BACKGROUND ON GRAPHS

Graphs – definitions & representations

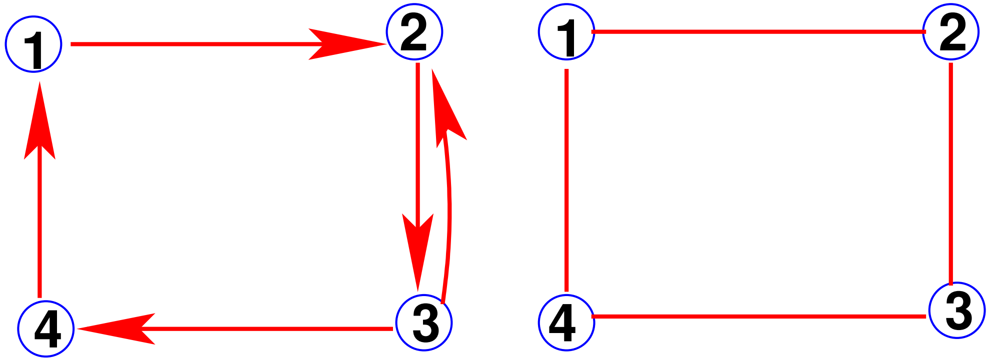
- Graph theory is a fundamental tool in many areas

Definition. A graph G is defined as a pair of sets $G = (V, E)$ with $E \subset V \times V$. So G represents a binary relation. The graph is undirected if the binary relation is symmetric. It is directed otherwise.

- V is the vertex set and E is the edge set
- A binary relation R in V can be represente by graph $G = (V, E)$ where:

$$(u, v) \in E \leftrightarrow u \; R \; v$$

Undirected graph $\leftrightarrow$ symmetric relation



Given the numbers 5, 3, 9, 15, 16, show the two graphs representing the relations

R1: Either $x < y$ or y divides x.

R2: x and y are congruent modulo 3. [mod(x,3) = mod(y,3)]

- $|E| \leq |V|^2$. For undirected graphs: $|E| \leq |V|(|V|+1)/2$.
- A sparse graph is one for which $|E| \ll |V|^2$.

Basic Terminology & notation:

- If $(u, v) \in E$, then v is adjacent to u. The edge (u, v) is incident to u and v.
- If the graph is directed, then (u, v) is an outgoing edge from u and incoming edge to v
- $Adj(i) = \{j | j \text{ adjacent to } i\}$
- The degree of a vertex v is the number of edges incident to v. Can also define the indegree and outdegree. (Sometimes self-edge $i \to i$ omitted)
- $|S|$ is the cardinality of set S [so $|Adj(i)|$ == deg(i)]
- A subgraph $G' = (V', E')$ of G is a graph with $V' \subset V$ and $E' \subset E$.

Representations of Graphs

- A graph is nothing but a collection of vertices (indices from 1 to n), each with a set of its adjacent vertices [in effect a 'sparse matrix without values']

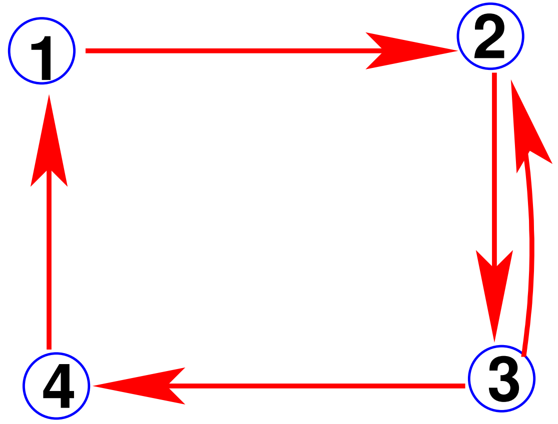
- For sparse graphs: use any of the sparse matrix storage formats - omit the real values arrays.

Adjacency matrix Assume $V = \{1, 2, \cdots, n\}$. Then the adjacency matrix of $G = (V, E)$ is the $n \times n$ matrix, with entries:

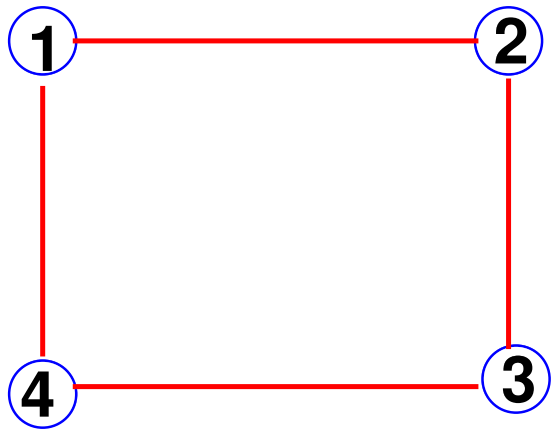
$$a_{i,j} = \begin{cases} 1 & \text{if } (i,j) \in E \\ 0 & \text{Otherwise} \end{cases}$$

Example:

$$\begin{bmatrix} & 1 & & \\ & & 1 & \\ & 1 & & 1 \\ 1 & & & \end{bmatrix}$$



$$\begin{bmatrix} & 1 & & 1 \\ 1 & & 1 & \\ & 1 & & 1 \\ 1 & & 1 & \end{bmatrix}$$



More terminology & notation

- Given $Y \subset X$, the section graph of Y is the subgraph $G_Y = (Y, E(Y))$ where $E(Y) = \{(x,y) \ \in \ E | x \ \in \ Y, \quad y \ \ in \ Y\}$

- A section graph is a clique if all the nodes in the subgraph are pairwise adjacent ($\rightarrow$ dense block in matrix)

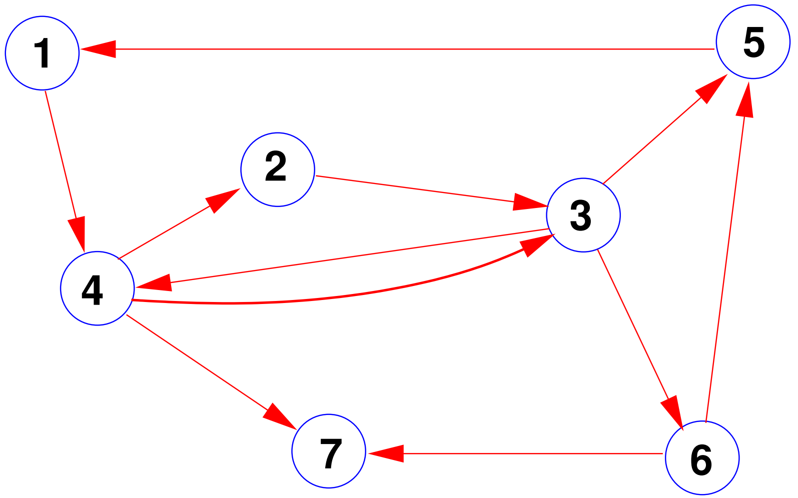
- A path is a sequence of vertices $w_0, w_1, \ldots, w_k$ such that $(w_i, w_{i+1}) \ \in E$ for $i = 0, \ldots, k-1$.

