

How to analyse networks?

- can be done at multiple levels
 - node level
 - graph of nodes
 - statistical properties

① Node level

- assign importance to each node
(microscopic level)

assign distribution to nodes

② Graph of nodes

- Similarity b/w nodes
- Analyze groups
- Communities

(mesoscopic scale/level)

③ Statistical Properties

- p_k (degree distribution)
- $\langle \rangle$ module models (Process on networks)
(Null hypothesis)

2 objects → ① Degree

② Shortest Path

Node level description

- Importance \rightarrow content
- Influential

node with most degree \rightarrow important (not influential)

Influential can be \rightarrow

- knowing many (acquaintance)
- knowing many important
- message passing / propagation

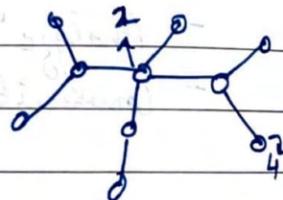
Graph theory has the concept of eccentricity.

Eccentricity $e(v) \rightarrow$ maximum of the shortest paths from a vertex to any other vertex in the network.

Radius $\rightarrow \min E(v)$

Diameter $\rightarrow \max E(v)$

Center / Central point



Eccentricity is same as the radius

Periphery \rightarrow eccentricity is same as the diameter



Centrality Measures

- lets you calculate how central / important a vertex is

① Degree Centrality

$$C_D(i) = k_i \quad (\text{degree of node } i \text{ is } k_i)$$

$$\frac{C_D(i)}{n-1} = C_D^*(i) \rightarrow \text{normalized degree centrality}$$

($n-1$ is the maximum degree a node can have where n is the no. of nodes) \hookrightarrow hubs

② Closeness Centrality

- has to do with avg. shortest path
- How close a vertex is to every other vertex in the network.

$d_{ij} \rightarrow$ length of the shortest path b/w vertex i & j
 \hookrightarrow distance matrix

$$\frac{1}{n-1} \sum_{j \neq i} d_{ij} = l_i \quad (\text{average shortest path length for node } i)$$

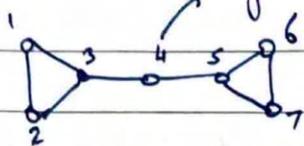
avg. value because every node has $n-1$ distances to $n-1$ nodes in a network.

$$C_C(i) = \frac{1}{l_i} = \frac{n-1}{\sum_{j \neq i} d_{ij}}$$

\downarrow
closeness
centrality

There is a positive correlation b/w degree and closeness.

highest closeness centrality



3 & 5 have highest degree centrality.

If we remove vertex 3, 4 or 5, we divide the graph in 2 parts.

Same does not apply to vertex 1, 2, 6 or 7.

Which brings us to Betweenness Centrality.

The closeness centrality for a network with large no. of nodes lies b/w $\frac{1}{n}$ to $\ln n$ (base e) (usually).

$$l_i = 1$$

$$l_i = \ln n$$

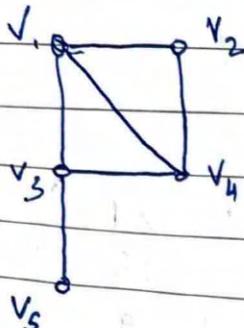
If there are multiple components in a network,
 $d_{ij} = \infty$.

$$C_c(i) = \sum_{j \neq i} \frac{1}{d_{ij}} \quad (\text{removing } \infty \text{ distance})$$

- Betweenness



(3) Eigenvector Centrality →



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

Adjacency Matrix

we assign some scores to the vertices

Scores → $\begin{pmatrix} n_1 \\ n_2 \\ n_3 \\ n_4 \\ n_5 \end{pmatrix} = X$

each row of AX will be the new score of a vertex, which should be equal to the initial score.

$$\begin{pmatrix} n_1 \\ n_2 \\ n_3 \\ n_4 \\ n_5 \end{pmatrix} = A \begin{pmatrix} n_1 \\ n_2 \\ n_3 \\ n_4 \\ n_5 \end{pmatrix}$$

$$\bar{X} = A\bar{X}$$

we want that the ranking of scores of vertices based on scores does not change b/w \bar{X} & X .

let initial scores \rightarrow

$$\begin{pmatrix} t_1 \\ 3 \\ 2 \\ 3 \\ 3 \\ 1 \end{pmatrix} \rightarrow \text{degree matrix}$$

$$t_2 \rightarrow \begin{pmatrix} 8 \\ 6 \\ 7 \\ 8 \\ 3 \end{pmatrix} \rightarrow t_3 \rightarrow \begin{pmatrix} 21 \\ 16 \\ 19 \\ 21 \\ 7 \end{pmatrix}$$

$$\bar{x}(t+1) = A^t n(t) = A^t n(0)$$

$A^t A^t n(t-1) \& \text{ so on}$

$$n(0) = \sum_i c_i v_i$$

$c_i \rightarrow \text{eigen value}$
 $v_i \rightarrow \text{eigen vector}$

all eigen vectors are going to be orthogonal to each other.

