NETWORK ANALYSIS

&

MINING

unit - 1

feedback/corrections: vibha@pesu.pes.edu

VIBHA MASTI

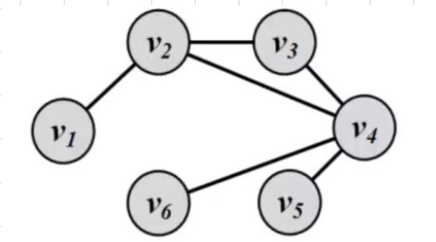
TYPES & REPRESENTATION of GRAPHS

CS224W - Machine Learning with Graphs
Stanford / Fall 2019
Jure Leskovec

REPRESENTATIONS

1. Adjacency Matrix

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

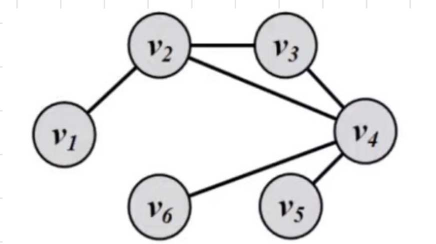


	V_1	V_2	V_3	V_4	V_5	V_6
V_1	0	1	0	0	0	0
V_2	1	0	1	1	0	0
V_3	0	1	0	1	0	0
V_4	0	1	1	0	1	1
V_5	0	0	0	1	0	0
V_6	0	0	0	1	0	0

- Social media networks are very sparse adjacency matrices

2. Adjacency List

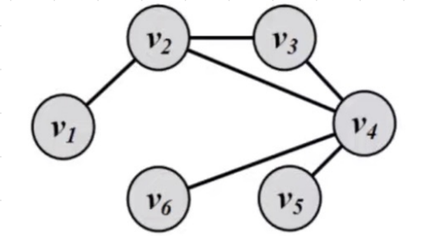
- List usually sorted based on preferences



Node	Connected To
v_1	v_2
v_2	v_1, v_3, v_4
v_3	v_2, v_4
v_4	v_2, v_3, v_5, v_6
v_5	v_4
v_6	v_4

3. Edge list

- Edge (u,v) when u is connected to v via an edge



(v_1, v_2)
(v_2, v_3)
(v_2, v_4)
(v_3, v_4)
(v_4, v_5)
(v_4, v_6)

TYPES

1. Null graph

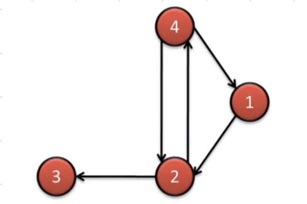
- $G(V, E)$ where $V = E = \phi$

2. Empty graph

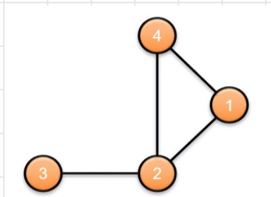
- $G(V, E)$ where $V \neq \phi$ but $E = \phi$

3. Directed and Undirected Graphs

- For directed graphs, $A_{ij} \neq A_{ji}$ $(A \neq A^T)$



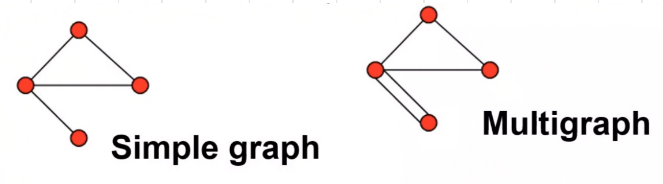
- For undirected graphs, $A_{ij} = A_{ji}$ $(A = A^T)$



4. Simple and Multigraphs

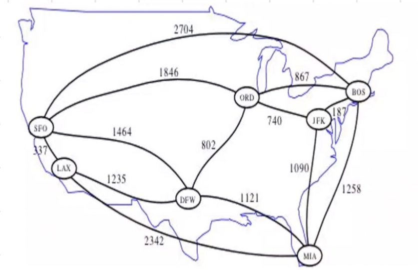
- Simple graphs : graphs where only single edge can be between any pair of nodes

- Multi graphs: multiple edges between two nodes
  - eg: web graphs (multiple hyperlinks between 2 webpages)
  - adjacency matrix: numbers larger than 1 as entries



5. Weighted graphs

- Weights: could be distances



6. Web graph

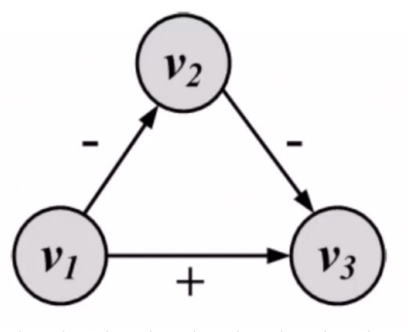
- Directed multigraph

- Nodes: sites, edges: links

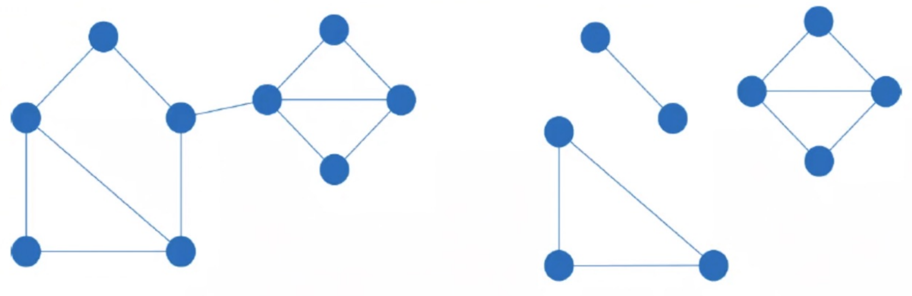
- Two sites can have multiple links and loops