- The length of the path $w_0, w_1, \ldots, w_k$ is k (# of edges in the path)

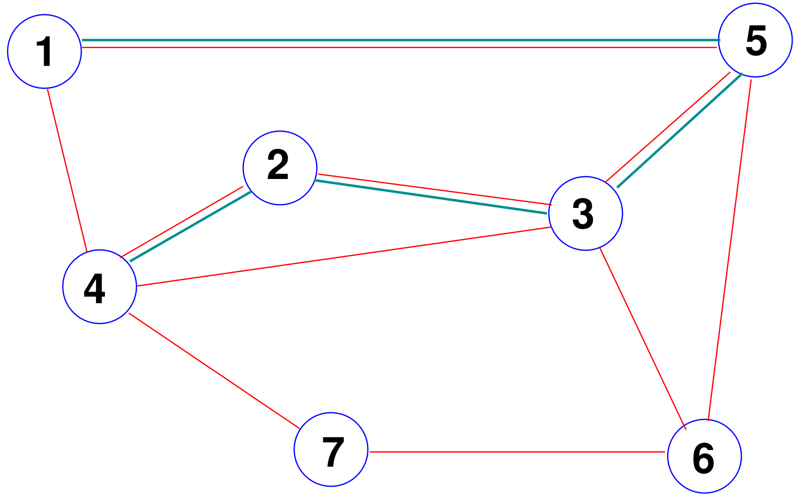
- A cycle is a closed path, i.e., a path with $w_k = w_0$.

- A graph is acyclic if it has no cycles.

Find cycles in this graph:



A path in an indirected graph



- A path $w_0, \ldots, w_k$ is simple if the vertices $w_0, \ldots, w_k$ are distinct (except that we may have $w_0 = w_k$ for cycles).
- An undirected graph is connected if there is path from every vertex to every other vertex.
- A digraph with the same property is said to be strongly connected

- The undirected (or symmetrized) form of a digraph = undirected graph obtained by removing the directions of all edges

- A directed graph whose undirected form is connected is said to be weakly connected or connected.

- Tree = a graph whose undirected form, i.e., symmetrized form, is acyclic & connected – Forest = a collection of trees

GRAPH MODELS FOR SPARSE MATRICES

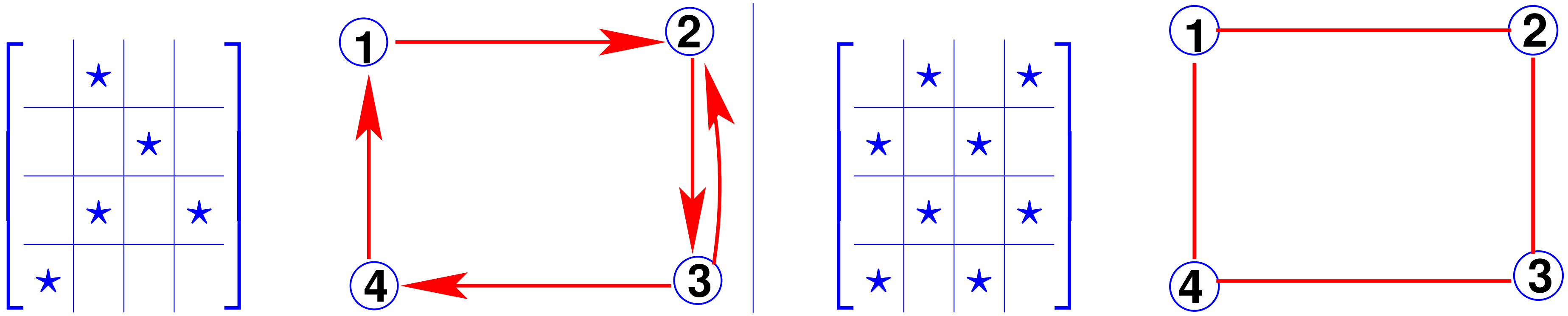
Graph Representations of Sparse Matrices. Recall:

Adjacency Graph $G = (V, E)$ of an $n \times n$ matrix A :

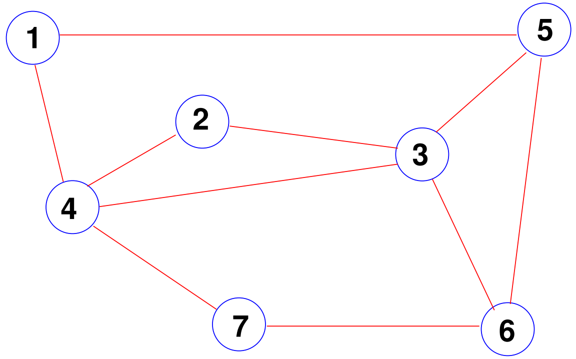
$$V = \{1, 2, ..., N\} \qquad E = \{(i, j) | a_{ij} \neq 0\}$$

- G == undirected if A has a symmetric pattern

Example:



✍ Show the matrix pattern for the graph on the right and give an interpretation of the path v_4, v_2, v_3, v_5, v_1 on the matrix



➤ A separator is a set Y of vertices such that the graph G_{X-Y} is disconnected.

Example: $Y = \{v_3, v_4, v_5\}$ is a separator in the above figure

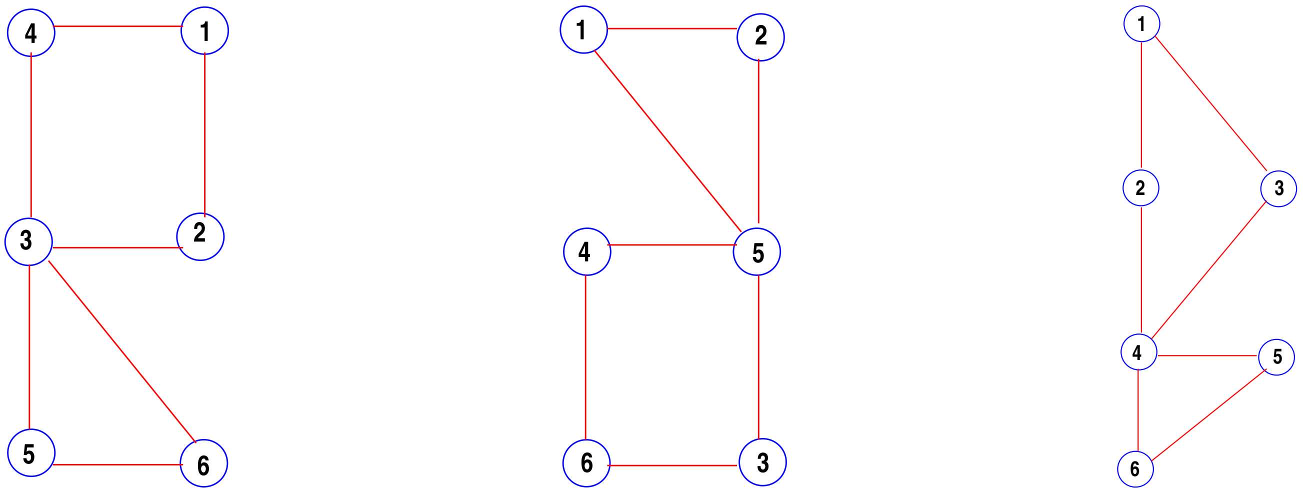
Example: Adjacency graph of:

$$A = \begin{bmatrix} & \star & & \star & & \\ \star & & \star & & & \\ & \star & & \star & \star & \star \\ \star & & \star & & & \\ & & \star & & & \star \\ & & \star & & \star & \end{bmatrix}.$$

Example: For any adjacency matrix A, what is the graph of A^2? [interpret in terms of paths in the graph of A]

- Two graphs are isomorphic is there is a mapping between the vertices of the two graphs that preserves adjacency.

✍ Are the following 3 graphs isomorphic? If yes find the mappings between them.



