf{A})$ is the set of vectors $\vec{x}$ s.t. $\mathbf{A}\vec{x} = \mathbf{0}$.

Definition 22.3 (Range) — The range of a matrix $\mathbf{A}$ denoted $\text{im}(\mathbf{A})$ is the set of vectors $\vec{x}$ s.t. there exists a vector $\vec{y}$ s.t. $\mathbf{A}\vec{y} = \vec{x}$.

Definition 22.4 (Eigenvalue) — λ is an eigenvalue of $n \times n$ matrix $\mathbf{A}$ with right-eigenvector (usually just called eigenvector if there's no chance of confusion) $\vec{x} \neq \mathbf{0}$ if $\mathbf{A}\vec{x} = \lambda\vec{x}$.
It has left-eigenvector $\vec{y} \neq \mathbf{0}$ if $\vec{y}^\top \mathbf{A} = \lambda\vec{y}^\top$ or equivalently $\mathbf{A}^\top\vec{y} = \lambda\vec{y}$.

Note: If $\vec{x}$ is a left or right eigenvector with eigenvalue λ, then so is $\alpha\vec{x}$ for any scalar $\alpha \neq 0$.

Definition 22.5 (Spectral Radius) — The spectral radius for an $n \times n$ matrix $\mathbf{A}$ denoted $\rho(\mathbf{A})$ is the maximum magnitude of its eigenvalues

$$\rho(\mathbf{A}) = \max\{|\lambda| : \lambda \text{ is an eigenvalue of } \mathbf{A}\}$$

Definition 22.6 (Span) — The span of a set of vectors is the set of all linear combinations of those vectors

$$\text{span}\{v_1, v_2, v_3\} = \{a_1v_1 + a_2v_2 + a_3v_3\}$$

where a_1, a_2, a_3 are scalars.

Question 22.1. How do we calculate eigenvalues/eigenvectors?

1. Calculate the characteristics polynomial
$$p_{\mathbf{A}}(\lambda) = \det(\mathbf{A} - \lambda\mathbf{I}_n)$$
2. The eigenvalues are the roots $\lambda_1, \ldots, \lambda_n$ of the characteristic polynomial
3. Calculate the eigenvectors, i.e., we want to solve for $\vec{x}$ s.t. $(\mathbf{A} - \lambda\mathbf{I})\vec{x} = \mathbf{0}$.

Theorem 22.7 (Spectral for Real Matrices)

Let $\mathbf{A}$ be a $n \times n$ symmetric matrix. Then $\mathbf{A}$ has real eigenvalues $\lambda_1, \lambda_2, \ldots, \lambda_n$ (not necessarily distinct) and there exists corresponding eigenvectors $\{v_1, v_2, \ldots, v_n\}$ that form an orthonormal basis for $\mathbb{R}^n$.

Furthermore, $\mathbf{A}$ is diagonalizable here, i.e., $\mathbf{QDQ}^\top = \mathbf{A}$ where $\mathbf{D}$ is the diagonal matrix of eigenvalues and $\mathbf{Q}$ is its corresponding matrix of eigenvectors. This theorem also applies to the adjacency matrix of any undirected network since it will be symmetric.

Theorem 22.8 (Perron-Frobenius Version 1)

If $\mathbf{A}$ is a $n \times n$ real matrix with all non-negative entries, then $\mathbf{A}$ has a non-negative leading eigenvalue κ_1 with

$$\kappa_1 = \rho(\mathbf{A}) = \max\{|\lambda| : \lambda \text{ is an eigenvalue of } \mathbf{A}\}$$

So for this leading eigenvalue κ_1, all other eigenvalues λ of $\mathbf{A}$ satisfy $|\lambda| \leq \kappa_1$.

In addition there exists left and right eigenvectors of κ_1 with all non-negative entries.

Proof. Hw 0 # 3a. □

Theorem 22.9 (Perron-Frobenius Version 2)

If $\mathbf{A}$ is the adjacency matrix (could be weighted) for a strongly-connected directed network (or any connected undirected network, then we call $\mathbf{A}$ irreducible). If $\mathbf{A}$ is a real $n \times n$ matrix that either

a) has all strictly positive entries or

b) is irreducible

then

i) The leading eigenvalue κ_1 with $\kappa_1 \geq |\lambda|$ is strictly positive ($\kappa_1 > 0$) and has one-dimensional eigenspace $\{v : \mathbf{A}v = \kappa_1 v\}$

ii) κ_1 has a leading left and right eigenvectors associated with it that have all strictly positive entries.

iii) The only non-negative eigenvectors of $\mathbf{A}$ are multiples of the leading eigenvectors. The eigenvectors for all other eigenvalues of $\mathbf{A}$ have at least one negative entry.

