

CSI 445/660 – Part 6
(Centrality Measures for Networks)

- 1 L. Freeman, “Centrality in Social Networks: Conceptual Clarification”, *Social Networks*, Vol. 1, 1978/1979, pp. 215–239.
- 2 S. Wasserman and K. Faust, *Social Network Analysis: Methods and Applications*, Cambridge University Press, New York, NY, 1994.
- 3 M. E. J. Newman, *Networks: An Introduction*, Oxford University Press, New York, NY, 2010.
- 4 Wikipedia entry on Centrality Measures:
<https://en.wikipedia.org/wiki/Centrality>

Some Pioneers on the Topic



- Alex Bavelas (1913–1993) (??)
- Received Ph.D. from MIT (1948) in Psychology.
- Dorwin Cartwright was a member of his Ph.D. thesis committee.
- Taught at MIT, Stanford and the University of Victoria (Canada).



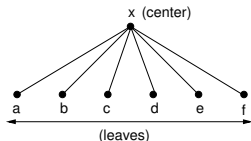
- Harold Leavitt (1922–2007)
- Received Ph.D. from MIT.
- Authored an influential text (“Managerial Psychology”) in 1958.
- Taught at Carnegie Mellon and Stanford.

Centrality:

- Represents a “measure of importance”.
 - Usually for nodes.
 - Some measures can also be defined for edges (or subgraphs, in general).
- Idea proposed by Alex Bavelas during the late 1940's.
- Further work by Harold Leavitt (Stanford) and Sidney Smith (MIT) led to **qualitative** measures.
- **Quantitative** measures came years later. (Many such measures have been proposed.)

Point Centrality – A Qualitative Measure

Example:



- The **center** node is “structurally more important” than the other nodes.

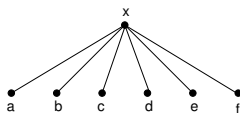
Reasons for the importance of the center node:

- The center node has the maximum possible degree.
- It lies on the shortest path (“geodesic”) between any pair of other nodes (leaves).
- It is the closest node to each leaf.
- It is in the “thick of things” with respect to any communication in the network.

Degree Centrality – A Quantitative Measure

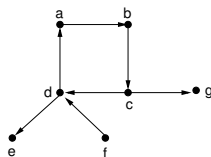
- For an **undirected** graph, the **degree** of a node is the number of edges incident on that node.
- For a **directed** graph, both **indegree** (i.e., the number of incoming edges) and **outdegree** (i.e., the number of outgoing edges) must be considered.

Example 1:



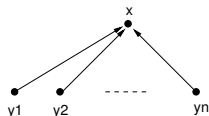
- Degree of $x = 6$.
- For all other nodes, degree = 1.

Example 2:



- Indegree of $b = 1$.
- Outdegree of $d = 2$.

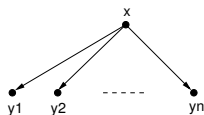
When does a large indegree imply higher importance?



- Consider the Twitter network.
- Think of x as a **celebrity** and the other nodes as followers of x .
- For a different context, think of each node in the directed graph as a web page.
- Each of the nodes y_1, y_2, \dots, y_n has a link to x .
- The larger the value of n , the higher is the “importance” of x (a crude definition of **page rank**).

Degree Centrality (continued)

When does a large outdegree imply higher importance?



- Consider the hierarchy in an organization.
- Think of x as the manager of y_1, y_2, \dots, y_n .
- Large outdegree may mean more “power”.

Undirected graphs:

- High degree nodes are called **hubs** (e.g. airlines).
- High degree may also also represent higher **risk**.

Example: In disease propagation, a high degree node is more likely to get infected compared to a low degree node.

Normalized Degree

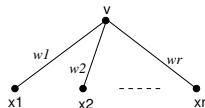
Definition: The **normalized degree** of a node x is given by

$$\text{Normalized Degree of } x = \frac{\text{Degree of } x}{\text{Maximum possible degree}}$$

- Useful in comparing degree centralities of nodes between two networks.

Example: A node with a degree of 5 in a network with 10 nodes may be relatively more important than a node with a degree of 5 in a network with a million nodes.

Weighted Degree Centrality (Strength):



- Weighted degree (or strength) of $v = w_1 + w_2 + \dots + w_r$.