$A^t \rightarrow \text{symmetric matrix}$

$$\bar{x}(t+1) = \sum_i A^t c_i v_i$$

$$= \sum_i c_i \lambda_i^t v_i$$

$$Av_i = \lambda v_i$$

$$A(Av_i) = \lambda^2 v_i$$

$$A^t v_i = \lambda^t v_i$$

$\lambda_i > 0$ for real symmetric matrices

DOMS

Page No.

Date

/ /

eigen values of an adjacency matrix are distinct

let's assume $\lambda_1 > \lambda_{i+1}$

$$\bar{n}(t+1) = \lambda_1^t \sum_i c_i \left(\frac{\lambda_i}{\lambda_1}\right)^t v_i$$

$$\frac{\lambda_i}{\lambda_1} < 1 \text{ for } i \neq 1.$$

$\therefore \bar{n}(t+1)$ is stable for all λ_i except λ_1 .

largest eigen value is the principle eigen value.

We are only looking at the eigen vector corresponding to the largest eigen value.

when ~~$\frac{\lambda_i}{\lambda_1} = 1$~~ we have

$$\bar{n}(t+1) = \lambda_1^t \sum_i c_i v_i$$

$\bar{n}(t+1) \sim c_1 \lambda_1^t v_i \rightarrow$ eventually $\bar{n}(t+1)$

because only $c_1 v_i$ is significant after long time.

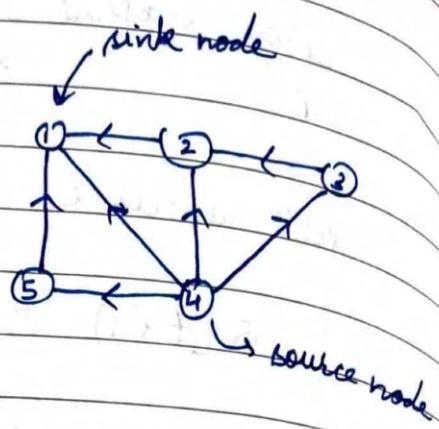
$$c_1 \lambda_1^t \rightarrow \text{constant}$$

\therefore only v_i matters.

(4)

Directed graph

- source nodes
- sink nodes
- Cycles



direction → ① In-degree
 ② Out-degree

which one should we think about?
 (depends on the problem at hand)

for a citation network → In-degree

(how many times an article has been cited)

for twitter network → Out-degree

(how much can a person spread data)

$$\frac{k_i^{\text{in}}}{k_i^{\text{out}}}$$

In-degree

(popularity)

∴ In & Out both degrees are significant

for web → In-degree

We will focus on In-degree.

$$A = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{pmatrix}$$

(incident edge is 1)

initial score based on in-degrees \rightarrow

$$\begin{pmatrix} t_0 \\ 3 \\ 2 \\ 1 \\ 0 \\ 1 \end{pmatrix}$$

$$t_1 = At_0$$

$$t_1 = \begin{pmatrix} 3 \\ 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

$$t_2 = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \rightarrow t_3 = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

vertex no. 4 propagates the 0 score to the rest of the graph.

$$n_i = \alpha \sum_j A_{ij} n_j + \beta$$

multiply by constant α

add constant β

Katz
Centrality

vertex 4 propagates 0 because it is 0. We can start it with a constant instead.

$\alpha A \rightarrow$ degree centrality
 α is always less than 1.

DOMS Date / /
Page No. / /

let $\beta = 1$

$$\bar{x} = \alpha A \bar{x} + 1$$

$$(I - \alpha A) \bar{x} = 1$$

$$\bar{x} = (I - \alpha A)^{-1} \cdot 1$$

$$= \left(I + \alpha A + \frac{\alpha^2 A^2}{2!} + \dots \right) \cdot 1$$

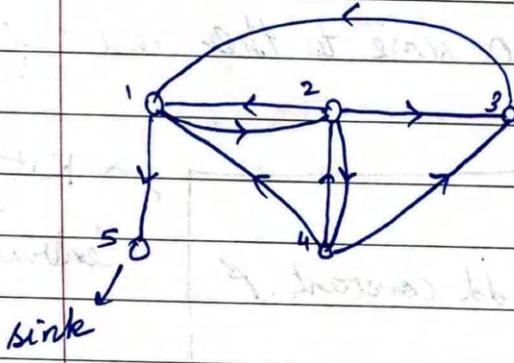
$$\text{if } \alpha = 0, \bar{x} = I \cdot 1$$

if we take just the first order term $\rightarrow \bar{x} = (I + \alpha A) \cdot 1$

this is a weighted sum. weight being α .
The further the neighbour, less the weight for
that neighbour.