§22.2 Review of Probability

Definition 22.10 (Joint Probability) — The probability of events A and B occurring is denoted $P(A \cap B)$ or $P(A, B)$.

Definition 22.11 (Conditional Probability) — The probability of $\mathbf{A}$ given that $\mathbf{B}$ occurred is denoted $P(A|B) = \frac{P(A \cap B)}{P(B)}$.

Theorem 22.12

The probability of A or B (or both) occurring is

$$P(A \cup B) = P(A) + P(B) - P(A \cap B)$$

Law of Total Probability

Let X denote the sample space of all possible events. Let $\{B_i\}_{i=1}^{\infty}$ be a countable (it could be finite) partition of X, so $X = \bigcup_{i=1}^{\infty} B_i$, then

$$P(A) = \sum_{i=1}^{\infty} P(A \cap B_i) = \sum_{i=1}^{\infty} P(A|B_i)P(B_i)$$

Bayes' Rule:

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

Definition 22.13 (Independent Events) — Events A and B are independent if an only if one of the following equivalent statements hold

i) $P(A \cap B) = P(A)P(B)$

ii) $P(A|B) = P(A)$

iii) $P(B|A) = P(B)$

Definition 22.14 (Expectation) — We have two cases

- Discrete: Let X be a discrete random variable with possible events $\{x_i\}_{i=1}^{\infty}$ and probability mass function (PMF) p_x. The expected value of $g(X)$ is

$$E[g(X)] = \langle g(X) \rangle = \sum_{i=1}^{\infty} g(x_i)p_x(x_i)$$

- Continuous: Let X be a continuous random variable with probability density function (PDF) f_X. Then

$$E[g(X)] = \int_{-\infty}^{\infty} g(x)f_x(x)dx$$

Note: Expectation is linear, i.e.,

$$E[X + Y] = E[X] + E[Y]$$
$$E[\alpha X] = \alpha E[X]$$

Definition 22.15 — A random variable X has

1. Mean $\mu_x = E[X]$
2. Variance $\text{var}(X) = \sigma_x^2 = E\left[(X - E[X])^2\right] = E\left[X^2\right] - E[X]^2$
3. n^{th} moment (about zero) $E[X^n]$

§22.3 Some Useful Inequalities

There are some useful inequalities to keep in mind

1. Cauchy-Schwartz for expectation
$$|E[XY]| \leq \sqrt{E[X^2]\,E[Y^2]}$$
2. Markov's Inequality: If X is a non-negative random variable and $a > 0$, then
$$P(X \geq a) \leq \frac{E[X]}{a}$$
3. Chebyshev's Inequality: Let X be a random variable with mean μ and variance σ^2. Then for $a > 0$,
$$P(|X - \mu| \geq a) \leq \frac{\sigma^2}{a^2}$$
4. Jensens's Inequality: If f is a convex function, then
$$E[f(X)] \geq f(E[X])$$
If f is a concave function, then the reverse inequality holds.

Note: $f : \Omega \to \mathbb{R}$ is convex if for all $0 \leq t \leq 1$ and $x_1, x_2 \in \Omega$,
$$f(tx_1 + (1-t)x_2) \leq tf(x_1) + (1-t)f(x_2)$$
f is concave if $-f$ is convex.

§23 | Dis 3: Apr 12, 2022

§23.1 Chebyshev's Inequality

Chebyshev's Inequality gives us:

Let X be a random variable with mean $\mu = \mathbb{E}[X]$ and variance $\sigma^2 = \text{var}(X)$. Then for $a > 0$

$$P(|X - \mu| \geq a) \leq \frac{\sigma^2}{a^2}$$

and

$$P(|X - \mu| > a) < \frac{\sigma^2}{a^2}$$

This is an upper bound $\frac{\sigma^2}{a^2}$ on the probability that a realization of our random variable X is far, more than a, away from the mean. If $\frac{\sigma^2}{a^2} \geq 1$, then we don't gain any info since $P(|X - \mu| \geq a) \leq 1$ is always true for a probability.

