CS 506 → Network Analysis (3/24)

→ Feel free to reach out for extensions — good to identify when you need more time

→ Last classes: Logistic/Linear Regression

→ Network Analysis: tools we may encounter, very present in daily live!

- Example Networks — social networks, epigenetic networks, networks between brain regions, internet wifi (obviously very DIVERSE)

— want to formularize the network to ask specific Q's

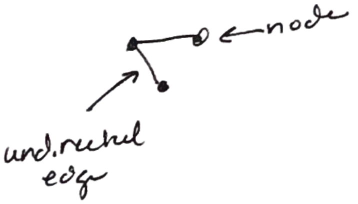
⇒ Internet:
- what is the internet traffic @ a location?
- Anomalous traffic patterns?
- model of the internet?

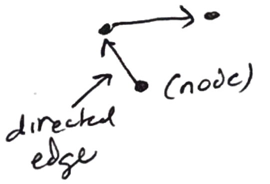
⇒ Biology:
- which regions of brain communicate for a specific task
- patterns of interactions between genes

⇒ Social:
- who is friends with whom? Who are influencers?
- what social groups are present? - Is there a clear representation of this?

Graphs (Also see: Lecture 4) → 2/8

- use graph/Graph Theory to model/represent and analyze Network



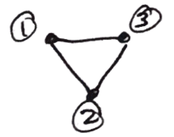


} express directionality "I follow you but you might not follow me"

Formally, graph G is ordered pair of sets $(V, \mathcal{E})$ → set of all edges; set of all nodes

- Let $G = (V, \mathcal{E})$

$V = \{1,2,3\}$ $\mathcal{E} = \{(1,2), (1,3), (2,3)\}$ ← undirected

G → 

They're all connected to each other! ⇒ triangle

symmetric matrix

computational perspective

- best way to store this info, so certain queries are more efficient?

ex. adjacency matrix

$$\begin{array}{c|ccc} & ① & ② & ③ \\ \hline ① & 0 & 1 & 1 \\ ② & 1 & 0 & 1 \\ ③ & 1 & 1 & 0 \end{array}$$

- entry represents whether there's an edge between nodes

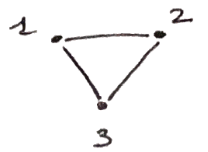
- could be non-symm. matrix when directed

- can also assign weights to matrix (indicates strength of connection)

note: typically no edge between node & itself (zero diagonal)

Adjacency matrix (further illustrated)

$$\text{nodes}\ \begin{array}{c|ccc} & \multicolumn{3}{c}{\text{nodes}} \\ & 1 & 2 & 3 \\ \hline 1 & 0 & 1 & 1 \\ 2 & 1 & 0 & 1 \\ 3 & 1 & 1 & 0 \end{array}$$



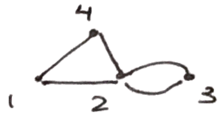
- walk through columns and rows and draw it out
- note: maximum possible # of edges
  ↳ when there are only 3 nodes, that means it's a triangle

Adjacency List → lists all the nodes something is connected to

1: {2, 3}
2: {1, 3}
3: {1, 2}

] more space efficient (could use a dict) BUT makes certain processes harder to solve

→ See Slides: Question from Goodwill Hunting



- [illegible]

"complex" (nah)

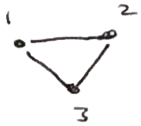
Graphs Characteristics

degree of a node → # edges connected to it
↳ average # connections on network? minimum? maximum?
↳ ~~[illegible]~~ want to make sure min degree is low esp. for machines w/ sensitive workload

path between two nodes → sequence of edges that join them

complete when there is an edge between ~~[illegible]~~ every pair of Nodes

Questions] ① sum of degrees of all Nodes in graph as a function of N_E (# edges) → see triangle $= 2 * N_E$



each node degrees = 2

answer = 6

- 2 endpoints to each edge
- so count the edges twice

note: must be undirected

② # edges in a complete graph as function of the # nodes (N_V)

(everyone meets everyone ... how many handshakes?) $= \frac{N_V(N_V-1)}{2}$

Graph Problems] ☹ ① Clique Problem (largest complete subgraph)

☹ ② coloring problem (no two vertex that share an edge share a color) → coloring a graph w/ certain # colors

☹ ③ Traveling salesman (most efficient route through all cities & return home)

☺ ④ shortest path (given edges have weights, path w/ shortest sum of weights)
↳ actually really easy
- Dijkstra algorithm (CS330)

☹ ⑤ vertex cover (min # of nodes to remove + get rid of all the edges)

Network characteristics

- Distribution of edges / node degrees:
  - anomaly detection — describe flow through the Network
  - ranking / recommendation

- Centrality of a node:
  - Identify influencers (internet → server that projects a lot of info)
  - Discover groups / clusterings — How nodes affect connectivity / flow

↳ find network metric ~~to~~ to characterize things in a mathematical way

Network Analysis = find a model of the network

- adding / removing nodes & edges = generation
- The state @ given point is the stochastic result of these processes
- model characteristics from prev. ↑ by modeling state of Graph
  ↳ can we model its creation?
  ↳ find distr. of nodes @ any given point

Random Graph Model

1) N nodes
M edges

$G(N, M) = \{ G = (V, E) \mid |V| = N, |E| = M \}$ → distribute M edges uniformly