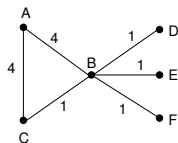
Degree Centrality (continued)

Assuming an **adjacency list** representation

- for an undirected graph $G(V, E)$, the degree (or weighted degree) of all nodes can be computed in **linear** time (i.e., in time $O(|V| + |E|)$) and
- for a directed graph $G(V, E)$, the indegree or outdegree (or their weighted versions) of all nodes can be computed in **linear** time.

Combining degree and strength: ([Opsahl et al. 2009])

Motivating Example:



- A and B have the same strength.
- However, B seems more central than A.

Proposed Measure by Opsahl et al.:

- Let d and s be the degree and strength of a node v respectively.
- Let α be a parameter satisfying the condition $0 \leq \alpha \leq 1$.
- The combined measure for node $v = d^\alpha \times s^{1-\alpha}$.
- When $\alpha = 1$, the combined measure is the **degree**.
- When $\alpha = 0$, the combined measure is the **strength**.
- A suitable value of α must be chosen for each context.

Farness and Closeness Centralities

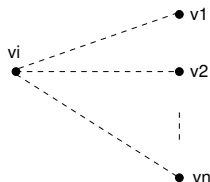
Assumptions:

- Undirected graphs. (Extension to directed graphs is straightforward.)
- Connected graphs.
- No edge weights. (Extension to weighted graphs is also straightforward.)

Notation:

- Nodes of the graph are denoted by v_1, v_2, \dots, v_n . The set of all nodes is denoted by V .
- For any pair of nodes v_i and v_j , d_{ij} denotes the number of edges in a shortest path between v_i and v_j .

Farness and Closeness Centralities (continued)



- A schematic showing shortest paths between node v_i and the other nodes of an undirected graph.

Definition: The **farness centrality** f_i of node v_i is given by

$$\begin{aligned} f_i &= \text{Sum of the distances between } v_i \text{ and the other nodes} \\ &= \sum_{v_j \in V - \{v_i\}} d_{ij} \end{aligned}$$

Definition: The **closeness centrality** (or **nearness centrality**) η_i of node v_i is given by $\eta_i = 1/f_i$.

Note: If a node x has a larger closeness centrality value compared to a node y , then x is more central than y .

Farness and Closeness Centralities (continued)

Example 1:

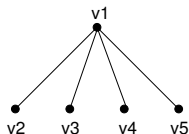


- $f_1 = 1 + 2 + 3 = 6$. So, $\eta_1 = 1/6$.
- $f_2 = 1 + 1 + 2 = 4$. So, $\eta_2 = 1/4$.
- $f_3 = 2 + 1 + 2 = 4$. So, $\eta_3 = 1/4$.
- $f_4 = 3 + 2 + 1 = 6$. So, $\eta_4 = 1/6$.

So, in the above example, nodes v_2 and v_3 are more central than nodes v_1 and v_4 .

Farness and Closeness Centralities (continued)

Example 2:



- $f_1 = 4$. So, $\eta_1 = 1/4$.
- For every other node, the farness centrality value = 7; so the closeness centrality value = $1/7$.
- Thus, v_1 is more central than the other nodes.

Remarks:

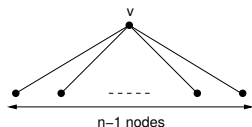
- For any graph with n nodes, the **farness centrality** of each node is **at least** $n - 1$.

Reason: Each of the other $n - 1$ nodes must be at a distance of at least 1.

Farness and Closeness Centralities (continued)

Remarks (continued):

- Since the farness centrality of each node is at least $n - 1$, the **closeness centrality** of any node must be **at most** $1/(n - 1)$.



- For the star graph on the left, the closeness centrality of the center node v is exactly $1/(n - 1)$.
- If G is an n -clique, then the closeness centrality of each node of G is $1/(n - 1)$.

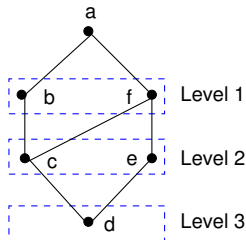
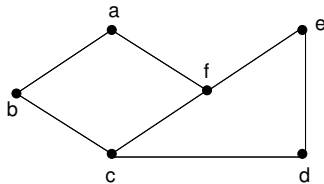
An Algorithm for Computing Farness and Closeness

Assumptions: The given undirected graph is **connected** and does **not** have edge weights.