If $x < f(n)$ and $\lim_{n\to\infty} f(n) = \infty$, that does not mean $x \to \infty$ since we have an inequality, i.e., $x < \infty \neq x \to \infty$.

Suppose $P(|X - \mu| > \varepsilon) < \frac{f(n)}{\varepsilon}$ where $\varepsilon > 0$ is arbitrary and $\lim_{n\to\infty} f(n) = 0$. Then as $n \to \infty$, $P(|X - \mu| > \varepsilon) \to 0$ and the probability X is away from the mean tends to 0. It is still possible that some realization x of X has $|x - \mu| > \varepsilon$ but this probability approaches 0.

§23.2 Random Graphs vs Realizations

1. In a random graph ensemble, for example $G(n, p)$ for the ER random graph model we consider a probability distribution of all graphs that can be drawn from the model. In class, we have been talking about quantities in expectation (over the ensemble). For example, the expected mean degree is

$$c = \sum_{G \in G(n,p)} \begin{pmatrix} \text{mean degree} \\ \text{of } G \end{pmatrix} \begin{pmatrix} \text{probability of} \\ \text{getting the graph } G \end{pmatrix} = p(n-1)$$

2. On the other hand, if we have a realization or instance of an ER network, then we have already selected some particular graph from the random graph ensemble, $G \in G(N, p)$. G can have any number of edges between 0 and $\binom{N}{2}$. It could even be

   - Graph of N isolated nodes (prob $(1-p)^{\binom{N}{2}}$)
   - N-clique where each nodes is adjacent to all other nodes (prob $p^{\binom{N}{2}}$)

   When we calculate quantities for a realization $G \in G(N, p)$ then they need not equal the expected value.

§23.3 Clustering Coefficient

1. The global clustering coefficient is defined on page 184 of the textbook as

$$C = \frac{\#\text{ of triangles} \times 3}{\#\text{ of connected triples}}$$

   We have $C \in [0, 1]$ where $C = 0$ if there are no triangles in the network (for example trees) and $C = 1$ if every component of the network is a clique.

2. The local clustering coefficient for node i in a network given by on page 186 of the textbook as

$$C_i = \frac{\#\text{ of pairs of neighbors of i that are connected}}{\#\text{ of pairs of neighbors of i}}$$
$$= \text{probability that if we select a pair of neighbors of i, then they will be connected}$$

3. The Watts-Strogatz clustering coefficient (a.k.a a network-average clustering coefficient) is the mean local clustering coefficient. It is given by

$$C_{\text{WS}} = \frac{1}{n}\sum_{i=1}^{n} C_i$$

The global clustering coefficient and the WS clustering coefficient are not equal. Both are popular to use.

§23.4 Trees

Definition 23.1 — A tree is a connected undirected graph with no cycles. The directed case is usually called a directed acyclic graph (DAG).

Definition 23.2 (Leaf) — A leaf is a node in a tree with degree 0 or 1.

Theorem 23.3
Every tree has at least one leaf.

Proof. For $n = 1$ and $n = 2$ all nodes are leaves.
For $n \geq 3$, we do a proof by contradiction. Suppose on the contrary that there are no leaves. Then, every node has degree at least 2. We claim we can find a walk as follows

1. Start at some node v_1. Follow one of its edges (it has at least 2 incident edges) to another node v_2
2. For node v_i with $i \geq 2$, we pick node v_{i+1} following an edge different from the edge (v_{i-1}, v_i). We can always find such an edge because $\deg(v_i) \geq 2$.

We can made this walk arbitrary long by allowing repeated edges in the walk as long as we don't take same edge twice in a row. Because the walk length is arbitrary, we can make it longer than $n+1$ nodes and therefore there must be some node v^* out of the n nodes repeated on the walk (pigeonhole principle). Take the part of the walk that goes and we have found a cycle. This contradicts that the graph is a tree. □

Theorem 23.4
Trees are planar graphs.

Proof. We will show this by induction on the number of nodes n.

- Base case: For $n = 1$, the only tree is $\cdot$ which is planar.