$$\boxed{0 < \alpha < \frac{1}{\lambda_1}}$$

$\lambda_1 \rightarrow$ largest eigen value
for the adjacency matrix



$A = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \end{pmatrix}$

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

in-degree initial scores \rightarrow

$$\begin{pmatrix} 3 \\ 2 \\ 2 \\ 1 \\ 1 \end{pmatrix}$$

$t_1 \rightarrow \begin{pmatrix} 5 \\ 4 \\ 3 \\ 2 \\ 3 \end{pmatrix} \rightarrow t_2 \rightarrow \begin{pmatrix} 9 \\ 7 \\ 6 \\ 4 \\ 5 \end{pmatrix}$

node 5 becomes imp. just because of node 1.
This should not happen. We don't want 1 to transfer its entire importance to 5.

The importance transferred should be a fraction of importance which should be related to the out-degree.

Page Rank

$$n_i = \sum_j \frac{A_{ij}}{k_{j\text{out}}} \cdot n_j$$

$$\bar{n} = A D^{-1} \bar{n}$$

$D \rightarrow$ diagonal matrix to do with the out-degree

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

$$n_1 = \frac{n_2 + n_3 + n_4}{3}$$

$$n_2 = \frac{n_1 + n_4}{2}$$

$$n_3 = \frac{n_2 + n_4}{3}$$

$$n_4 = \frac{n_2}{3}$$

$$n_5 = \frac{n_1}{2}$$

$(D - A) D^{-1} \bar{n} = 0 D^{-1} \bar{n}$

Graph Laplacian

$$L D^{-1} \bar{n} = 0 D^{-1} \bar{n}$$

0 eigen value \downarrow eigen vector

$$\begin{pmatrix} n_1 \\ n_2 \\ n_3 \\ n_4 \\ n_5 \end{pmatrix} = \begin{pmatrix} 0 & 1/3 & 1 & 1/3 & 0 \\ 1/2 & 0 & 0 & 1/3 & 0 \\ 0 & 1/3 & 0 & 1/3 & 0 \\ 0 & 1/3 & 0 & 0 & 0 \\ 1/2 & 0 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} n_1 \\ n_2 \\ n_3 \\ n_4 \\ n_5 \end{pmatrix}$$

column sum
is 1

sink
node
column

Such columns are called column stochastic matrix

$$x = \pi x \quad \pi = A D^{-1}$$

transition matrix π should be a column stochastic matrix.

\therefore eigen value is 0, \therefore eigen vector is 1.

$$\therefore D^{-1}\bar{n} = 1$$

$$\bar{n} = D \cdot 1$$

$D^{-1}\bar{n}$ is column vector

\therefore importance is the out-degree for such a case calculation.

$$\bar{x} = \alpha \pi \bar{n} + \beta 1 \quad \rightarrow \text{completely general graph}$$

$$(I - \alpha \pi) \bar{x} = 1 \quad (\text{if } \beta = 1)$$

$$\bar{n} = (I - \alpha \pi)^{-1} 1$$

$$\left(0 < \alpha < \frac{1}{\lambda_1}\right)$$

where λ_1 is the largest eigen value of π .

- Big law of large nos

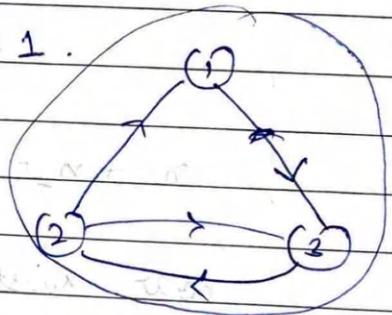
- stationary distribution

- avg. or standard deviation of various distribution

Combining Page rank & Katz Centrality to solve both source & sink problems, we get →

$$n_i = \alpha \sum_{j, k_j \text{ out}} \frac{A_{ij}}{k_j} n_j + \beta$$

$$\bar{n}(0) = \begin{bmatrix} 1/3 \\ 1/3 \\ 1/3 \end{bmatrix} \quad \xrightarrow{\text{assumed sum as 1.}}$$



by page rank

$$\bar{n}(t) = \sum_j \frac{A_{ij}}{k_j \text{ out}} n_j(t-1)$$

Markov chains

$$n_1 = \frac{n_2}{2} \quad \xrightarrow{\frac{n_3}{2}} \quad \xrightarrow{\text{probability of being at 1.}}$$

$$n_2 = \frac{n_3}{2}$$

$$n_3 = \frac{n_1}{2} + \frac{n_2}{2}$$

we are looking for stationary soln. so that it no longer depends on time.

$$\text{if } n(t) = f(n(t-1))$$

for stationary
 $n = f(n)$

\therefore for initial value, $\sum_i n_i(0) = 1$

this should be true not just for $t=0$, but for any point of time.