Computing Farness (or closeness) Centrality (Idea):

- A Breadth-First-Search (BFS) starting at a node v_i will find shortest paths to all the other nodes.

Example:



An Algorithm for Farness ... (continued)

Let $G(V, E)$ denote the given graph.

- Recall that the time for doing a BFS on $G = O(|V| + |E|)$.
- So, farness (or closeness) centrality for any node of G can be computed in $O(|V| + |E|)$ time.
- By carrying out a BFS from each node, the time to compute farness (or closeness) centrality for **all** nodes of G
 $= O(|V|(|V| + |E|))$.
- The time is $O(|V|^3)$ for **dense** graphs (where $|E| = \Omega(|V|^2)$)
and $O(|V|^2)$ for **sparse** graphs (where $|E| = O(|V|)$).

Eccentricity Measure

- Recall that **farness centrality** of a node v_i is given by

$$f_i = \sum_{v_j \in V - \{v_i\}} d_{ij}$$

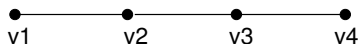
- The **eccentricity** μ_i of node v_i is defined by replacing the **summation** operator (\sum) by the **maximization** operator; that is,

$$\mu_i = \max_{v_j \in V - \{v_i\}} \{d_{ij}\}$$

- This measure was studied by two graph theorists (Gert Sabidussi and Seifollah L. Hakimi).
- **Interpretation:** If μ_i denotes the eccentricity of node v_i , then every other node is within a distance of **at most** μ_i from v_i .
- If the eccentricity of node x is less than that of y , then x is more central than y .

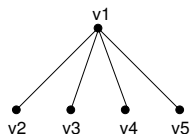
Examples: Eccentricity Computation

Example 1:



- $\mu_1 = \max\{1, 2, 3\} = 3.$
- $\mu_2 = \max\{1, 1, 2\} = 2.$
- $\mu_3 = \max\{2, 1, 1\} = 2.$
- $\mu_4 = \max\{3, 2, 1\} = 3.$

Example 2:

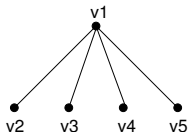


- $\mu_1 = 1.$
- For every other node, eccentricity = 2.

Eccentricity – Additional Definitions

Definition: A node v of a graph which has the smallest eccentricity among all the nodes is called a **center** of the graph.

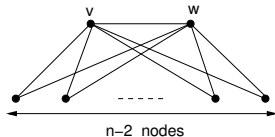
Example:



- The center of this graph is v_1 .
(The eccentricity of $v_1 = 1$.)

Note: A graph may have two or more centers.

Example:



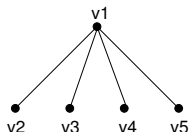
- Both v and w are centers of this graph.
(Their eccentricities are $= 1$.)
- If G is clique on n nodes, then every node of G is a center.

Eccentricity – Additional Definitions (continued)

Definition: The smallest eccentricity value among all the nodes is called the **radius** of the graph.

Note: The value of the radius is the eccentricity of a center.

Example:



- The radius of this graph is 1 (since v_1 is the center of this graph and the eccentricity of $v_1 = 1$.)

Facts:

- The **largest eccentricity value** is the **diameter** of the graph.
- For any graph, the diameter is at most twice the radius.
(Students should try to prove this result.)

An Algorithm for Computing Eccentricity

Let $G(V, E)$ denote the given graph.

- **Recall:** By carrying out a BFS from node v_i , the shortest path distances between v_i and all the other nodes can be found in $O(|V| + |E|)$ time.
- So, the eccentricity of any node of G can be computed in $O(|V| + |E|)$ time.
- By repeating the BFS for each node, the time to compute eccentricity for **all** nodes of $G = O(|V|(|V| + |E|))$.
- So, the radius, diameter and all centers of G can be found in $O(|V|(|V| + |E|))$ time.