7. Signed Graphs

- Binary weights (eg: friends & foes)



8. Connected and Disconnected Graphs



Connected Graph

Disconnected Graph
Includes 3 components.

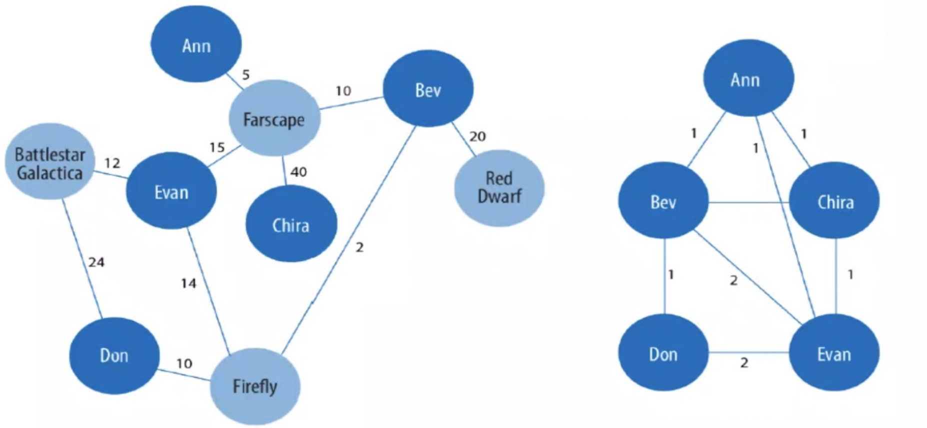
9. Bipartite and Monopartite Graph

- **Bipartite:** nodes belong to 2 sets such that there is no relation among members of a set; relations only between members of sets
  - eg: users vs items (recommendation system)

- **Monopartite:** only one type of node

- Bipartite graph can be projected into mono partite graphs with inferred connections

- K-partite: K node types



Graph 1

Viewers and TV Shows

Bipartite Graph

Relationship weights = Number of episodes watched

Graph 2

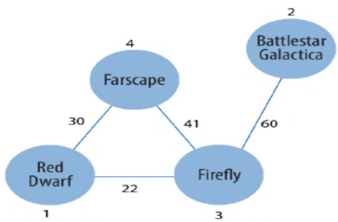
Projection of Viewers

Monopartite Graph

Relationship weights = Number of shows in common

- Relationship weights of projected graph: similarity measure

Projection of bipartite graph based on inferred relation



Graph 3

Projection of TV Shows

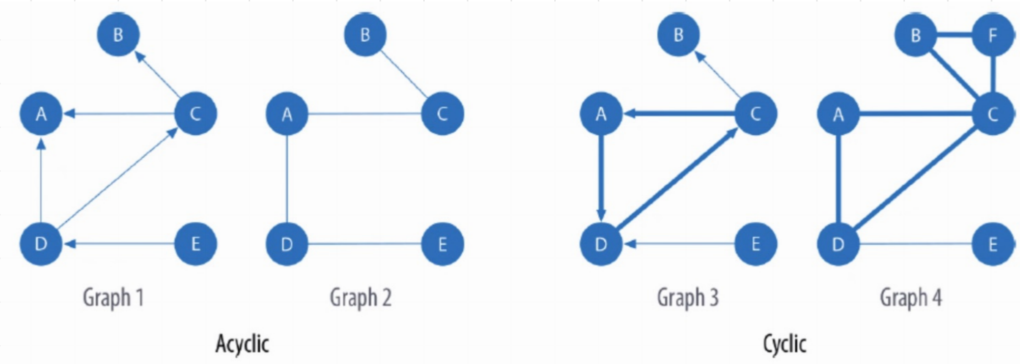
Monopartite Graph

Node weights = Number of active viewers

Relationship weights = Combined episodes watched by viewers in common

10. Cyclic and Acyclic Graphs

- Acyclic: impossible to start and end on same node without retracing steps



11. Affiliation Networks

- Unipartite networks: network of friends

- Bipartite networks:

  (a) Affiliation network: people not connected to each other, organisations not connected to each other; people individually affiliated to one or more organisations

  (b) Membership of people on corporate board: members not connected, boards not connected; members connected to one or more boards