↳ all graphs w/ that # N and E

↳ pick a graph randomly from set of graphs

(Ex) $G(3,2) = \{$  $\}$ → pick a graph w/ prob. 1/3

In general, probability w/ which you pick graph from $G(N, M)$? → choose N nodes from all poss. combo

aka $\frac{1}{c\left(\frac{n(n-1)}{2}, m\right)}$ ← $p = \binom{\binom{N}{2}}{M}^{-1}$ → # max possible edges? → how many ways are there to choose M from all possible N

inverse of N "choose" 2

2) Let $G(N, p)$ be generated ... randomly connect nodes w/ prob. p, independently → N "choose" 2

Note: $\frac{N(N-1)}{2}$ Bernouilli trials inserting edges w/ prob. p

What is $G(N, p)$ as a function of M?

↙ generate graphs randomly

$$f_{G(N,M)} = p^{M}(1-p)^{\binom{N}{2}-M}$$

↳ both methods related...

$G(N,p)$ given M edges $= G(N,M)$

- see slides for PROOF — what's the

$$P(G(N,p) \,\|\, |\mathcal{E}_{G(N,p)}| = M)$$

- what's the distribution of degree of Nodes?

$$P(\deg(v) = k) = \binom{N-1}{k} p^{k} (1-p)^{N-1-k}$$ ← Binomial distribution

As $N \to \infty$ (Np constant), converge to Poisson Dist.

Q] - Expected # of connections / degree for a node in this Graph?

~Np — does this make sense?] think: social networks → what does Bin ~~distr~~ dist look like?

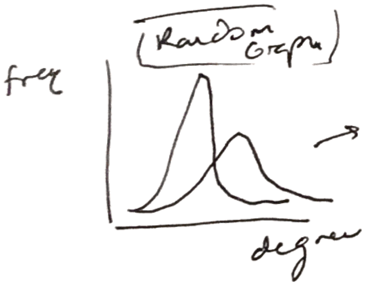
↳ means on avg. everyone has similar # connections

- NOT so accurate re: social nets

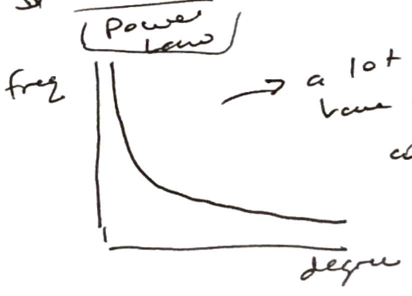
↑ N = # nodes

Social Networks = power law distribution $P(k) = Ck^{-\alpha}$ → tune constants C and α

↑ degree of node



→ note: BINOMIAL distributions



→ a lot of people have very little connections but there are people w/ high degree (just lower freq)

- Always think — what are characteristics graph based on model? Does it accurately describe the situation?

- After that, compute things on the graph/model to get intuition about the network

Describing / Comparing Graphs

- define metrics that represent characteristics & ~~describe~~ compare these

- metrics → describe whole graphs
  → describe individual nodes

① **Diameter** = ~~d_{ij}~~, maximum shortest path between nodes

d_{ij} = shortest path between i and j $\text{Diam}(G) = \max_{ij} d_{ij}$

"Small world phenomenon" (for example)
↳ only takes few jumps to know someone famous

Q]  max diameter = 2

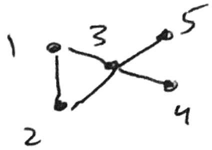
② **Clustering coefficient**

$C = \# \text{triangles} / \# \text{triplets}$

→ closed triplet

→ 3 nodes connected by 2 edges

defined as being centered on a node



$$C = \frac{\overset{(1)}{1} + \overset{(2)}{1} + \overset{(3)}{1} + \overset{(4)}{0} + \overset{(5)}{0}}{1 + 1 + 6 + 0 + 0} = \frac{3}{8}$$ → captures # times you don't close the triangle

- doesn't matter path, just what node you start on
- start @ each node, count # triangles, add ⇒ numerator
- start @ each node, count # triplets, add ⇒ denominator

③ **Density** N = # nodes ; M = # edges

$$\text{density} = \frac{2M}{N(N-1)}$$ → max possible edges

Q] density of complete graph? 1

$2M = N(N-1)$

metrics on Nodes....

Degree centrality → more central a Node is, higher # connections

$$C_{deg}(v) = Deg(v) \leftarrow \text{the degree}$$

closeness centrality → more central = closer to all other nodes

$$C_{close}(v) = \frac{1}{\sum_u d(\underset{\substack{\uparrow \\ \text{node \#1}}}{u}, \underset{\substack{\uparrow \\ \text{node \#2}}}{v})}$$

→ Look @ slides & do exercises

Harmonic Centrality →

- sum of inverse distances

$$C_h(v) = \sum_{u \neq v} \frac{1}{d(u,v)}$$

→ need to set $d^{-1}(u,v) = 0$ if no path between u and v

Betweenness Centrality

→ # times a node acts as a bridge along shortest path to other nodes

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

σ_{st} → total # shortest path between s and t

$\sigma_{st}(V)$ → # those shortest paths that go through v

(see ~~B~~ code demo)

↳ networkx library in python