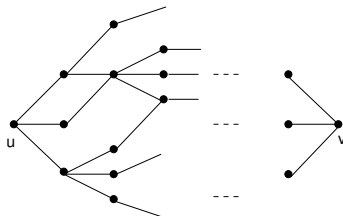
Random Walk Based Centrality (Brief Discussion)

Ref: [Noh & Rieger 2004]

Motivation:

- Definitions of centrality measures (such as **closeness** centrality) assume that “information” propagates along shortest paths.
- This may not be appropriate for certain other types of propagation. For example, propagation of diseases is a **probabilistic** phenomenon.

Idea of Random Walk Distance in a Graph:



Random Walk ... (Brief Discussion)

Random Walk Algorithm – Outline:

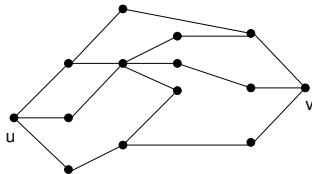
- Suppose we want to find the random walk distance from u to v .
- **Initialize:** Current Node = u and No. of steps = 0.
- **Repeat**
 - 1 Randomly choose a neighbor x of the Current Node.
 - 2 No. of steps = No. of steps + 1.
 - 3 Set Current Node = x .

Until Current Node = v .

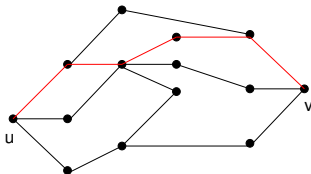
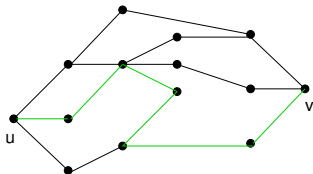
Note: In Step 1 of the loop, if the Current Node has degree d , probability of choosing any neighbor is $1/d$.

Examples of Random Walks

A graph for carrying out a random walk:



Examples of random walks on the above graph:



Random Walk ... (Brief Discussion)

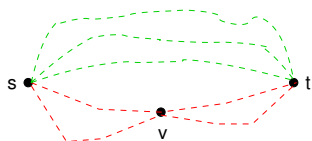
Definition: The **random walk distance** (or **hitting time**) from u to v is the expected number of steps used in a random walk that starts at u and ends at v .

- One can define farness/closeness centrality measures based on random walk distances.
- **Weakness:** Even for undirected graphs, the random walk distances are **not symmetric**; that is, the random walk distance from u to v may **not** be the same as the random walk distance from v to u .

Betweenness Centrality (for Nodes)

- Measures the importance of a node using the **number of shortest paths** in which the node appears.
- Suggested by Bavelas; however, he didn't formalize it.
- The measure was developed by Linton Freeman and J. M. Anthonisse.

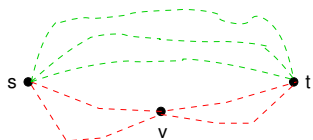
Consider a node v and two other nodes s and t .



- Each shortest path between s and t shown in **green doesn't** pass through node v .
- Each shortest path between s and t shown in **red** passes through node v .

Betweenness Centrality ... (continued)

Notation: Any shortest path between nodes s and t will be called an s - t **shortest path**.



- Let σ_{st} denote the number of all s - t shortest paths.
- Let $\sigma_{st}(v)$ denote the number of all s - t shortest paths that pass through node v .

Consider the ratio $\frac{\sigma_{st}(v)}{\sigma_{st}}$:

- This gives the fraction of s - t shortest paths passing through v .
- The larger the ratio, the more important v is with respect to the pair of nodes s and t .
- To properly measure the importance of a node v , we need to consider all pairs of nodes (not involving v).

Betweenness Centrality ... (continued)

Definition: The **betweenness centrality** of a node v , denoted by $\beta(v)$, is defined by

$$\beta(v) = \sum_{\substack{s, t \\ s \neq v, t \neq v}} \left[\frac{\sigma_{st}(v)}{\sigma_{st}} \right]$$

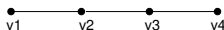
Interpreting the above formula: Suppose we want to compute $\beta(v)$ for some node v . The formula suggests the following steps.

- Set $\beta(v) = 0$.
- For each pair of nodes s and t such that $s \neq v$ and $t \neq v$,
 - 1 Compute σ_{st} and $\sigma_{st}(v)$.
 - 2 Set $\beta(v) = \beta(v) + \sigma_{st}(v)/\sigma_{st}$.
- Output $\beta(v)$.