- Tripartite networks
  - users, communities, interest terms

  - A network of LiveJournal users
  - **users** (part 1), **communities** (part2), and **interest terms** (part 3).
  - A user belongs to a community (1-> 2 type edge)
  - A user is interested in a term (a 1-> 3 type edge)
  - A community declares a term as an interest (a 2 -> 3 type edge)

12. Heterogeneous graphs

- Nodes & edges of diff types handled differently

13. Homogeneous graphs

- Nodes & edges instances of single type

14. Dynamic and Static Graphs

- Dynamic graphs change over time

15. Knowledge Graph

- Directed multi relational graph where an edge represents a tuple <h,r,t> — head, relation, tail

- Eg: <Star Trek, Genre, Science Fiction>

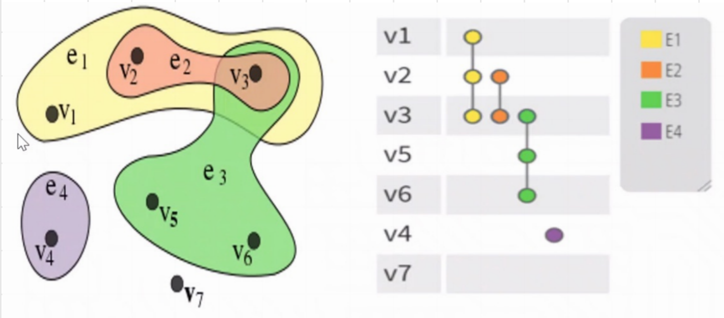
16. Hypergraph

- Generalisation of undirected graphs in which edges are subsets of 2 or more vertices

- Hyperedges connect arbitrary number of nodes

- Size of vertex set: order of hypergraph

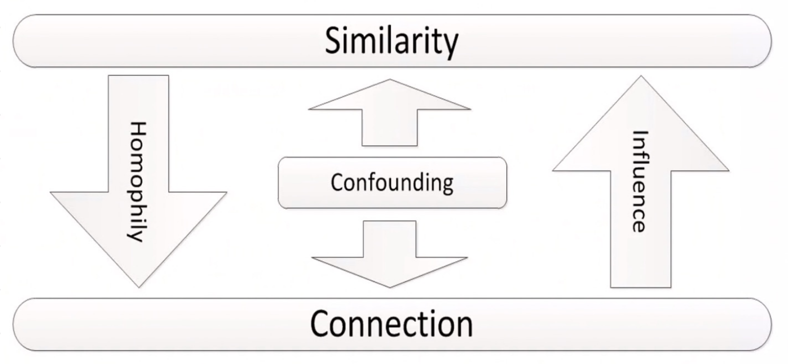
- No. of hyperedges: size of hypergraph



- $X = \{v_1, v_2, v_3, v_4, v_5, v_6, v_7\}$
- $E = \{e_1, e_2, e_3, e_4\}$
- $V = \{\{v_1, v_2, v_3\}, \{v_2, v_3\}, \{v_3, v_5, v_6\}, \{v_4\}\}$
- Order = 7 (no. of nodes)
- Size = 4 (no. of hyperedges)

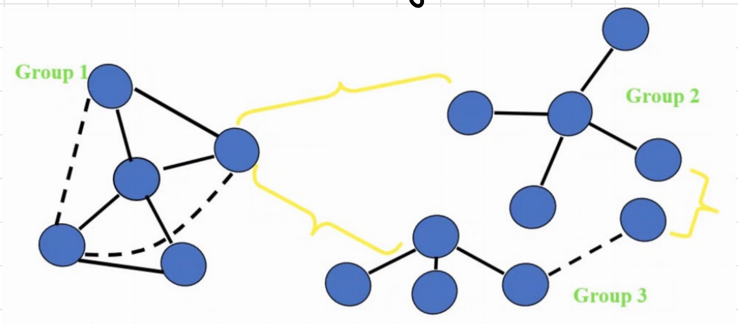
Selection & Social Influence

- Homophily: similarity
- Selection: friend selection
- Social influence: peer influence (eg: junk food, drinking etc)
- Context dependent
- Either pure social, pure selection or combination of social + selection



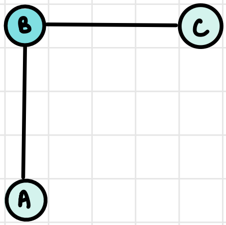
STRUCTURAL HOLE

coined by Ronald Burt



- Holes in information flow between groups

Brokerage Opportunity for Ego



- B: ego taking advantage of structural hole between A & C
  - in ego networks, ego is central node that all other nodes are connected to, and alters are all the surrounding nodes directly connected to the ego

- Real-life interpretation of brokerage opportunity

1. If A, B and C are in the same group
   - Coordination Broker: B brokers between members of the same group
   - Eg: manager of a team

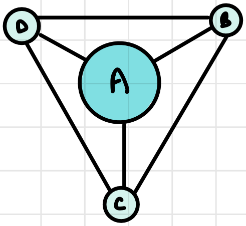
2. A and C in the same group, B different
   - Consultant Broker: B connects members of the group together who want 2 different ends of a service/task
   - Eg: stock broker connecting a buyer and a seller