- Inductive step: Suppose that all trees with n nodes are planar. Consider an arbitrary tree G with $n+1$ nodes. This tree has at least one leaf node v^*

Consider the tree $G^{'}$ which has v^* and its single incident edge removed. G' has n nodes and by inductive hypothesis it is planar. To go from G' to G we just need to add back our leaf node and its edge. IT is always possible to do this in a planar way. So G is planar. We showed that all trees with $n = 1$ are planar, and if all trees with n nodes are planar, then so are all trees with $n+1$ nodes. By induction, all trees are planar. □

Theorem 23.5

A tree with n nodes has $n-1$ edges.

Proof. Homework 2 # 2a. □

§23.5 Summary of ER Random Graph Model

Consider $G(N,p)$ denoted the ER model for graphs with N nodes and independent edge probability p.

- Probability of selecting realization $G \in G(N,p)$ if G has M edges is: $p^M(1-p)^{\binom{N}{2}-M}$.
- Probability of drawing any graph with M edges is

$$P(M) = \binom{\binom{N}{2}}{M} p^M (1-p)^{\binom{N}{2}-M}$$

In expectation, we have

- Expected degree distribution

$$p_k = \binom{N-1}{k} p^k (1-p)^{N-1-k} \approx \text{Poisson}(c) \text{ as } n \to \infty$$

- Expected mean degree $c = p(n-1)$
- Expected number of edges $\mathbb{E}[m] = \binom{n}{2} p = \frac{1}{2}nc$
- As $n \to \infty$, the global clustering coefficient approaches p.
- For each node, the expected local clustering coefficient is p.
- The expected WS clustering coefficient is p.
- Giant connected components (GCC).

Definition 23.6 — A GCC is a connected component that grows in size proportional to the size of the network N, i.e., $\mathcal{O}(N)$.

For an ER random graph model, recall that $c = (N-1)p$ is the mean degree

- If $c < 1$, we expect no GCC
- If $c > 1$, we expect GCC wit size S fraction of nodes in GCC satisfies

$$S = 1 - e^{-cS}$$

Note: $S = 0$ always is a solution and we want $S > 0$ solution.

§24 | Dis 5: Apr 25, 2022

§24.1 Rand Index

Question 24.1. How can we quantify if a community detection algorithm is doing well with finding known communities?

The rand index calculates what fraction of pairs of nodes are labeled correctly relative to each other. Suppose we have two vectors of n labels g and t. t is the true labels and g is the labels we found that we want to compare.

$$\delta_{ij} = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{otherwise} \end{cases}$$

$$s = \sum_{j=1}^{n} \sum_{i<j} \delta_{g_i g_j} \delta_{t_i t_j} = \#\text{ pairs of distinct nodes in the same groups}$$

$$d = \sum_{j=1}^{n} \sum_{i<j} (1 - \delta_{g_i g_j})(1 - \delta_{t_i t_j}) = \#\text{ pairs of distinct nodes in different groups}$$

The rand index of g and t is

$$R(g,t) = \frac{s+d}{\binom{n}{2}} = \frac{\#\text{ of pair of nodes correctly labeled relative to each other}}{\text{total } \#\text{ pairs of nodes}}$$

We always have $0 \leq R \leq 1$. Higher R means the groups we found better match the groups of pre-existing labels.

Example 24.1

Consider

$$g = \begin{bmatrix} 1 & 1 & 0 & 0 \end{bmatrix}$$
$$t = \begin{bmatrix} 0 & 0 & 1 & 1 \end{bmatrix}$$

Then, we can deduce that $s = 2$, $d = 4$, and $R = 1$. Notice that

- both g and t have 2 of the same pairs of nodes in the same group
- both g and t have 4 of the same pairs of nodes in different groups

§24.2 Centrality Measures

- For an undirected network, $\vec{k} = \mathbf{A}\vec{1}$ where $\mathbf{A}$ is the adjacency matrix and entry k_i = degree of node i.
- For a directed network

$$\text{in-degree: } \vec{k}^{\text{in}} = \mathbf{A}\vec{1} \quad \text{row sums: } k_i^{\text{in}} = \sum_{j=1}^{n} A_{ij}$$

$$\text{out-degree: } \vec{k}^{\text{out}} = \mathbf{A}^\top \vec{1} \quad \text{col sums: } k_j^{\text{out}} = \sum_{i=1}^{n} A_{ij}$$

- The degree is how many incident edges we have. The higher the degree, the more neighbors/connections we have.