Note: For two nodes x and y , if $\beta(x) > \beta(y)$, then x is more central than y .

Examples: Betweenness Computation

Example 1:



Note: Here, there is **only one** path between any pair of nodes. (So, that path is also the shortest path.)

Consider the computation of $\beta(v_2)$ first.

- The s - t pairs to be considered are: (v_1, v_3) , (v_1, v_4) and (v_3, v_4) .
- For the pair (v_1, v_3) :
 - The number of shortest paths between v_1 and v_3 is 1; thus, $\sigma_{v_1, v_3} = 1$.
 - The (only) path between v_1 and v_3 passes through v_2 ; thus, $\sigma_{v_1, v_3}(v_2) = 1$.
 - So, the ratio $\sigma_{v_1, v_3}(v_2)/\sigma_{v_1, v_3} = 1$.
- In a similar manner, for the pair (v_1, v_4) , the ratio $\sigma_{v_1, v_4}(v_2)/\sigma_{v_1, v_4} = 1$.

Examples: Betweenness Computation (continued)

Computation of $\beta(v_2)$ continued:



- For the pair (v_3, v_4) :
 - The number of shortest paths between v_3 and v_4 is 1; thus, $\sigma_{v_3, v_4} = 1$.
 - The (only) path between v_3 and v_4 **does not** pass through v_2 ; thus, $\sigma_{v_3, v_4}(v_2) = 0$.
 - So, the ratio $\sigma_{v_3, v_4}(v_2)/\sigma_{v_3, v_4} = 0$.

Therefore,

$$\begin{aligned}\beta(v_2) &= 1 \quad (\text{for the pair } (v_1, v_3)) \\ &\quad + 1 \quad (\text{for the pair } (v_1, v_4)) \\ &\quad + 0 \quad (\text{for the pair } (v_3, v_4)) \\ &= 2.\end{aligned}$$

Note: In a similar manner, $\beta(v_3) = 2$.

Examples: Betweenness Computation

Example 1: (continued)



Now, consider the computation of $\beta(v_1)$.

- The s - t pairs to be considered are: (v_2, v_3) , (v_2, v_4) and (v_3, v_4) .
- For each of these pairs, the number of shortest paths is 1.
- v_1 **doesn't** lie on any of these shortest paths.
- Thus, for each pair, the fraction of shortest paths that pass through $v_1 = 0$.
- Therefore, $\beta(v_1) = 0$.

Note: In a similar manner, $\beta(v_4) = 0$.

Examples: Betweenness Computation (continued)

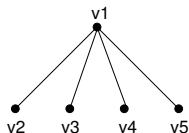
Summary for Example 1:



- $\beta(v_1) = \beta(v_4) = 0.$

- $\beta(v_2) = \beta(v_3) = 2.$

Example 2:

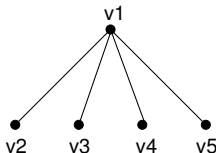


- Here also, there is **only one** path between any pair of nodes.

- Consider the computation of $\beta(v_1)$ first.

Examples: Betweenness Computation (continued)

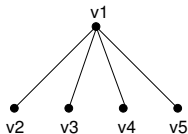
Computation of $\beta(v_1)$ (continued):



- We must consider all pairs of nodes from $\{v_2, v_3, v_4, v_5\}$.
- The number of such pairs = 6. (They are: (v_2, v_3) , (v_2, v_4) , (v_2, v_5) , (v_3, v_4) , (v_3, v_5) , (v_4, v_5) .)
- For each pair, there is only one path between them and the path passes through v_1 .
- Therefore, the ratio contributed by each pair is 1.
- Since there are 6 pairs, $\beta(v_1) = 6$.

Examples: Betweenness Computation (continued)

Computation of $\beta(v_2)$:



- We must consider all pairs of nodes from $\{v_1, v_3, v_4, v_5\}$.
- The number of such pairs = 6.
- For each pair, there is only one path between them and the path **doesn't** pass through v_2 .
- Therefore, $\beta(v_2) = 0$.

Notes:

- In a similar manner, $\beta(v_3) = \beta(v_4) = \beta(v_5) = 0$.
- Summary for Example 2:
 - $\beta(v_1) = 6$ and
 - $\beta(v_i) = 0$, for $i = 2, 3, 4, 5$.