3. A and C are in different groups
   - Representative Broker: B negotiates on behalf of A to C
   - Gatekeeper Broker: transaction begins at A, passes to C's group's gatekeeper B
   - Eg (gatekeeper): B - census bureau, C - govt dept, A - public

4. A, B, C different groups
   - Liason Broker: B plays neutral role between A & C
   - Eg: A - actor, B - agent, C - producer

Measures Related to Structural Holes

1. Effective size: number of alters that ego has minus average number of ties each alter has to other alters



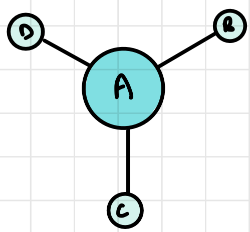
A: ego
B, C, D: alters

- A has 3 ties, each tied to 2 others (many redundant ties)

- Average degree of other alters is 2

- Effective size of A's ego network = 3 − 2 = 1

- If no alters interconnected, effective size of A's ego network is 3 − 0 = 3



2. **Efficiency:** portion of ego's ties that are non-redundant

   - Efficiency = $\dfrac{\text{Effective size of A's ego net}}{\text{size of A's ego net}}$

   - For first graph, size = 3 (3 alters) and efficiency = 1/3

   - For second graph, size = 3 and efficiency = 1

EGO NETWORKS

- Focus on individual

- Ego network data commonly arises in 2 ways
  1. Collect data from ego (respondent) about interactions with alters in diff social settings & ask about ties between alters
  2. Snowball method: ask ego to identify alters, then ask those alters to identify their ties/alters

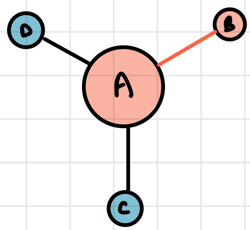
- Use ego nets to make predictions about ego (health, longevity, economic success etc.)

- Effect of social context on individual attributes, behaviours and conditions

- Sum of ego networks = social network

Homophily in Ego Networks

- Portion of ties in ego net that are homophilous
  - $\frac{\text{count of ties where ego and alter share same attr}}{\text{total no. of ties in ego net}}$
  - correlation between ego attr and alter attr
  - Eg: nodes are either members of class red (A, B)
    homophily = $\frac{1}{3}$



- Assortative mixing / assortavity is a bias in favour of connections between network nodes with similar characteristics
  - Homophily

Density of Ego Network

- Size = no. of alters

- Density = fraction of total no. of possible ties in the ego net that are actually present, excluding the ego

- consider an ego connected to n alters

- Assume ego net has L connections between alters

- Assuming undirected graph, total no. of possible edges b/w the alters = $\frac{n(n-1)}{2}$

$$\text{Density}_{\text{undir}} = \frac{L}{n(n-1)/2}$$

- Assuming directed graph, total no. of possible edges b/w the alters = $n(n-1)$

$$\text{Density}_{\text{dir}} = \frac{L}{n(n-1)}$$

Strength of Weak Ties

- Mark Granovetter

- People with many weak ties can gain speedy advantages in learning about and cashing in on new entrepreneurial opportunities

- Irony: weak ties provide stronger form of social capital for career advancements

- Eg: LinkedIn – connect to weak ties to get opportunity

Strong Ties

- Determined by 3 characteristics
  1. Long duration of relationship
  2. Closeness of relationship – close/very close
  3. Frequency of contact – frequent

- In network maps, strong ties are solid lines

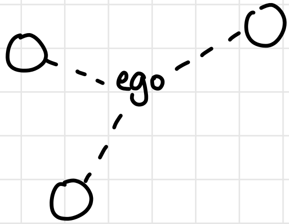
- Benefits
  1. Generally trustworthy
  2. Provides depth of information

Weak Ties

- Determined by 3 characteristics
  1. short duration of relationship
  2. Closeness of relationship – not close
  3. Frequency of contact – infrequent

- In network maps, weak ties are dotted lines

- Benefits
  1. Provide unique perspectives
  2. Helpful for identifying opportunities



Clique

- Subset of network in which actors are more closely and intensely tied to one another than they are to other members of the network

- Human tendency: on the basis of age, gender, race, religion, ethnicity etc.

- Dyad: clique of 2 actors (smallest)

Cliques vs Weak Ties

- Cliques are great at knowledge preservation, not good at knowledge generation (cliques are knowledge reservoirs)

- Weak ties serve as bridges

- Weak & strong ties are just weighted edges with appropriate weights

- Eg: finding a job
  - Strong: more motivation to help you
  - Weak: less likely overlap with leads
  - Study: most job referrals come through those we see rarely

- Board of directors: must model such that boards are independent and not influenced by vested interest

Centrality Analysis in Graphs

- Centrality . importance of a node in a graph
- Nodes maybe important from an angle