$$n_1 + n_2 + n_3 = 1 \text{ for all times}$$

now, substituting

$$\frac{n_2}{2} + \cancel{\frac{n_3}{2}} + \cancel{\frac{n_2}{2}} + \frac{n_1 + n_2}{2}$$

$$\frac{n_2}{2} + n_2 + n_2 = \frac{5n_2}{2} = 1$$

$$n_2 = \frac{2}{5} = 0.4$$

~~$n_2 = 0.4$~~

$$n_1 = 0.2$$

$$n_3 = 0.4$$

(E)
P_i: prob. for the walker to be at vertex labelled i.
at time t

Markov chains \rightarrow from a node, can visit all other nodes.

$$P_1^{(t)} = \cancel{P_2^{(t-1)}}$$

$$P_1^{(t)} = P_2^{(t-1)}/2 \rightarrow \text{from 2 can go to 1 or 3 ...}$$

only if it is at vertex 2 at time $t-1$.

$$P_2^{(t)} = \cancel{P_3^{(t-1)}}$$

$$P_3^{(t)} = P_2^{(t-1)}/2 + P_1^{(t-1)}$$

half prob. to go to 3 when at 2.

This is the basic idea of Google's page ranking algorithm.

$$R_i^{t+1} = \alpha \sum_j \frac{A_{ij} R_j^t}{k_j^{\text{out}}} + \frac{(1-\alpha)}{n}$$

\rightarrow probability of loop is $(1-\alpha)$

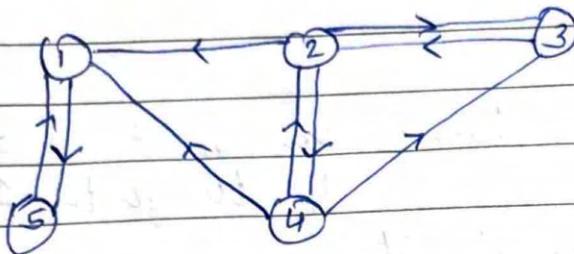
Brun & Page

$$\bar{R} = \alpha A D^{-1} R + \frac{(1-\alpha)}{n} \mathbf{1}$$

$$\bar{R} (I - \alpha A D^{-1}) = \frac{(1-\alpha)}{n} \mathbf{1}$$

$$\bar{R} = (I - \alpha A D^{-1})^{-1} \left(\frac{1-\alpha}{n} \right) \mathbf{1}$$

For Google page rank, $\alpha = 0.85$



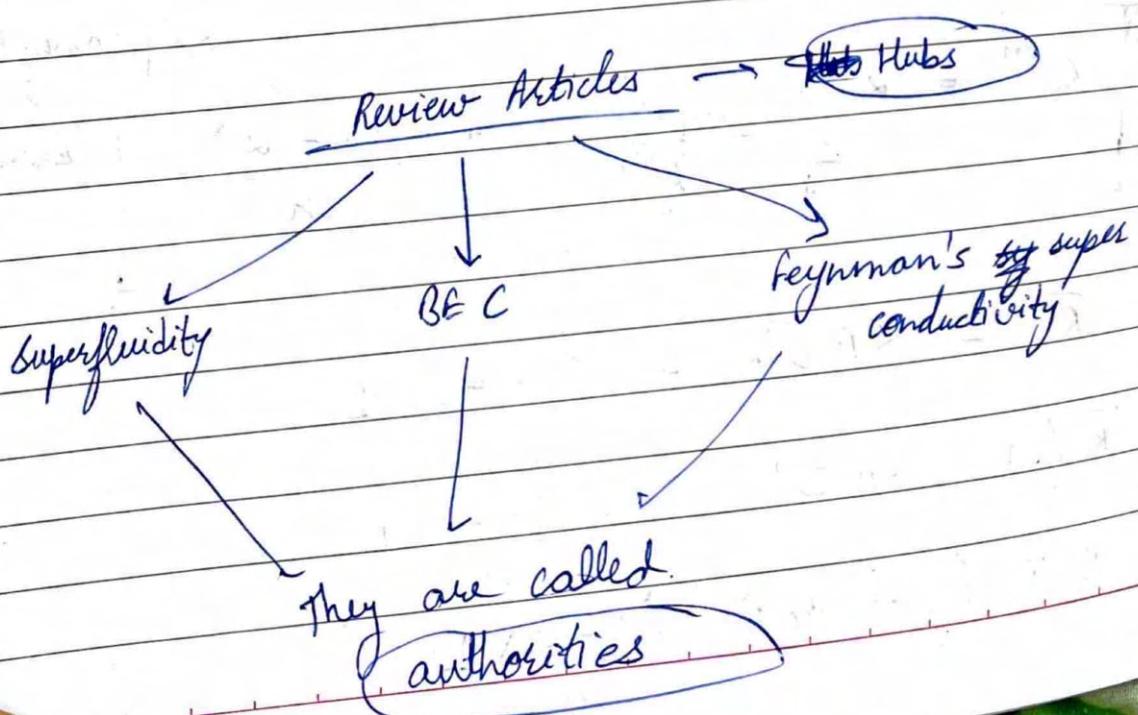
calculate R .