Computing Betweenness: Major Steps

Requirement: Given graph $G(V, E)$, compute $\beta(v)$ for each node $v \in V$.

Note: A straightforward algorithm and its running time will be discussed.

Major steps: Consider one node (say, v) at a time.

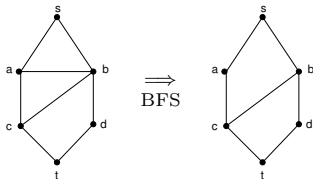
- For a given pair of nodes s and t , where $s \neq v$ and $t \neq v$, compute the following values:
 - 1 The no. of s - t shortest paths (i.e., the value of σ_{st}).
 - 2 The no. of s - t shortest paths passing through v (i.e., the value of $\sigma_{st}(v)$).

Major Step 1: Computing the **number** of shortest paths between a pair of nodes s and t .

Method: Breadth-First-Search (BFS) from node s followed by a **top down** computation procedure.

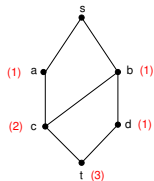
Example for Major Step 1

(a) Carrying out a BFS:



Note: The edge $\{a, b\}$ does not play any role in the computation of σ_{st} .

(b) Computing the value of σ_{st} :



- For each node, the value shown in **red** gives the number of shortest paths from s to that node.
- These numbers are computed through a top-down computation (to be explained in class).
- In this example, $\sigma_{st} = 3$.

Running Time of Major Step 1

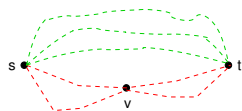
Assume that $G(V, E)$ is the given graph.

- For each node s , the time for BFS starting at s is $O(|V| + |E|)$.
- For the chosen s , computing the σ_{st} value for for all other nodes t can also be done in $O(|V| + |E|)$ time.
- So, the computation time for each node s is $O(|V| + |E|)$.
- Since there are $|V|$ nodes, the time for Major Step 1 is $O(|V|(|V| + |E|))$.
- The running time is $O(|V|^3)$ for **dense** graphs and $O(|V|^2)$ for **sparse** graphs.

Idea for Major Step 2

Goal of Major Step 2: Given an (s, t) pair and a node v (which is neither s nor t), compute $\sigma_{st}(v)$, the number of s - t shortest paths **passing through** v .

Idea:



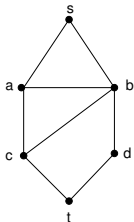
- Compute the the number of of s - t shortest paths that **don't** pass through v (i.e., the number of **green** paths). Let $\gamma_{st}(v)$ denote this value.
- Then, $\sigma_{st}(v) = \sigma_{st} - \gamma_{st}(v)$.

How can we compute $\gamma_{st}(v)$?

- If we delete node v from the graph, all the **green** paths remain in the graph.
- So, $\gamma_{st}(v)$ can be computed by considering the graph G_v obtained by deleting v and all the edges incident on v .

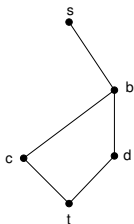
Example for Major Step 2

Graph $G(V, E)$:



Goal: Compute the number of s - t shortest paths that **don't** pass through a .

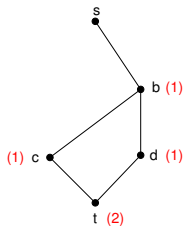
Graph G_a :



- The number of s - t shortest paths in G that **don't** pass through a is the number of s - t shortest paths in G_a .
- The required computation is exactly that of Major Step 1, except that it must be done for graph G_v .

Example for Major Step 2 (continued)

Graph G_a :



- For each node, the number in **red** gives the number of shortest paths between s and the node in G_a .
- From the figure, $\gamma_{st}(a) = 2$.
- Since $\sigma_{st} = 3$, $\sigma_{st}(a) = 3 - 2 = 1$.

Running Time of Major Step 2

As before, assume that $G(V, E)$ is the given graph.