Normalised Degree Centrality

- Normalised by max possible degree (n-1)

$$C_d^{norm}(v_i) = \frac{d_i}{n-1}$$

} meso-level analysis by Freeman

- Normalised by max degree

$$C_d^{max}(v_i) = \frac{d_i}{\max_j d_j}$$

- Normalised by degree sum

$$C_d^{sum}(v_i) = \frac{d_i}{\sum_j d_j} = \frac{d_i}{2|E|}$$

- Linton Freeman developed basic measures of centrality
- Degree centrality: for undirected graphs, unweighted graphs (as it biases towards stronger ties)
- Degree centrality only used to make comparisons between actors of the same net or diff nets of same size
  - Still should not compare networks that differ greatly in size (higher norm score for smaller graphs)
- Neighbour focused, not global

- Indegree centrality : no of ties received by an actor ; measure of popularity or prestige

$$C_d(v_i) = d_i^{in}$$

- Outdegree centrality: no of ties given by an actor ; measure of expansiveness or gregariousness

$$C_d(v_i) = d_i^{out}$$

- Combination

$$C_d(v_i) = d_i^{in} + d_i^{out}$$

BETWEENNESS CENTRALITY

- Degree centrality only looks at ego network (immediate ties) and does not consider the rest of the network
- Betweenness centrality: how much potential control an actor has over the flow of information
- How many times an actor sits on the geodesic / shortest path between two actors together
- How central v_i's role in connecting a pair of nodes s and t is

$$C_b(v_i) = \sum_{s \neq t \neq v_i} \frac{\sigma_{st}(v_i)}{\sigma_{st}}$$

all s & t values

- σ_{st} : no. of shortest paths from s to t (info pathways)
- $\sigma_{st}(v_i)$: no. of shortest paths from s to t that pass through v_i

- Best case: $\sigma_{st}(v_i) = \sigma_{st}$ or v_i lies on all shortest paths

$$\frac{\sigma_{st}(v_i)}{\sigma_{st}} = 1$$

- Max value of c_b in undirected graphs

$$c_{b\,max} = \sum_{\underbrace{s \neq t}_{\text{s \& t chosen from n-1 nodes (nth is } v_i)} \neq v_i} 1 = {}^{n-1}C_2 = \binom{n-1}{2} = \frac{(n-1)!}{2!\,(n-3)!}$$

$$c_{b\,max} = \frac{(n-1)(n-2)}{2}$$

- Max value of c_b in directed graphs

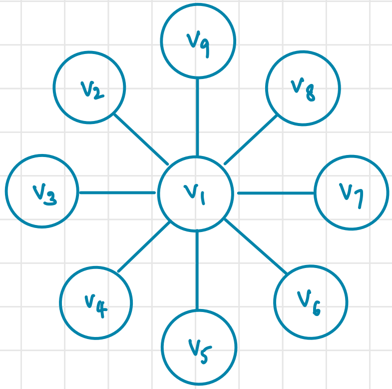
$$c_{b\,max} = \sum_{s \neq t \neq v_i} 1 = {}^{n-1}P_2 = \frac{(n-1)!}{(n-3)!} = (n-1)(n-2)$$

$$c_{b\,max} = (n-1)(n-2) = 2\,{}^{n-1}C_2$$

- Normalised betweenness centrality

$$c_b^{norm}(v_i) = \frac{c_b(v_i)}{2({}^{n-1}C_2)}$$

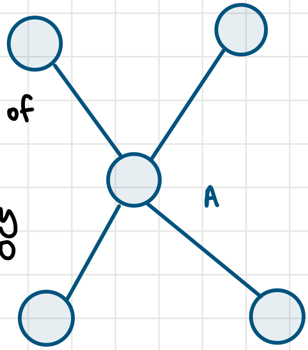
Q: Calculate $c_b(v_1)$ for the given graph

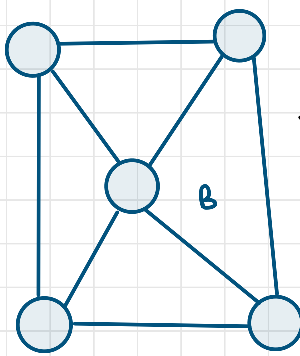


- All paths between s & t go through v_1 (all values of s & t)
- $\therefore \quad c_b(v_1) = {}^{8}C_2$
- Normalised $c_b^{norm}(v_1) = 1$
- Betweenness centrality for any other node in the graph is 0
- Higher betweenness centrality characterised by more structural holes
- Low density score means more structural holes

diversity of info

density score = 0





same info

density score = 0.67

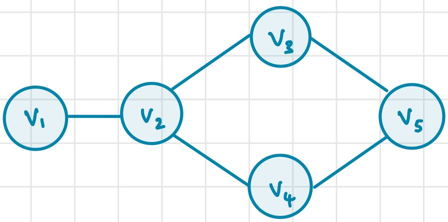
Closeness Centrality

- How close a node is from the rest of the graph
- Closeness centrality = 0 means it has no neighbours and is severed from the rest of the net
- Closeness centrality = 1 means it is directly connected to every other node
- Central nodes: smaller average shortest path lengths to other nodes