- Graphs are identical – labels are different
- Determinig graph isomorphism is a hard problem

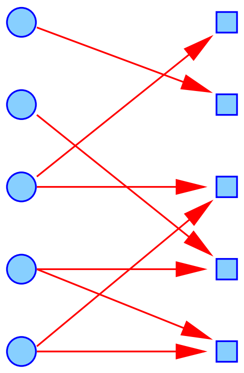
Bipartite graph representation

- Rows and columns are (both) represented by vertices;

- Relations only between rows and columns: Row i is connected to column j if $a_{ij} \neq 0$

Example:

$$\begin{bmatrix} & \star & & & \\ & & & \star & \\ \star & & \star & & \\ & & & \star & \star \\ & & \star & & \star \end{bmatrix}$$



- Bipartite models used only for specific cases [e.g. rectangular matrices, ...] - By default we use the standard definition of graphs.

Interpretation of graphs of matrices

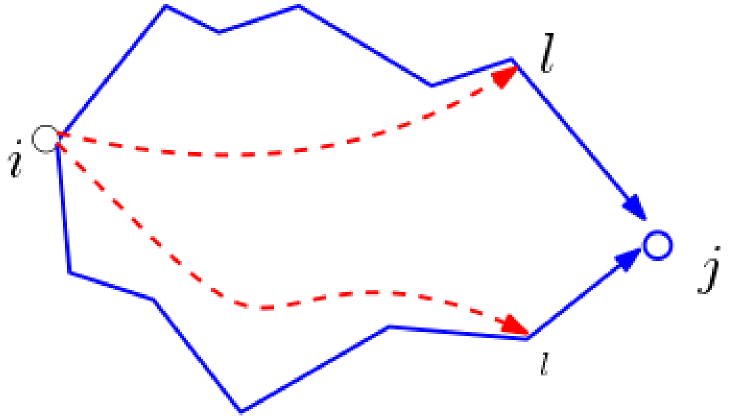
- What is the graph of $A + B$ (for two $n \times n$ matrices)?
- What is the graph of A^T ?
- What is the graph of $A.B$?

Paths in graphs

✍ What is the graph of A^k?

Theorem Let A be the adjacency matrix of a graph $G = (V, E)$. Then for $k \geq 0$ and vertices u and v of G, the number of paths of length k starting at u and ending at v is equal to $(A^k)_{u,v}$.

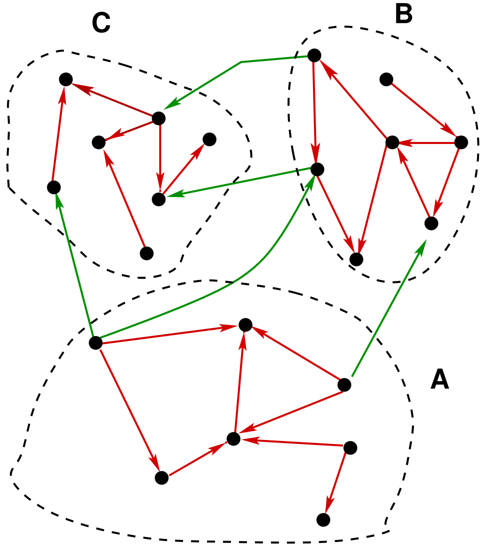
Proof: Proof is by induction. ■

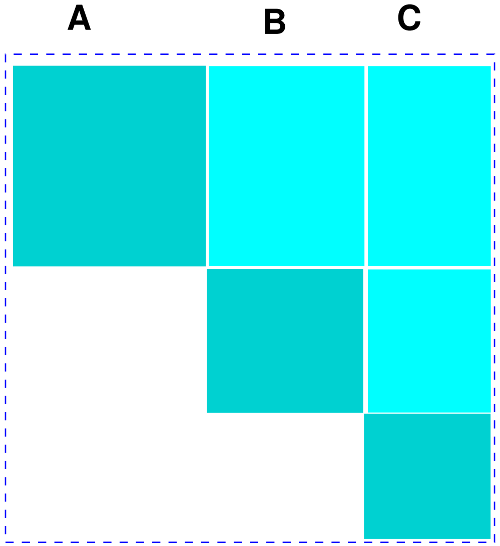


If $C = BA$ then $c_{ij} = \Sigma_l b_{il} a_{lj}$. Take $B = A^{k-1}$ and use induction. Any path of length k is formed as a path of length $k-1$ to some node l completed by an edge from l to j. Because a_{lj} is one for that last edge, c_{ij} is just the sum of all possible paths of length k from i to j

- Recall (definition): A matrix is *reducible* if it can be permuted into a block upper triangular matrix.

- Note: A matrix is reducible iff its adjacency graph is not (strongly) connected, i.e., iff it has more than one connected component.





- No edges from C to A or B. No edges from B to A.

Theorem: Perron-Frobenius An irreducible, nonnegative $n \times n$ matrix A has a real, positive eigenvalue λ_1 such that:
(i) λ_1 is a simple eigenvalue of A;
(ii) λ_1 admits a positive eigenvector u_1 ; and
(iii)$|\lambda_i| \leq \lambda_1$ for all other eigenvalues λ_i where $i > 1$.

- The spectral radius is equal to the eigenvalue λ_1

- Definition : a graph is d regular if each vertex has the same degree d.

Proposition: The spectral radius of a d regular graph is equal to d.

Proof: The vector e of all ones is an eigenvector of A associated with the eigenvalue $\lambda = d$. In addition this eigenvalue is the largest possible (consider the infinity norm of A). Therefore e is the Perron-Frobenius vector u_1. ■

Application: Markov Chains

- Read about Markov Chains in Sect. 10.9 of:

  https://www-users.cs.umn.edu/~saad/eig_book_2ndEd.pdf

- Let $\pi \equiv$ row vector of stationary probabilities
- Then π satisfies the equation $\rightarrow$ $\pi P = \pi$

- P is the probabilty transition matrix and it is 'stochastic':

A matrix P is said to be *stochastic* if :
(i) $p_{ij} \geq 0$ for all i, j
(ii) $\sum_{j=1}^{n} p_{ij} = 1$ for $i = 1, \cdots, n$
(iii) No column of P is a zero column.

- Spectral radius is ≤ 1

✍ Why?

- Assume P is irreducible. Then:

- Perron Frobenius $\rightarrow \rho(P) = 1$ is an eigenvalue and associated eigenvector has positive entries.

- Probabilities are obtained by scaling π by its sum.

- Example: One of the 2 models used for page rank.

Example: A college Fraternity has 50 students at various stages of college (Freshman, Sophomore, Junior, Senior). There are 6 potential stages for the following year: Freshman, Sophomore, Junior, Senior, graduated, or left-without degree. Following table gives probability of transitions from one stage to next

To From	Fr	So.	Ju.	Sr.	Grad	lwd
Fr.	.2	0	0	0	0	0
So.	.6	.1	0	0	0	0
Ju.	0	.7	.1	0	0	0
Sr.	0	0	.8	.1	0	0
Grad	0	0	0	.75	1	0
lwd	.2	.2	.1	.15	0	1

✍ What is P? Assume initial population is $x_0 = [10, 16, 12, 12, 0, 0]$ and do a follow the population for a few years. What is the probability that a student will graduate? What is the probability that s/he leaves without a degree?