§24.3 Eigenvector Centrality

This is defined for a strongly-connected directed network or an undirected network for. For these networks, Perron-Frobenius theorem gives us there is a largest positive eigenvalue k with corresponding eigenvectors $\vec{x}$ with all non-negative entries for our adjacency matrix $\mathbf{A}$. Let $\vec{x}$ be the eigenvector centrality vector where $x_i =$ eigenvector centrality of node i. Note that $\vec{x}$ satisfies

$$\mathbf{A}\vec{x} = k\vec{x}$$

Unlike for the degree, the entries of $\vec{x}$ depend on each other with $x_i = \frac{1}{k}\sum_{j=1}^{n} A_{ij}x_j$.
Interpretation: You are more "important" if other "important" nodes point at you (in-edge indirected network or any edge in an undirected network). We can think of influences as flowing towards high EV centrality nodes. A node has higher EV centrality if its neighbors also have high EV centrality. You are "important" in EV centrality if you have lots of connections that also have lots of connections.
One downside of EV centrality is that in a directed network, nodes with no in-edges have EV centrality of 0. That means they also don't give any EV to their neighbors. Also it breaks the strongly connected assumptions we used to apply Perron-Frobenius theorem and get non-negative EV centralities that make sense. Katz centrality addresses this issue.

§24.4 Katz Centrality

We no longer need strongly connected and can calculate Katz centrality for any weakly connected/directed or connected/undirected network with some largest positive eigenvalue k. Let $0 < \alpha < \frac{1}{k}$. We can pick this parameter to calculate the Katz centrality. Every node gets "for free" a base centrality $\beta > 0$ even if it has no in-degree. Usually we can take $\beta = 1$. The Katz centrality of node i is

$$x_i = \alpha \sum_{j=1}^{N} A_{ij}x_j + \beta$$

If $\beta = 1$, in matrix/vector form

$$\vec{x} = (\mathbf{I} - \alpha\mathbf{A})^{-1}\vec{1}$$

- As $\alpha \to 0^+$, $\vec{x}$ ranks the nodes like degree centrality.
- As $\alpha \to \frac{1}{k}^-$, $\vec{x}$ ranks the nodes like eigenvector centrality.

§25 | Dis 6: May 3, 2022

§25.1 Closeness Centrality

Let's first denote $d(i,j)$ = geodesic distance (shortest path length) from node i to node j. Let there be n nodes in the network. The mean geodesic distance from node i to another node is

$$l_i = \frac{1}{n}\sum_{j=1}^{n} d(i,j)$$

The closeness centrality of node i is

$$x_i = \frac{1}{l_i} = \frac{n}{\sum_{j=1}^{n} d(i,j)}$$

Sometimes closeness centrality is calculated without including the distance from i to itself $d(i,i) = 0$, so

$$l_i' = \frac{1}{n-1}\sum_{j\neq i} d(i,j)$$
$$x_i' = \frac{n-1}{\sum_{j\neq i} d(i,j)} = \frac{n-1}{n}x_i$$

Interpretation: Nodes with high closeness centrality can on average reach other nodes faster. They might be effective at spreading news, advertising, etc. in a social network. They might be good transportation hubs or locations for distribution centers in a supply-chain network.

§25.2 Betweenness Centrality

The definition of betweenness centrality of node i in the textbook is

$$x_i = \sum_{s=1}^{n}\sum_{t=1}^{n} \frac{n_{\text{st}}^i}{g_{\text{st}}}$$

where

$$n = \#\text{ nodes in the network}$$
$$g_{\text{st}} = \#\text{ of shortest paths from s to t}$$
$$n_{\text{st}}^i = \#\text{ of shortest paths from s to t that include } i$$

Nodes with high betweenness centrality tend to act as "bridges" between groups in the network

- Removing them would be disruptive to traffic in a road network or the ability to spread information in a social network
- In a professional networking context, people with high betweenness are who we want to meet since they have access to connections we might not have.

A few more points on centrality

- For closeness, betweenness and degree centrality, you can calculate this for each node one at a time.
- For eigenvector, Katz and PageRank centrality, the centrality of one node depends on the centrality of other nodes and we calculate them all at once using matrix equations.