$$C_c(v_i) = \frac{1}{\bar{l}_{v_i}} \quad \text{where} \quad \bar{l}_{v_i} = \frac{1}{n-1} \sum_{v_j \neq v_i} l_{i,j}$$

length of geodesic b/w i & j

Q: Calculate $C_c(V_3)$, $C_c(V_1)$, $C_c(V_2)$, $C_c(V_4)$, $C_c(V_5)$



$n = 5$

$l_{3,1} = 2$

$l_{3,2} = 1$

$l_{3,4} = 2$

$l_{3,5} = 1$

$$\bar{l}_3 = \frac{1}{4}(2+1+2+1) = \frac{3}{2}$$

$$C_c(V_3) = \frac{2}{3} = 0.67$$

$C_c(V_1) = 1/((1+2+2+3)/4) = 0.5$

$C_c(V_2) = 1/((1+1+1+2)/4) = 0.8$

$C_c(V_4) = 1/((2+1+2+1)/4) = 0.67$

$C_c(V_5) = 1/((3+2+1+1)/4) = 0.57$

Harmonic Centrality

- Mean reciprocal distance

PAGE RANK

Discrete Markov Process (recall MI)

- Series of experiments performed at regular time intervals
- Always same set of possible outcomes
- Discrete steps of time
- Outcomes called states

- Model can be in any one state at any given timestep
- Next step: can stay in same state or move to another state
- Movement b/w states: probability

Stochastic Transition Probability Matrix

- $a_{ij} = P(q_{t+1} = S_j \mid q_t = S_i)$
- Sum of column = 1
- Stochastic: all row sums = 1, all values non-negative

State		Time t+1				Total
		S_1	S_2	S_3	S_4	
Time t	S_1	a_{11}	a_{12}	a_{13}	a_{14}	1
	S_2	a_{21}	a_{22}	a_{23}	a_{24}	1
	S_3	a_{31}	a_{32}	a_{33}	a_{34}	1
	S_4	a_{41}	a_{42}	a_{43}	a_{44}	1

Eigen Decomposition Theorem

- Square matrix can be decomposed into eigen vectors & eigen values
- Let $C_{M \times M}$ be a matrix, x_R be a column vector and λ_R be a constant

$$Cx_R = \lambda_R x_R$$

- The n column vectors x_R and the n values of λ_R are the right eigenvectors and eigenvalues respectively
- Principle right eigenvector: corresponds to eigenvalue of largest magnitude
- Left eigenvector is a row vector

$$x_L C = \lambda x_L$$

Steady State

- Probability for transitioning to a state reaches a limiting value as $t \to \infty$
- Probability vector × Transition matrix = Probability vector
- Steady state: eigenvector for stochastic matrix

ERGODIC MARKOV CHAIN

- Let Markov chain start at $t=0$ in state i
- If there exists a timestamp value T_0 and a state j such that for all $t > T_0$ the probability of being in state j is > 0, the chain is ergodic
- Conditions for ergodicity
  1. Irreducibility: sequence of transitions of non-zero probability from any state to any other
  2. Aperiodicity: states are not partitioned into sets such that all state transitions occur cyclically from one set to another

STEADY STATE THEOREM

- For any Ergodic Markov chain, there is a unique steady state probability vector π (left principle eigenvector of transition matrix P)

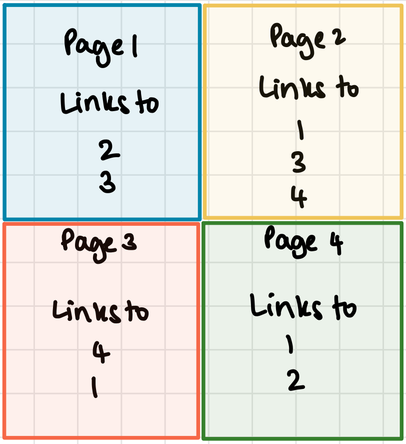
$$\pi P = \lambda \pi \qquad \lambda = 1$$

Page Rank Formula

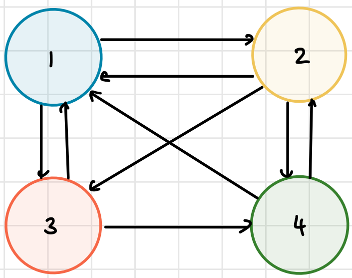
- Web: network of webpages
- Page rank: numeric value representing importance of a page on the web
- One hyperlink from one page to another: one vote
- Needs to be immune to link spamming