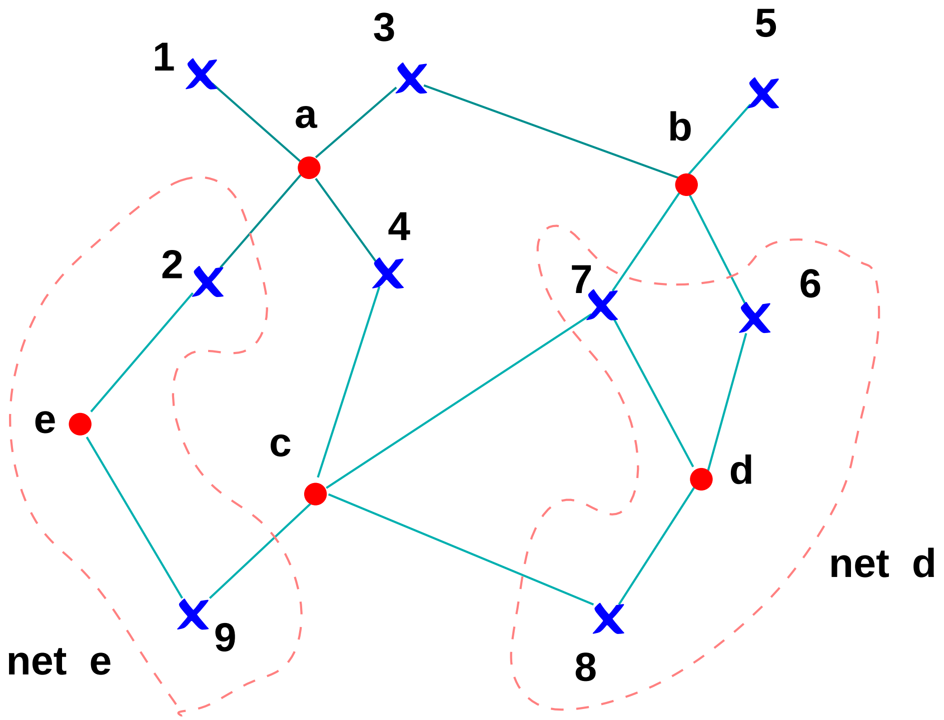
A few words on hypergraphs

- Hypergraphs are very general.. Ideas borrowed from VLSI work
- Main motivation: to better represent communication volumes when partitioning a graph. Standard models face many limitations
- Hypergraphs can better express complex graph partitioning problems and provide better solutions.
- Example: completely nonsymmetric patterns ...
- .. Even rectangular matrices. Best illustration: Hypergraphs are ideal for text data

Example: $V = \{1, \ldots, 9\}$ and $E = \{a, \ldots, e\}$ with

$a = \{1, 2, 3, 4\}$, $b = \{3, 5, 6, 7\}$, $c = \{4, 7, 8, 9\}$,

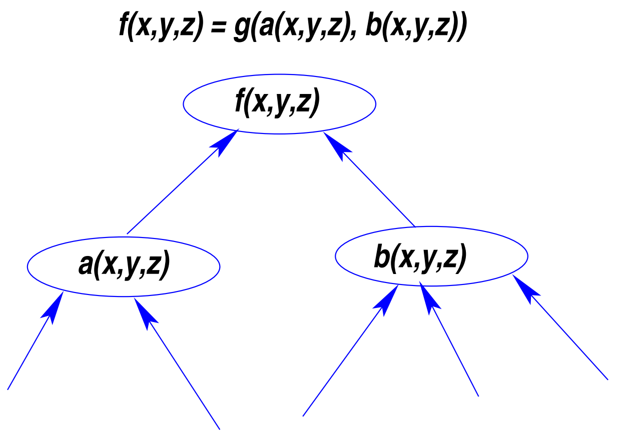
$d = \{6, 7, 8\}$, and $e = \{2, 9\}$



Boolean matrix:

$$A = \begin{array}{c} \begin{array}{ccccccccc} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 \end{array} \\ \left[\begin{array}{ccccccccc} 1 & 1 & 1 & 1 & & & & & \\ & & 1 & & 1 & 1 & 1 & & \\ & & & 1 & & & 1 & 1 & 1 \\ & & & & & 1 & 1 & 1 & \\ & 1 & & & & & & & 1 \end{array}\right] \begin{array}{c} a \\ b \\ c \\ d \\ e \end{array} \end{array}$$

A few words on computational graphs



- Computational graphs: graphs where nodes represent computations whose evaluation depend on other (incoming) nodes.

- Consider the following expression:

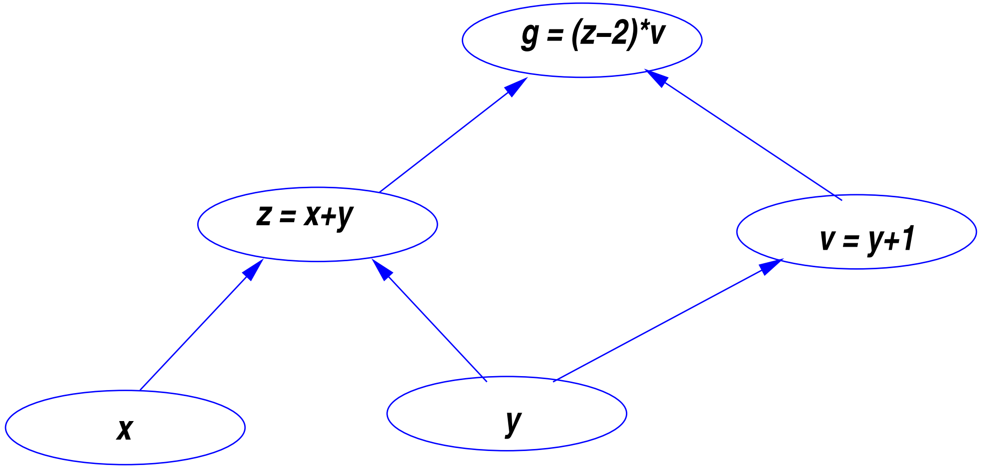
$$g(x, y) = (x + y - 2) * (y + 1)$$

We can decompose this as
$$\begin{cases} z = x + y \\ v = y + 1 \\ g = (z - 2) * v \end{cases}$$

- Computational graph $\rightarrow$
- Given x, y we want:

(a) Evaluate the nodes and

(b) derivatives w.r.t x, y



(a) is trivial - just follow the graph up - starting from the leaves (that contain x and y)

(b): Use the chain rule – here shown for x only using previous setting

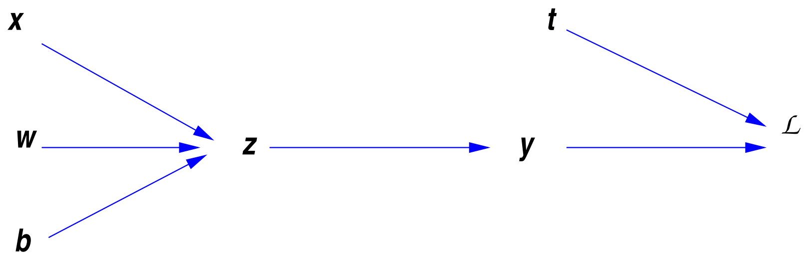
$$\frac{\partial g}{\partial x} = \frac{\partial g}{\partial a}\frac{da}{dx} + \frac{\partial g}{\partial b}\frac{db}{dx}$$

✍ For the above example compute values and derivatives at all nodes when $x = -1, y = 2.$