$$R_1 = \frac{1-\alpha}{5} + \alpha \left(\frac{R_2}{3} + \frac{R_4}{3} + \frac{R_5}{1} \right)$$

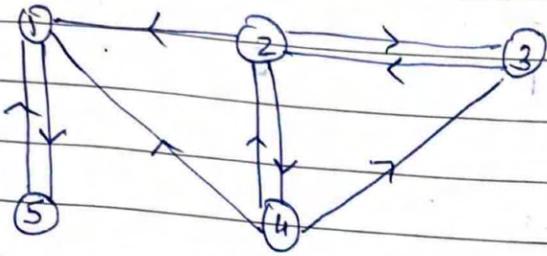
$$\alpha \sum_i R_i = 1$$

HITS (Hyperlink induced topic search)

- Jon Kleinberg
CS - Cornell University

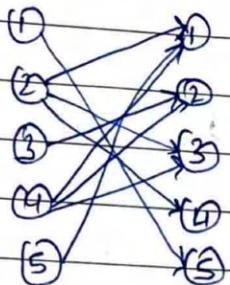


Assumption - Good hubs ~~are~~ those that link to good authorities. Good authorities are linked to by good hubs



considering every node to be both a hub & an authority we make a bipartite graph.

hubs authorities



authorities initial score based on in degrees \rightarrow

3	7
2	
2	
1	
1	

hubs score based on authorities initial score \rightarrow

1	7	→	3	1	3
6		→	2	1	2
2		→	2	0	2
7		→	1	1	1
3		→	1	1	1

next iteration for authority score \rightarrow

$$\begin{bmatrix} 16 \\ 9 \\ 13 \\ 6 \\ 1 \end{bmatrix} \leftarrow \begin{bmatrix} 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ 6 \\ 2 \\ 7 \\ 3 \end{bmatrix}$$

& so on

authorities $a_i = \alpha \sum_j A_{ij} h_j$ $\xrightarrow{\text{hubs}}$
 $h_i = \beta \sum_j A_{ji} a_j$

$$\begin{aligned} \bar{a} &= \alpha A \bar{h} \\ \bar{h} &= \beta A^T \bar{a} \end{aligned} \rightarrow \text{matrix form}$$

$$\begin{aligned} \bar{a} &= \alpha \beta A A^T \bar{a} \\ (\alpha \beta)^{-1} \bar{a} &= A A^T \bar{a} \end{aligned} \longrightarrow \lambda v = Av \rightarrow \text{eigen vector form}$$

$$(\alpha \beta)^{-1} \bar{h} = A^T A \bar{h}$$

$$A^T A = A A^T$$

\therefore eigen values for both \bar{a} & \bar{h} will be the same, but eigen vectors will be different

\therefore Here we can similarly conclude that the eigen vector corresponding to the highest eigen value will be only of use to us.

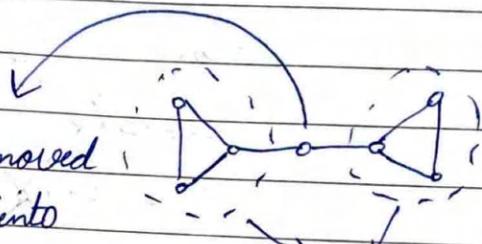
Centrality

Degree
 Eigen Vector
 Katz
 -simrank
 Page Rank
 HITS

closeness $\langle \frac{1}{d_{ij}} \rangle$
 Betweenness node
 edge

Betweenness Centrality

only node which when removed
 will divide the graph into
 2 strongly connected components



strongly connected components

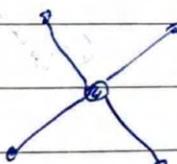
betweenness centrality of a node i

$$C_B(i) \propto \sum_{s,t} \sigma_{st}(i) / g_{st}$$

σ_{st} → no. of shortest paths from node s to t
 where node i lies in the shortest path.
 i is not s or t .

g_{st} → total no. of shortest paths b/w nodes s & t .