- See: Big Data unit 2

- Model www as a directed graph

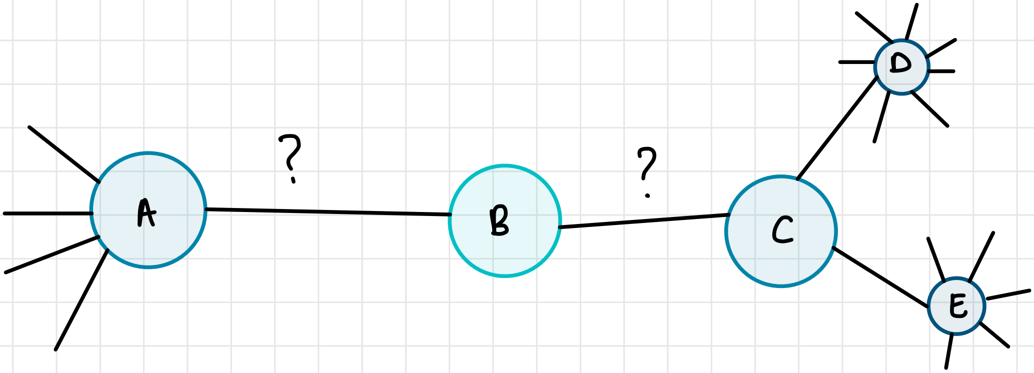


- Directed graph

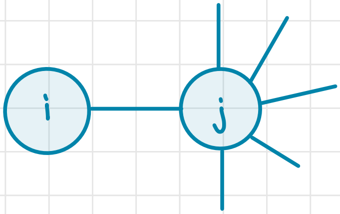


SPECTRAL ANAYSIS TO EXPLAIN PAGE RANK

- Eigenvector centrality : generalise degree centrality (global centrality) – undirected graphs
- Think: which edge A-B or B-C should be severed to make B safe from STDs?
  - recursive (depends on probability of being infected by A or C, their neighbours, so on)



- Degree centrality: uses no. of neighbours
- We want centrality of v_i to be a function of its neighbours' centralities
- Proposal: centrality of $v_i \propto \sum_j$ centrality of neighbour
- Assume everyone has an initial score (rank) of 1 and update centralities recursively
- Let X = popularity vector



$X_j(0)$ = popularity of j at $t=0$

$X_j(1) = \sum_i A_{ij} X_j(0)$ for all nodes $i \neq j$ where A_{ij} is the number of links between i & j (adjacency matrix)

- Vector form

$$X(1) = A \cdot X(0)$$

- At time t

$$x(t) = A\,x(t-1) = A^2\,x(t-2) = \dots = A^t\,x(0)$$

$$x(t) = A^t\,x(0)$$

- For some large value of t, it will stabilise

- Let $x(0)$ be a linear combination of eigenvectors of A

$$x(0) = \sum c_i v_i$$

- Plugging in, $x(t) = A^t \sum c_i v_i$

- We know $Av = \lambda v$ (v - eigenvector)

$$\therefore\ A^t v = \lambda^t v$$

- Plugging in,

$$x(t) = \sum \lambda_i^t c_i v_i$$

- Divide by principle eigenvector λ_1^t

$$\frac{x(t)}{\lambda_1^t} = \sum \frac{\lambda_i^t c_i v_i}{\lambda_1^t}$$

$$\lim_{t \to \infty} \frac{x(t)}{\lambda_1^t} = c_1 v_1 \qquad \left(\text{all other } \frac{\lambda_i}{\lambda_1} \to 0\right)$$

- Popularity value converges to principle eigenvector of adjacency matrix

For directed graphs

- An initial popularity β is added to the centrality values for all nodes

- α depends on neighbourhood

$$x_i(1) = \alpha \sum A_{ji}\, x_j(0) + \beta$$

$$C_{katz}(v_i) = \alpha \sum_{j=1}^{N} A_{ji}\, C_{katz}(v_j) + \beta$$

- Vector Form

vector of all 1's

easier to solve this iteratively $\rightarrow$ $C_{katz} = \alpha A^T C_{katz} + \beta(1)$

harder to solve $\rightarrow$ $C_{katz} = \beta(I - \alpha A^T)^{-1} \cdot 1$

- PageRank – divide value of passed centrality by number of outgoing links

$$C_p(v_i) = \alpha \sum_{j=1}^{n} A_{ji} \frac{C_p(v_j)}{d_j^{out}} + \beta$$

RANDOM SURFER TO EXPLAIN PAGE RANK

- Model of user behaviour

1. Initially every web page chosen uniformly at random
2. Choose a hyperlink on the page with probability α
3. With probability $1-\alpha$, perform random walk on web by randomly choosing a node and then restart at step 1

Algorithm - Basic Page Rank

1. In a network of **n nodes**, we assign all nodes same initial PageRank **1/n**
2. We choose **no of steps k**
3. We then perform **a sequence of k updates** as
   1. Each page divides its current PageRank equally across its outgoing links and passes that to the pages it points to
   2. Each page updates its PageRank to be sum of the shares that it receives
4. So, the PageRank **remains constant (no scaling needed)** – neither created nor destroyed **but is just moved around**
5. As K -> ∞ , the **PageRank of all nodes converge to limiting values** (don't change any further)

Algorithm - Scaled Page Rank

1. First: apply basic PR update rule, second: scale down by a factor of s so that total PR is down from 1 to s ($s < 1$)
2. Divide residual PR $(1-s)$ into n nodes as $\frac{1-s}{n}$