- Edge weights and centrality measures
  - many Networkx centrality functions have a parameter "weight" that we can give the name of an edge attribute
  - Betweenness and closeness centrality treat edge weights as distances, os higher edge weight means nodes are farther apart/less similar
  - Degree, eigenvector, Katz and PageRank centrality treat edge weights as some sort of connection strength, so higher edge weight implies nodes have stronger ties/are more similar
  - A heuristic way to get a distance or similarity from the other is

$$\text{similarity} = 1 - \frac{\text{distance}}{\text{max distance}} \in [0,1]$$
$$\text{distance} = 1 - \frac{\text{similarity}}{\text{max similarity}} \in [0,1]$$

§25.3 PageRank

The PageRank centrality of a node is its probability in the stationary distribution of a random walk with teleportation. Let

- $\vec{v} \in \mathbb{R}^n_+$ satisfying $\sum_{i=1}^n v_i = 1$ be the teleportation vector, it is common to take $\vec{v} = \frac{1}{n}\vec{1}$
- $\alpha \in [0,1]$ be the teleportation rate

The random walk with teleportation has transition probabilities

$$\begin{aligned} P_{ij} &= P\left(X_{t+1} = i | X_t = j\right) \\ &= \underbrace{(1-\alpha)\frac{A_{ij}}{k_j^{\text{out}}}}_{\text{prob of randomly walking to node i}} + \underbrace{\alpha v_i}_{\text{prob of teleporting to node i}} \end{aligned}$$

Note that if $\mathbf{D}$ is a diagonal matrix

- $\mathbf{AD}$ has column i of $\mathbf{A}$ multiplied by D_{ii}
- $\mathbf{DA}$ has row i of $\mathbf{A}$ multiplied by D_{ii}
- If $D_{ii} \neq 0$ for each i, then $\mathbf{D}^{-1}$ is a diagonal matrix with $\left[\mathbf{D}^{-1}\right]_{ii} = \frac{1}{D_{ii}}$

To get the transition matrix $\mathbf{P}$

- $\frac{A_{ij}}{k_j^{\text{out}}}$, i.e., we want to multiply column j of $\mathbf{A}$ by $\frac{1}{k_j^{\text{out}}}$, so we want $\mathbf{A}\left(\mathbf{K}^{\text{out}}\right)^{-1}$
- We want all entries in row i of $\mathbf{P}$ to have αv_i in it. Let $v = \vec{v}\vec{1}^\top$

We have

$$\mathbf{P} = (1-\alpha)\mathbf{A}\left(\mathbf{K}^{\text{out}}\right)^{-1} + \alpha\vec{v}\vec{1}^\top$$

The PageRank centrality of node i is π_i where $\vec{\pi}$ is the stationary distribution vector of $\mathbf{P}$ satisfying $\vec{\pi} = \mathbf{P}\vec{\pi}$, which is

$$\vec{\pi} = \left[(1-\alpha)\mathbf{A}\left(\mathbf{K}^{\text{out}}\right)^{-1} + \alpha\vec{v}\vec{1}^\top\right]\vec{\pi}$$

Nodes with high PageRank centrality have higher stationary distribution of the random walker being there. Google's original PageRank algorithm found important webpages people are likely to find relevant.

Newman's Version of PageRank

$$\vec{x} = \alpha'\mathbf{A}(\mathbf{K}^{\text{out}})^{-1}\vec{x} + \beta\vec{1}$$

If in our more general PageRank, we take $\vec{v} = \frac{1}{n}\vec{1}$

$$\vec{\pi} = \left[(1-\alpha)\mathbf{A}(\mathbf{K}^{\text{out}})^{-1} + \alpha\vec{v}\vec{1}^{\top}\right]\vec{\pi}$$
$$\vec{\pi} = (1-\alpha)\mathbf{A}(\mathbf{K}^{\text{out}})^{-1}\vec{\pi} + \alpha\frac{1}{n}\vec{1}\vec{1}^{\top}\vec{\pi}$$

Now let

$$\vec{x} = \vec{\pi}$$
$$\alpha' = 1-\alpha$$
$$\beta = \alpha\frac{1}{n} = (1-\alpha')\frac{1}{n}$$

Substituting we get

$$\vec{x} = \alpha'\mathbf{A}(\mathbf{K}^{\text{out}})^{-1}\vec{x} + \beta\vec{1}$$

So Newman's equation is PageRank with teleportation vector $\vec{v} = \frac{1}{n}\vec{1}$.