$$\therefore C_B(i) = \sum_{s,t} \frac{\sigma_{st}(i)}{g_{st}}$$



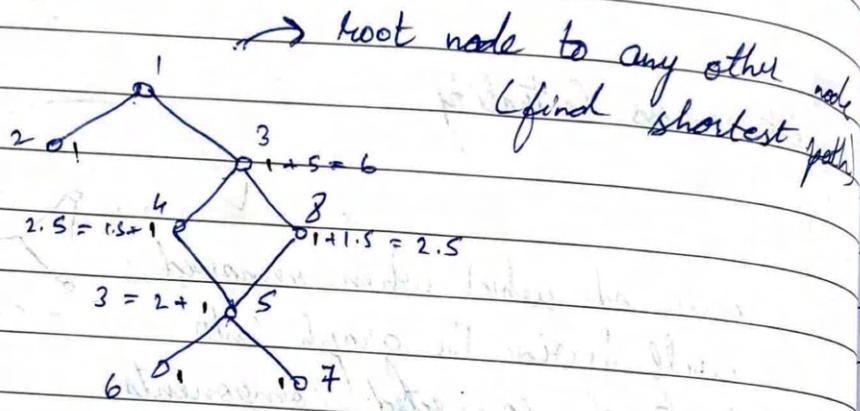
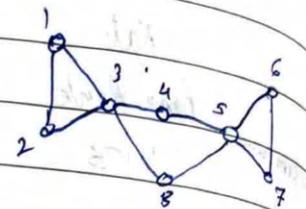
highest betweenness centrality of the central node

no. of shortest paths that pass through the central node → $n-1 C_2$ (all $n-1$ vertices excluding central vertex)

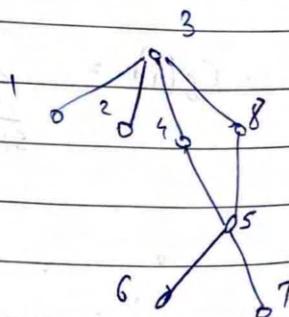
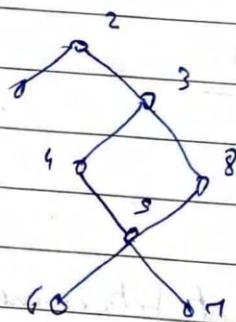
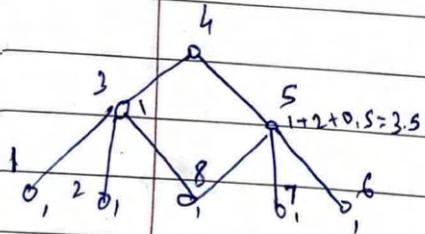
Normalized betweenness centrality

$$C_B^*(i) = \frac{C_B(i)}{\binom{n-1}{2}}$$

Now we'll make a BFS structure for the graph, where cycles are possible.



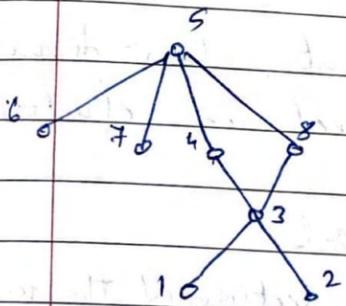
- assign a value to every node except the root node.
- Thinking terms of flow. It flows from leaf to root. Initially we assign unit flow.
- We draw the BFS structure of considering every node as root node.



total flow

Root Nodes

	1	2	3	4	5	6	7	8	Total	B/w
scores of every node	0	1	1	1	1	1	1	1	7	0
2	1	0	1	1	1	1	1	1	7	0
3	6	6	0	3.5	3	3	3	3.5	28	21
4	2.5	2.5	2.5	0	2.5	2.5	2.5	1	16	9
5	3	3	3	3.5	0	6	6	3.5	28	21
6	1	1	1	1	1	0	1	1	7	0
7	1	1	1	1	1	1	0	1	7	0
8	2.5	2.5	2.5	1	2.5	2.5	2.5	0	16	9



the flow sent upwards is not always half

if this node sends flow upward the ratio will be 1:2.

To calculate betweenness, we'll subtract the initial scores from total. In this example, the no. to be subtracted is 7. \therefore initial score assigned will be 1 for 7 cases & 0 for one case.

vertices 3 & 5 are most imp. in this case in terms of betweenness centrality.

Oct - 10

Final Submission - Nov 20

abstract submission -

- working title
- both groupmembers
- abstract - describe the dataset
- Type of analysis goal/aim (questions you're raising)
- source citation

DOMS

Page No.

Date _____
Date _____

Python - Networkx package.

Zachary Karate Club dataset - 34 nodes, 78 edges

→ very strong dataset. (download .nmf from Newman)

- importance / centrality
- density
- Adjacency matrix
- probability distribution

group of nodes (Cohesive behavior)

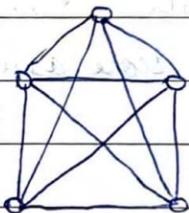
compactness → everybody knows everyone else



reachability → within a few hops its possible to reach all other nodes.

Density → high density within a component, low density b/w 2 components. Used in clustering algorithms.

- microscopic - single level
- mesoscopic - a small section of the system
- macroscopic - entire system



→ close tied structure

every node is connected with every other node

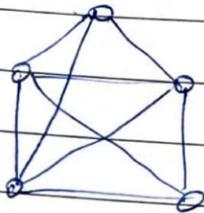
complete sub-graph aka Clique

to find all the cliques in a graph

→ the problem is NP-hard; minimal ^{subset} of vertices which forms a complete graph.

Instead of complete subgraph, we look for k -plex.