Back-Propagation

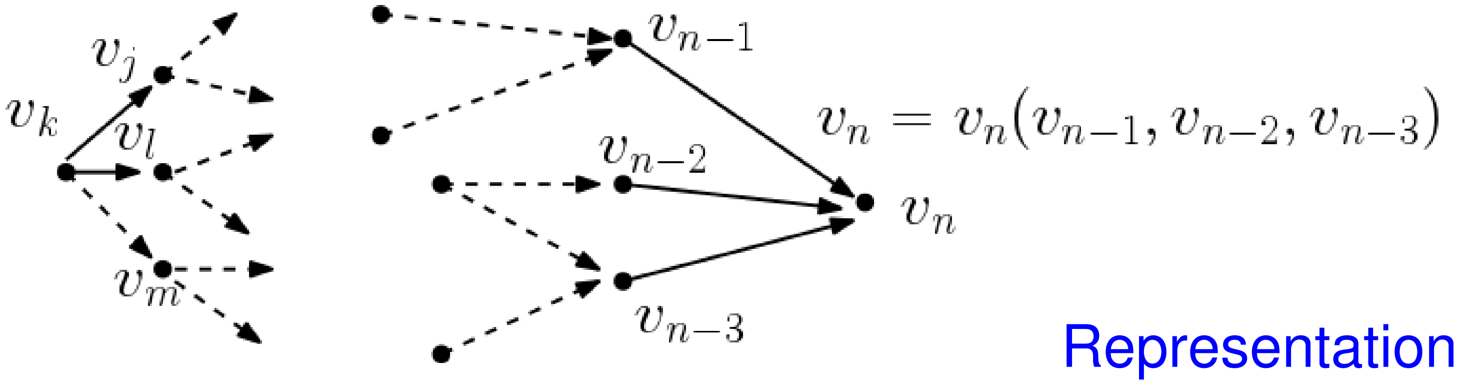
- Often we want to compute the gradient of the function at the root, once the nodes have been evaluated
- The derivatives can be calculated by going backward (or down the tree)
- Here is a very simple example from Neural Networks

$$\begin{cases} L = \frac{1}{2}(y-t)^2 \\ y = \sigma(z) \\ z = wx + b \end{cases}$$



- Note that t (desired output) and x (input) are constant.

Back-Propagation: General computational graphs



Representation: *a DAG*

- Last node (v_n) is the target function. Let us rename it f.
- Nodes $v_i, i = 1, \cdots, e$ with indegree 0 are the variables
- Want to compute $\partial f/\partial v_1, \partial f/\partial v_2, \cdots, \partial f/\partial v_e$
- Use the chain rule.

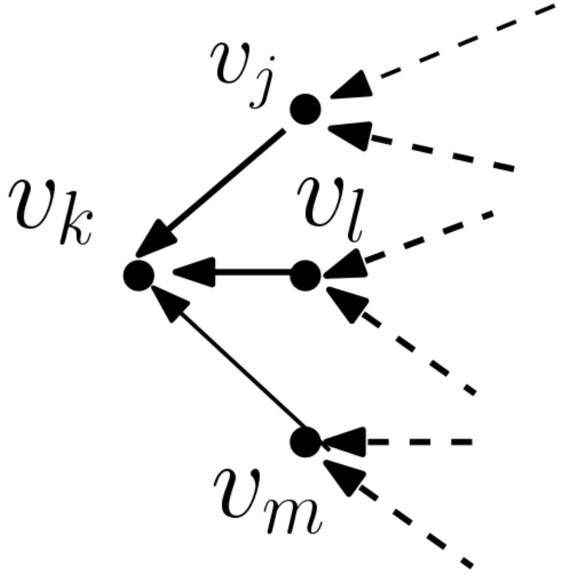
For $v_k(v_j, v_l, v_m) \longrightarrow$

$$\frac{\partial f}{\partial v_k} = \frac{\partial f}{\partial v_j}\frac{\partial v_j}{\partial v_k} + \frac{\partial f}{\partial v_l}\frac{\partial v_l}{\partial v_k} + \frac{\partial f}{\partial v_m}\frac{\partial v_m}{\partial v_k}$$

- Let $\delta_k = \frac{\partial f}{\partial v_k}$ (called 'errors'). Then

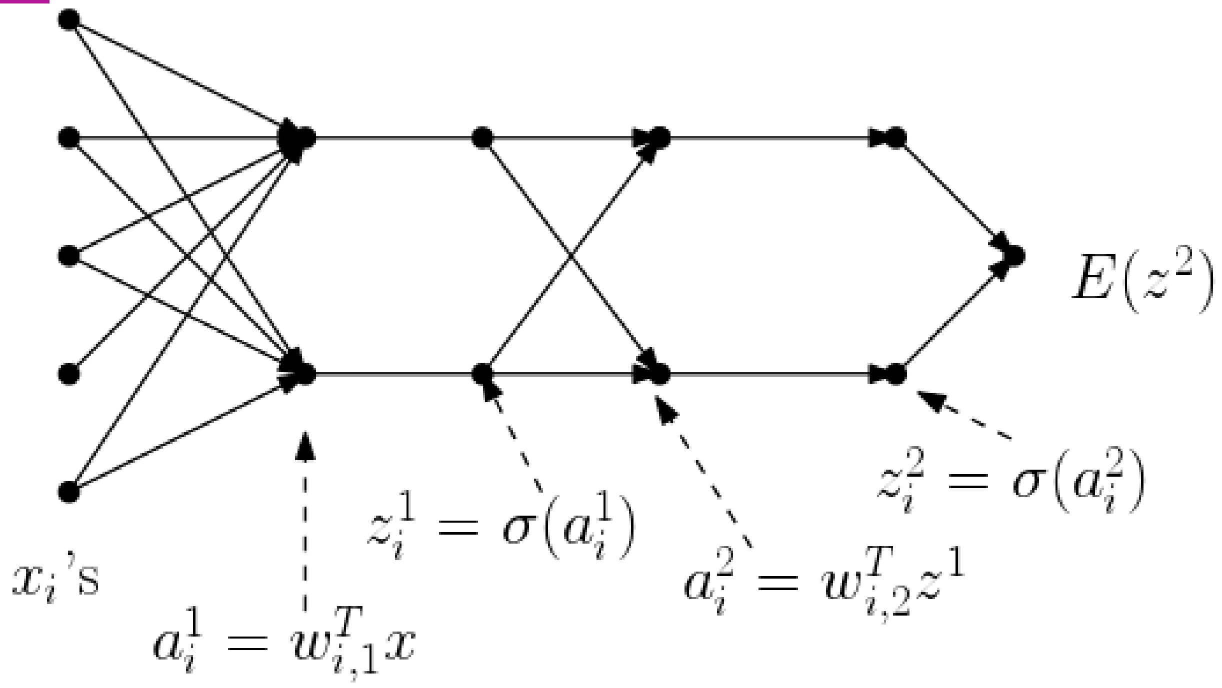
$$\delta_k = \delta_j \frac{\partial v_j}{\partial v_k} + \delta_l \frac{\partial v_l}{\partial v_k} + \delta_m \frac{\partial v_m}{\partial v_k}$$

- To compute the δ_k's once the v_j's have been computed (in a 'forward' propagation) – proceed backward.
- $\delta_j, \delta_l, \delta_m$ available and $\partial v_i / \partial v_k$ computable. Nore $\delta_n \equiv 1$.



- However: cannot just do this in any order. Must follow a topological order in order to obey dependencies.

Example:



GRAPH CENTRALITY

Centrality in graphs

- Goal: measure importance of a node, edge, subgraph, .. in a graph
- Many measures introduced over the years
- Early Work: Freeman '77 [introduced 3 measures] – based on 'paths in graph'
- Many different ways of defininf centrality! We will just see a few

Degree centrality: (simplest) 'Nodes with high degree are important'

(note: scaling $n-1$ is unimportant)

$$C_D(v) = \frac{\deg(v)}{n-1}$$

Closeness centrality: 'Nodes that are close to many other nodes are important'

$$C_C(v) = \frac{n-1}{\sum_{w \neq v} d(v,w)}$$

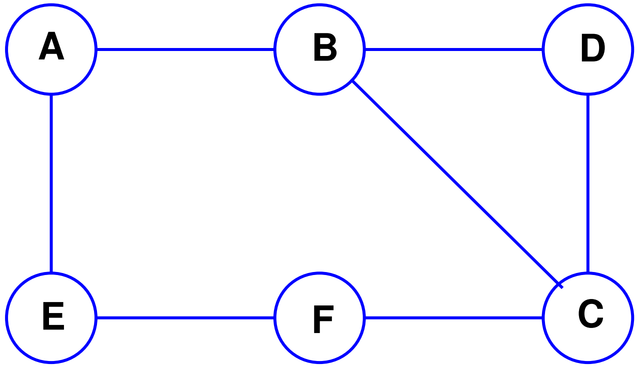
Betweenness centrality: (Freeman '77)

$$C_B(v) = \sum_{u \neq v, w \neq v} \frac{\sigma_{uw}(v)}{\sigma_{uw}}$$

- σ_{uw} = total # shortest paths from u to w
- $\sigma_{uw}(v)$ = total # shortest paths from u to w passing through v