Limitations of Basic Page Rank

1. Rank Sinks
   - Page has no outgoing edges, only incoming edges
   - Monopolise scores

2. Hoarding / Circular Reference
   - Group of pages that only link between each other

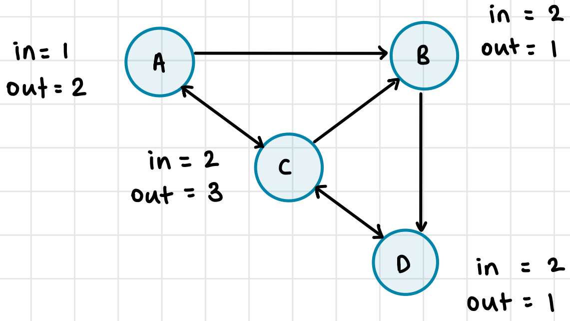
3. Dangling nodes
   - Isolated node (no links at all)

Solution

Teleporting: random surfer will teleport to random URLs for a proportion of hops $(1-\alpha)$

Usually: $\alpha = 0.85$, $1-\alpha = 0.15$

Basic PR



	Iteration 0	Iteration 1	Iteration 2	Rank
A	1/4	1/12	1/8 = 0.125	4
B	1/4	5/24	5/24 = 0.208	3
C	1/4	3/8	3/8 = 0.375	1
D	1/4	1/3	1/3 = 0.333	2

Iteration 1

$$PR(A) = \frac{PR(C)}{out(C)} = \frac{1/4}{3} = \frac{1}{12}$$

$$PR(B) = \frac{PR(A)}{out(A)} + \frac{PR(C)}{out(C)} = \frac{1/4}{2} + \frac{1/4}{3} = \frac{5}{24}$$

$$PR(C) = \frac{PR(A)}{out(A)} + \frac{PR(D)}{out(D)} = \frac{1/4}{2} + \frac{1/4}{1} = \frac{3}{8}$$

$$PR(D) = \frac{PR(B)}{out(B)} + \frac{PR(C)}{out(C)} = \frac{1/4}{1} + \frac{1/4}{3} = \frac{1}{3}$$

Iteration 2

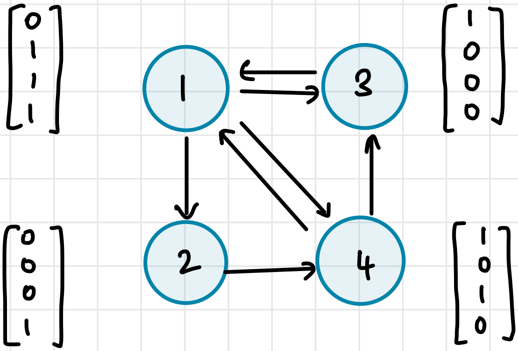
$$PR(A) = \frac{PR(C)}{out(C)} = \frac{3/8}{3} = \frac{1}{8}$$

$$PR(B) = \frac{PR(A)}{out(A)} + \frac{PR(C)}{out(C)} = \frac{1/12}{2} + \frac{3/8}{3} = \frac{1}{6}$$

$$PR(C) = \frac{PR(A)}{out(A)} + \frac{PR(D)}{out(D)} = \frac{1/12}{2} + \frac{1/3}{1} = \frac{3}{8}$$

$$PR(D) = \frac{PR(B)}{out(B)} + \frac{PR(C)}{out(C)} = \frac{5/24}{1} + \frac{3/8}{3} = \frac{1}{3}$$

Google PR - no need to divide $1-\alpha$ by n



$\alpha = 0.5$

Transition matrix (divide by no of outlinks)

$$
\begin{array}{cc}
 & \text{source} \\
\text{dest} & \begin{bmatrix} 0 & 0 & 1 & 1/2 \\ 1/3 & 0 & 0 & 0 \\ 1/3 & 0 & 0 & 1/2 \\ 1/3 & 1 & 0 & 0 \end{bmatrix}
\end{array} = T
$$

Initial page rank

$$x = \begin{bmatrix} 1/4 \\ 1/4 \\ 1/4 \\ 1/4 \end{bmatrix} = X(0)$$

Iteration 1

$$X(1) = (1-\alpha) + \alpha\, T X(0)$$

$$= 0.5 + 0.5 \times \begin{bmatrix} 0 & 0 & 1 & 1/2 \\ 1/3 & 0 & 0 & 0 \\ 1/3 & 0 & 0 & 1/2 \\ 1/3 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} 1/4 \\ 1/4 \\ 1/4 \\ 1/4 \end{bmatrix}$$

$$= 0.5 + 0.5 \times \begin{bmatrix} 3/8 \\ 1/12 \\ 5/24 \\ 1/3 \end{bmatrix}$$

$$= 0.5 + \begin{bmatrix} 3/16 \\ 1/24 \\ 5/48 \\ 1/6 \end{bmatrix} = \begin{bmatrix} 11/16 \\ 13/24 \\ 29/48 \\ 2/3 \end{bmatrix}$$

$$x(1) = \begin{bmatrix} 0.6875 \\ 0.5417 \\ 0.6042 \\ 0.6667 \end{bmatrix}$$