1 -plex is a clique.



→ 2 plex

at least connected with 3 other vertices. 5 nodes

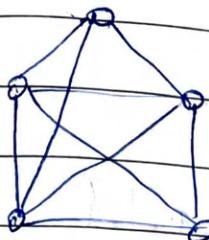
$$\therefore k = 2$$

$$n \swarrow \quad \nwarrow \quad 5 - 2 = 3$$

$\rightarrow k$ → no. of edges from each node

Instead of complete subgraph, we look for k -plex.

1 -plex is a clique.



→ 2 plex

at least connected with 3 other vertices... 5 nodes
 $\therefore k = 2$

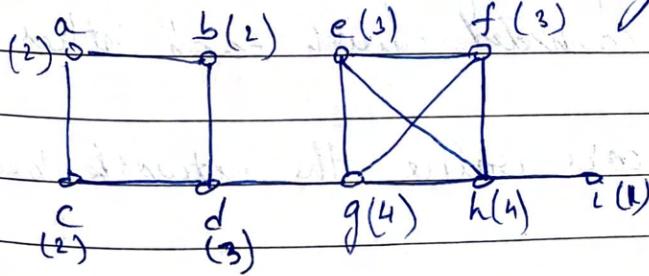
$$\begin{matrix} n & \xrightarrow{k} & \xrightarrow{\text{no. of edges from each node}} \\ S - 2 = 2 & 3 \end{matrix}$$

k -core →

remove node with the lowest degree.

after doing that if any node has degree less than or equal to the removed node, remove that too. — recursive process.

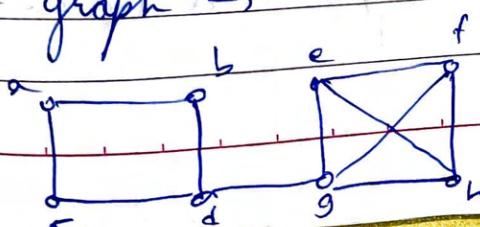
all the nodes removed belong to the same k -core value.



removing vertex i →

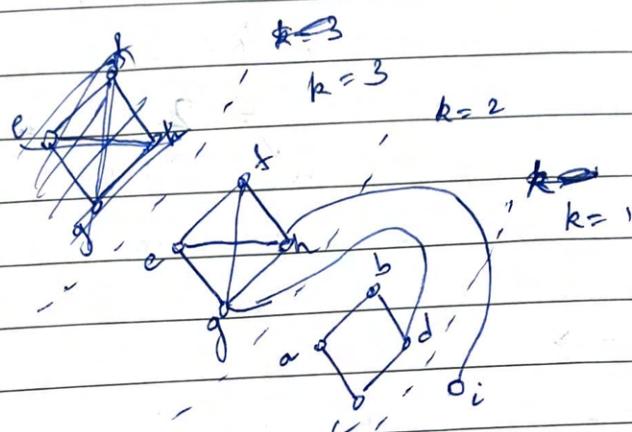
vertex i will belong to $k=1$.

new graph →



next iteration we can remove a vertex a.

iteration	$k = 1$	$k = 2$	$k = 3$
1	{i, j}		
2		{a, j}	
3		{a, b, c, j}	
4		{a, b, c, d, j}	
5			{e, f, g, h}



as we move into the core, the density of graph increases, $k=3 \rightarrow$ more towards the core.

We are trying to find group of nodes that are cohesive - highly connected with each other.

as we increase the core value, the network becomes more & more dense.

Here we have a group of nodes based centrality.

high degree nodes inside a higher core value are more influential.

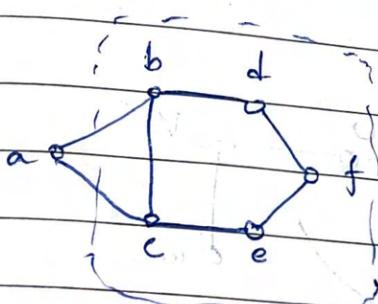
k -core allows us to identify more influential spreaders. Node with inside a higher core value will be more influential than a node at lower core value. no matter the degree of those nodes. \therefore Hubs are not the most important.

Distance based grouping \rightarrow

- k -clique
- k -clan
- k -club

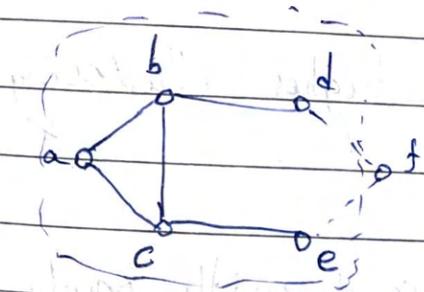
k -clique \rightarrow

it puts all those vertices that are at a distance k with each other.



every node is at a max. of dist. 2 from each other

They can be grouped at 2-clique.
distance is shortest path



problem with k -clique.

this 2-clique includes a path from e to d passing through f. but f is not part of the said 2-clique.

k -clan \rightarrow

- extract the k -clique
- check the diameter of the k -clique graph.
- if diameter $\leq k$
 - then k -clan
 - else
not k -clan

(this is to avoid the problem with k -clique)

k -club \rightarrow

- if diameter of a ~~max~~ maximal subset of a graph is k , then k -club.
- no more nodes can be added.