➤ 'Nodes that are on many shortest paths are important'

Example: Find $C_D(v)$; $C_C(v)$; $C_B(v)$ when $v = C$



(u,w)	$\sigma_{uw}(v)$	σ_{uw}	/	(u,w)	$\sigma_{uw}(v)$	σ_{uw}	/
(A,B)	0	1	0	(B,E)	0	1	0
(A,D)	0	1	0	(B,F)	1	1	1
(A,E)	0	1	0	(D,E)	1	2	.5
(A,F)	0	1	0	(D,F)	1	1	1
(B,D)	0	1	0	(E,F)	0	1	0

- $C_D(v) = 3/5 = 0.6$;
- $$C_C(v) = 5/[d_{CA} + d_{CB} + d_{CD} + d_{CE} + d_{CF}]$$
  $$= 5/[2 + 1 + 1 + 2 + 1] = 5/7$$
- $C_B(v) = 2.5$ (add all ratios in table)

Redo this for $v = B$

Eigenvector centrality:

- Supppose we have n nodes v_j, $j = 1, \cdots, n$– each with a measure of importance ('prestige') p_j
- Principle: prestige of i depends on that of its neighbors.
- Prestige x_i = multiple of sum of prestiges of neighbors pointing to it

$$\lambda x_i = \sum_{j \in \mathcal{N}(i)} x_j = \sum_{j=1}^{n} a_{ji} x_j$$

- x_i = component of eigenvector associated with λ.
- Perron Frobenius theorem at play again: take largest eigenvalue $\rightarrow$ x_i's nonnegative

Page-rank

➤ Can be viewed as a variant of Eigenvector centrality

Main point: A page is important if it is pointed to by other important pages.

➤ Importance of your page (its PageRank) is determined by summing the page ranks of all pages which point to it. [→ same as EV centrality]

➤ Weighting: If a page points to several other pages, then the weighting should be distributed proportionally.

➤ Imagine many tokens doing a random walk on this graph:

- (δ/n) chance to follow one of the n links on a page,
- $(1-\delta)$ chance to jump to a random page.
- What's the chance a token will land on each page?

Page-Rank - definitions

If T_1, ..., T_n point to page T_i then

$$\rho(T_i) \;=\; 1 - \delta + \delta \left[\frac{\rho(T_1)}{|T_1|} + \frac{\rho(T_2)}{|T_2|} + \cdots \frac{\rho(T_n)}{|T_n|} \right]$$

- $|T_j|$ = count of links going out of Page T_i. So the 'vote' $\rho(T_j)$ is spread evenly among $|T_j|$ links.
- Sum of all PageRanks == 1: $\Sigma_T \rho(T) = 1$
- δ is a 'damping' parameter close to 1 – e.g. 0.85
- Defines a (possibly huge) Hyperlink matrix H

$$h_{ij} = \begin{cases} \frac{1}{|T_i|} & \text{if } \; i \text{ points to } j \\ 0 & \text{otherwise} \end{cases}$$

4 Nodes

A points to B and D

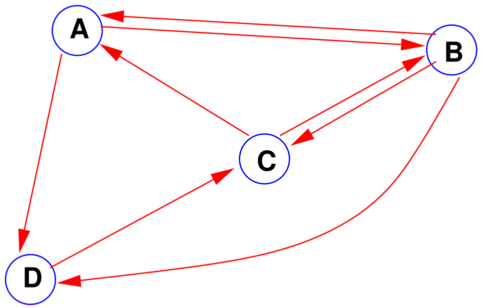
B points to A, C, and D

C points to A and B

D points to C

1) What is the H matrix?

2) the graph?



	A	B	C	D
A		$1/2$		$1/2$
B	$1/3$		$1/3$	$1/3$
C	$1/2$	$1/2$		
D			1	

- Row- sums of H are = 1.
- Sum of all PageRanks will be one: $$\sum_{\text{All-Pages}_A} \rho(A) = 1.$$
- H is a stochastic matrix [actually it is forced to be by changing zero rows]

Algorithm (PageRank)

1. Select initial row vector v ($v \geq 0$)
2. For i=1:maxitr
3 $\quad v := (1-\delta)e^T + \delta v H$
4. end

✍ Do a few steps of this algorithm for previous example with $\delta = 0.85$.

➤ This is a row iteration..

$$\boxed{v} = \boxed{(1-\delta)e^T} + \boxed{v} \cdot \boxed{\delta H}$$

A few properties:

- v will remain ≥ 0. [combines non-negative vectors]
- More general iteration is of the form

$$v := v\underbrace{[(1-\delta)E + \delta H]}_{G} \quad \textbf{with} \quad E = ez^T$$

where z is a probability vector $e^T z = 1$ [Ex. $z = \frac{1}{n}e$]

- A variant of the power method.
- e is a right-eigenvector of G associated with $\lambda = 1$. We are interested in the left eigenvector.

Kleinberg's Hubs and Authorities

➤ Idea is to put order into the web by ranking pages by their degree of Authority or "Hubness".

➤ An Authority is a page pointed to by many important pages.

• Authority Weight = sum of Hub Weights from In-Links.

➤ A Hub is a page that points to many important pages:

• Hub Weight = sum of Authority Weights from Out-Links.

➤ Source:

http://www.cs.cornell.edu/home/kleinber/auth.pdf

Computation of Hubs and Authorities

- Simplify computation by forcing sum of squares of weights to be 1.
- $\mathrm{Auth}_j = x_j = \sum_{i:(i,j)\in \mathrm{Edges}} \mathrm{Hub}_i$.
- $\mathrm{Hub}_i = y_i = \sum_{j:(i,j)\in \mathrm{Edges}} \mathrm{Auth}_j$.
- Let A = Adjacency matrix: $a_{ij} = 1$ if $(i,j) \in \mathrm{Edges}$.
- $\mathrm{y} = A\mathrm{x},\ \mathrm{x} = A^T\mathrm{y}$.
- Iterate . . . to leading eigenvectors of A^TA & AA^T.
- Answer: Leading Singular Vectors!

GRAPH LAPLACEANS AND THEIR APPLICATIONS

Graph Laplaceans - Definition

➤ "Laplace-type" matrices associated with general undirected graphs – useful in many applications

➤ Given a graph $G = (V, E)$ define

- A matrix W of weights w_{ij} for each edge
- Assume $w_{ij} \geq 0,$, $w_{ii} = 0$, and $w_{ij} = w_{ji} \; \forall(i, j)$
- The diagonal matrix $D = diag(d_i)$ with $d_i = \sum_{j \neq i} w_{ij}$

➤ Corresponding *graph Laplacean* of G is: $L = D - W$

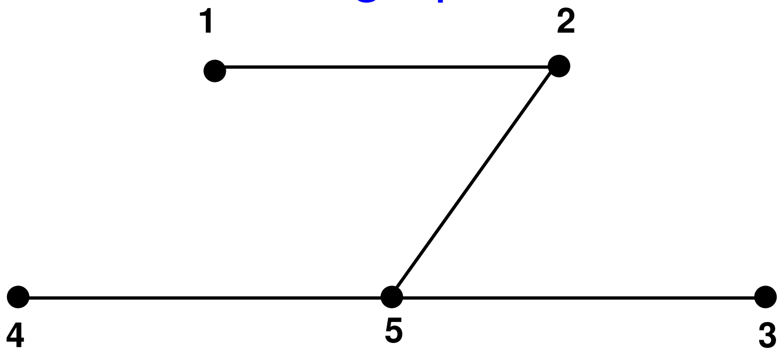
➤ Gershgorin's theorem $\rightarrow L$ is positive semidefinite.

- Simplest case:

$$w_{ij} = \begin{cases} 1 \text{ if } (i,j) \in E \& i \neq j \\ 0 \quad \text{else} \end{cases} \qquad D = \text{diag}\left[d_i = \sum_{j \neq i} w_{ij} \right]$$

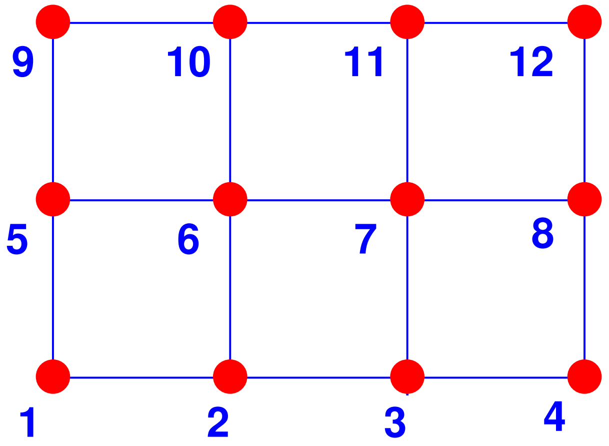
Example:

Consider the graph



$$L = \begin{bmatrix} 1 & -1 & 0 & 0 & 0 \\ -1 & 2 & 0 & 0 & -1 \\ 0 & 0 & 1 & 0 & -1 \\ 0 & 0 & 0 & 1 & -1 \\ 0 & -1 & -1 & -1 & 3 \end{bmatrix}$$

✍ Define the graph Laplacean for the graph associated with the simple mesh shown next. [use the simple weights of 0 or 1]. What is the difference with the discretization of the Laplace operator for case when mesh is the same as this graph?



Proposition:

(i) L is symmetric semi-positive definite.

(ii) L is singular with $\mathbb{1}$ as a null vector.

(iii) If G is connected, then $\mathbf{Null}(L) = \mathbf{span}\{\mathbb{1}\}$

(iv) If G has $k > 1$ connected components $G_1, G_2, \cdots, G_k$, then the nullity of L is k and $\mathbf{Null}(L)$ is spanned by the vectors $z^{(j)}$, $j = 1, \cdots, k$ defined by:

$$(z^{(j)})_i = \begin{cases} 1 & \text{if } i \in G_j \\ 0 & \text{if not.} \end{cases}$$

Proof: (i) and (ii) seen earlier and are trivial. (iii) Clearly $u = \mathbb{1}$ is a null vector for L. The vector $D^{-1/2}u$ is an eigenvector for the matrix $D^{-1/2}LD^{-1/2} = I - D^{-1/2}WD^{-1/2}$ associated with the smallest eigenvalue. It is also an eigenvector for $D^{-1/2}WD^{-1/2}$ associated with the largest eigenvalue. By the Perron Frobenius theorem this is a simple eigenvalue... (iv) Can be proved from the fact that L can be written as a direct sum of the Laplacian matrices for $G_1, \cdots, G_k$. ■

A few properties of graph Laplaceans

Define: oriented incidence matrix H: (1)First orient the edges $i \sim j$ into $i \to j$ or $j \to i$. (2) Rows of H indexed by vertices of G. Columns indexed by edges. (3) For each (i, j) in E, define the corresponding column in H as $\sqrt{w(i,j)}(e_i - e_j)$.

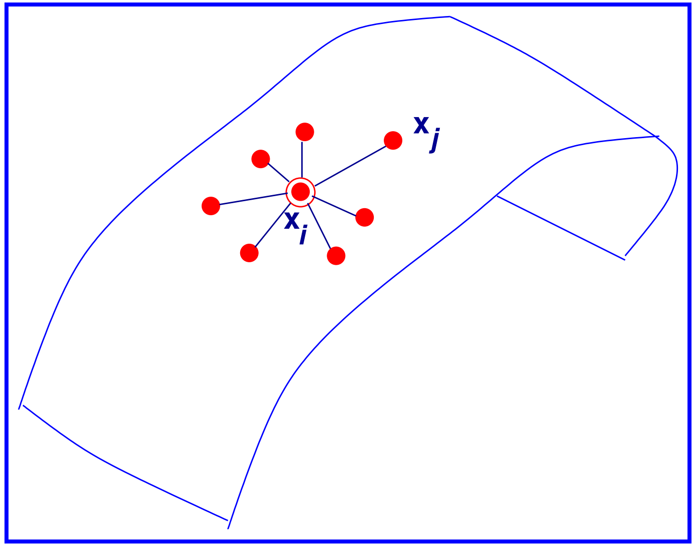
Example: In previous example (4 p. back) orient $i \to j$ so that $j > i$ [lower triangular matrix representation]. Then matrix H is:

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 \\ -1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & -1 & -1 & -1 \end{bmatrix}$$

Property 1 $\boxed{L = HH^T}$

✍ Re-prove part (iv) of previous proposition by using this property.

A few properties of graph Laplaceans



Strong relation between $x^T L x$ and local distances between entries of x

➤ Let L = any matrix s.t. $L = D - W$, with $D = diag(d_i)$ and

$$w_{ij} \geq 0, \qquad d_i = \sum_{j \neq i} w_{ij}$$

Property 2: for any $x \in \mathbb{R}^n$:

$$x^\top L x = \frac{1}{2} \sum_{i,j} w_{ij} |x_i - x_j|^2$$

Property 3: (generalization) for any $Y \in \mathbb{R}^{d\times n}$:

$$\text{Tr}\,[YLY^\top] = \frac{1}{2}\sum_{i,j} w_{ij}\|y_i - y_j\|^2$$

➤ Note: y_j = j-th column of Y. Usually $d < n$. Each column can represent a data sample.

Property 4: For the particular $L = I - \frac{1}{n}\mathbb{1}\,\mathbb{1}^\top$

$$XLX^\top = \bar{X}\bar{X}^\top == n \times \text{Covariance matrix}$$

Property 5: L is singular and admits the null vector

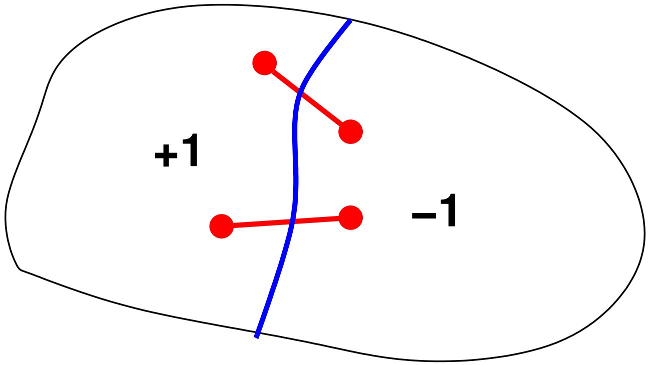
$\mathbb{1}$ =`ones(n,1)`

Property 6: (Graph partitioning) Consider situation when $w_{ij} \in \{0, 1\}$. If x is a vector of signs (± 1) then

$$x^\top L x = 4 \times (\text{'number of edge cuts'})$$

edge-cut = pair (i, j) with $x_i \neq x_j$

➤ Consequence: Can be used to partition graphs



- Would like to minimize (Lx, x) subject to $x \in \{-1, 1\}^n$ and $e^T x = 0$ [balanced sets]

- WIl solve a relaxed form of this problem

✍ What if we replace x by a vector of ones (representing one partition) and zeros (representing the other)?