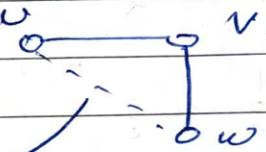
Transitivity \rightarrow (Clustering)

$v - v$

$v - w$

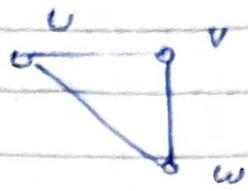
are v & w related?

(especially in the context of social networks)

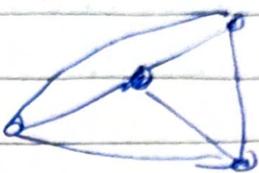


- Social networks have much higher clustering than any other network.
- If all your friends are friends with each other, then it will be a complete graph.

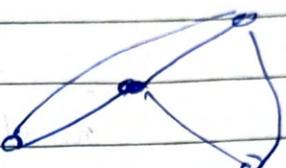
- each of these triangles increase the transitivity



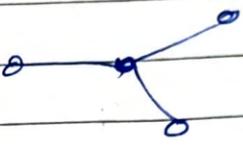
- case where all my friends are friends with each other (perfectly transitive)



- case where not all my friends are friends with each other



- case where none of my friends are friends with each other.



The total no. of triangles present in the network gives us how many of the connections are transitive.

We normalize the no. of triangles obtained by the total no. of triangles possible in the network.

This normalized quantity is called the clustering coefficient.

If clustering coefficient = 1, then completely connected graph. — perfect transitivity

Lack of transitivity \rightarrow clustering coefficient = 0 (star graph).



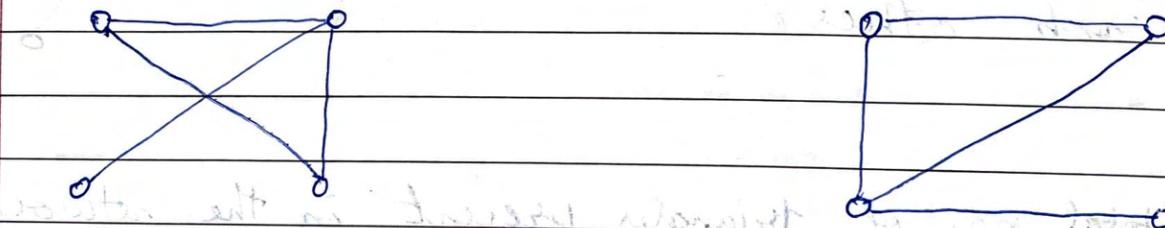
$$C = \frac{\text{no. of closed paths of length 2}}{\text{no. of paths of length 2}}$$

for every triangle, there are 6 paths of length 2. (both sides)

$$C = \frac{6 \times \text{no. of } \Delta's}{\text{no. of paths of length 2}}$$

$$C = \frac{3 \times \text{no. of } \Delta's}{\text{no. of connected triplets}}$$

→ easiest to implement



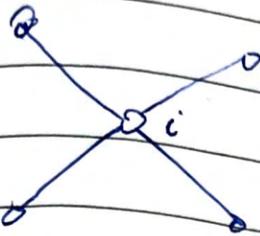
$$\text{probability of connection of 2 nodes} = \frac{|E|}{\binom{n}{2}}$$

Local clustering coeff. →

$$C_i = \frac{\text{no. of edges b/w the neighbours}}{\binom{k_i}{2}}$$

$k_i \rightarrow$ degree of vertex i

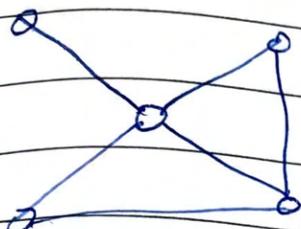
$\binom{k_i}{2} \rightarrow$ no. of edges possible b/w neighbours



$$c_i = \frac{0}{4c_2} = 0$$



$$\begin{aligned} c_i &= \frac{1}{4c_2} \\ &= \frac{1}{6} \end{aligned}$$



$$\begin{aligned} c_i &= \frac{2}{4c_2} \\ &= \frac{1}{3} \end{aligned}$$

Local clustering coeff. is also a centrality measure, \because it can be calculated for each node.

$$\langle c \rangle = \frac{1}{N} \sum_i c_i \rightarrow \text{clustering coeff for the entire network}$$

\hookrightarrow no. of nodes (to normalize).

the more correct definition of clustering coeff. is the previous one, but this is more used.

structural holes \rightarrow absence of links to connect clusters.