✍ Let x be any vector and $y = x + \alpha \mathbb{1}$ and L a graph Laplacean. Compare (Lx, x) with (Ly, y).

- Consider any symmetric (real) matrix A with eigenvalues $\lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_n$ and eigenvectors $u_1, \cdots, u_n$

- Recall that:
(Min reached for $x = u_1$)

$$\min_{x \in \mathbb{R}^n} \frac{(Ax, x)}{(x, x)} = \lambda_1$$

- In addition:
(Min reached for $x = u_2$)

$$\min_{x \perp u_1} \frac{(Ax, x)}{(x, x)} = \lambda_2$$

- For a graph Laplacean $u_1 = \mathbb{1}$ = vector of all ones and

- ...vector u_2 is called the Fiedler vector. It solves a relaxed form of the problem -

$$\min_{x \in \{-1,1\}^n;\ \mathbb{1}^T x = 0} \frac{(Lx, x)}{(x, x)} \quad \rightarrow \quad \min_{x \in \mathbb{R}^n;\ \mathbb{1}^T x = 0} \frac{(Lx, x)}{(x, x)}$$

- Define $v = u_2$ then $lab = sign(v - med(v))$

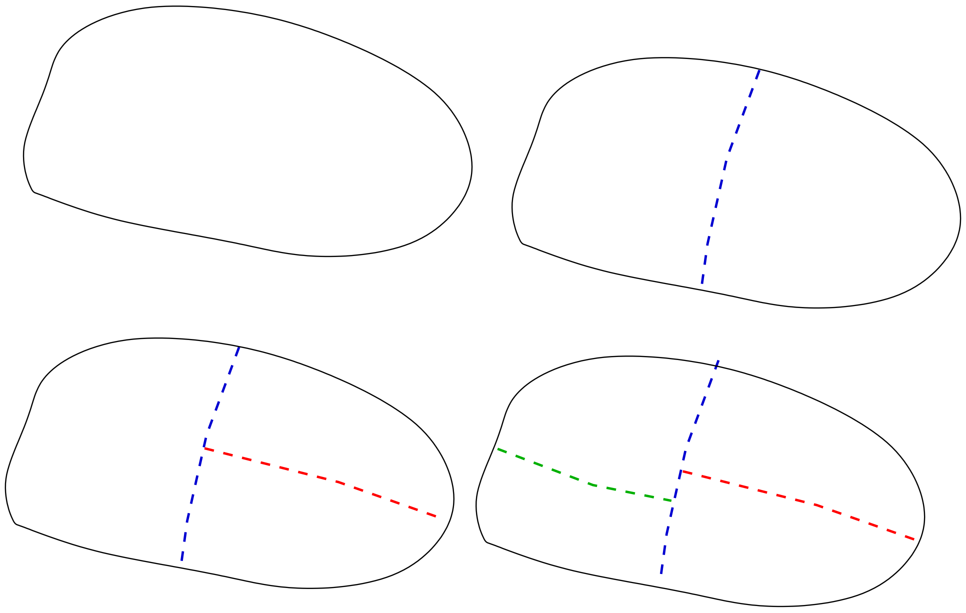
Recursive Spectral Bisection

1 Form graph Laplacean

2 Partition graph in 2 based on Fielder vector

3 Partition largest subgraph in two recursively ...

4 ... Until the desired number of partitions is reached

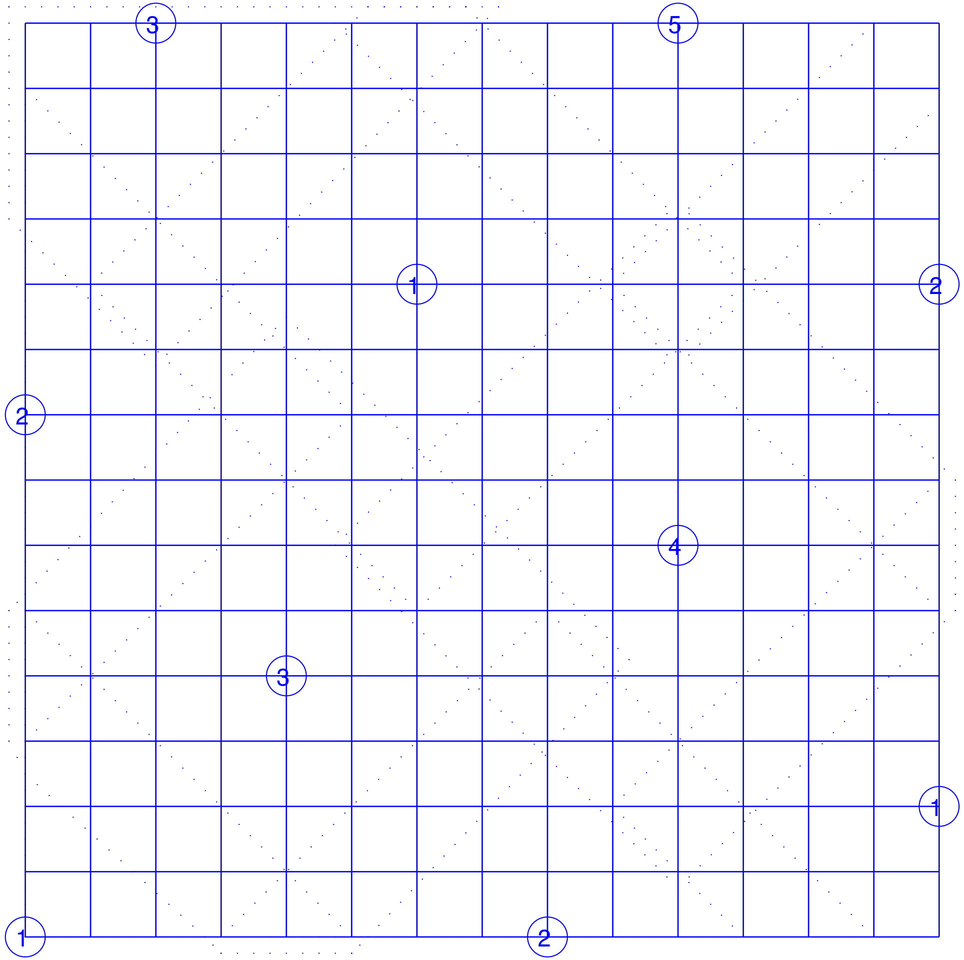


Three approaches to graph partitioning:

1. Spectral methods - Just seen + add Recursive Spectral Bisection.
2. Geometric techniques. Coordinates are required. [Houstis & Rice et al., Miller, Vavasis, Teng et al.]
3. Graph Theory techniques – multilevel,... [use graph, but no coordinates]
   - Currently best known technique is Metis (multi-level algorithm)
   - Simplest idea: Recursive Graph Bisection; Nested dissection (George & Liu, 1980; Liu 1992]
   - Advantages: simplicity – no coordinates required

Example of a graph theory approach

- Level Set Expansion Algorithm
- Given: p nodes 'uniformly' spread in the graph (roughly same distance from one another).
- Method: Perform a level-set traversal (BFS) from each node simultaneously.
- Best described for an example on a 15×15 five – point Finite Difference grid.
- See [Goehring-YS '94, See Cai-YS '95]
- Approach also known under the name 'bubble' algorithm and implemented in some packages [Party, DibaP]


3
5
1
2
2
4
3
1
1
2