- For each node $v \in V$, the following steps are carried out.
 - Construct graph G_v . (This can be done in $O(|V| + |E|)$ time.)
 - For each node s of G_v , computing the number of s - t shortest paths for all other nodes can be done in $O(|V| + |E|)$ time.
 - Since there are $|V| - 1$ nodes G_v , the time for Major Step 2 for each node v is $O(|V|(|V| + |E|))$.
- So, over all the nodes $v \in V$, the running time for Major Step 2 is $O(|V|^2(|V| + |E|))$.
- The running time is $O(|V|^4)$ for **dense** graphs and $O(|V|^3)$ for **sparse** graphs.

Algorithm for betweenness computation: See Handout 6.1.

Review of Concepts Related to Matrices

Example:

$$\begin{bmatrix} 7 & -8 & -14 \\ 2 & 4 & -3 \end{bmatrix}$$

- A matrix with 2 rows and 3 columns.
- Also referred to as a 2×3 matrix.
- This matrix is **rectangular**.
- In a **square** matrix, the number of rows equals the number of columns.

Matrix addition or subtraction:

- Two matrices can be added (or subtracted) only if they have the same number of rows and columns.
- The result is obtained by adding (or subtracting) the **corresponding** entries.

Example:

$$\begin{bmatrix} 7 & -8 & -14 \\ 2 & 4 & -3 \end{bmatrix} + \begin{bmatrix} 3 & 2 & -1 \\ 1 & 2 & -4 \end{bmatrix} = \begin{bmatrix} 10 & -6 & -15 \\ 3 & 6 & -7 \end{bmatrix}$$

Review of Matrices (continued)

Matrix multiplication:

- Given matrices P and Q , the product PQ is defined only when **the number of columns of P = the number of rows of Q** .
- If P is an $m \times n$ matrix and Q is an $n \times r$ matrix, the product PQ is an $m \times r$ matrix.

Example: (The procedure will be explained in class.)

$$\begin{bmatrix} 1 & 0 & 3 \\ 2 & 1 & 0 \end{bmatrix} * \begin{bmatrix} 3 & 2 \\ 1 & 2 \\ 2 & 0 \end{bmatrix} = \begin{bmatrix} 9 & 2 \\ 7 & 6 \end{bmatrix}$$

Main diagonal of a square matrix:

$$\begin{bmatrix} \mathbf{3} & 4 & 5 & 0 \\ 2 & \mathbf{4} & 3 & 7 \\ 3 & 1 & \mathbf{9} & 4 \\ 7 & 9 & 2 & \mathbf{8} \end{bmatrix}$$

- A 4×4 (square) matrix.
- The **main diagonal** entries are in **blue**.

Review of Matrices (continued)

Identity Matrix: For any positive integer n , the $n \times n$ **identity** matrix, denoted by I_n , has 1's along the main diagonal and 0's in every other position.

Example: Identity matrix I_4 .

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Property: For any $n \times n$ matrix A , $I_n A = A I_n = A$.

Example:

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} 3 & 2 & 2 & 3 \\ 1 & 2 & 3 & -2 \\ 2 & 1 & 1 & 4 \\ 7 & 5 & 4 & 1 \end{bmatrix} = \begin{bmatrix} 3 & 2 & 2 & 3 \\ 1 & 2 & 3 & -2 \\ 2 & 1 & 1 & 4 \\ 7 & 5 & 4 & 1 \end{bmatrix}$$

Review of Matrices (continued)

Example – Multiplying a matrix by a number (scalar):

$$3 \times \begin{bmatrix} 1 & 2 \\ -5 & 4 \end{bmatrix} = \begin{bmatrix} 3 & 6 \\ -15 & 12 \end{bmatrix}.$$

Determinant of a square matrix:

- For a 2×2 matrix $A = \begin{bmatrix} a & b \\ c & d \end{bmatrix}$, the value of the **determinant** is given by

$$\text{Det}(A) = ad - bc.$$

Example: Suppose $A = \begin{bmatrix} 2 & -1 \\ 3 & -2 \end{bmatrix}$. Then

$$\text{Det}(A) = (-2 \times 2) - (3 \times -1) = -1.$$

Review of Matrices (continued)

Example: Computing the determinant of a 3×3 matrix.

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

$\text{Det}(B)$ can be computed as follows.

$$\begin{aligned} \text{Det}(B) &= 3 \times \text{Det} \begin{bmatrix} -1 & 2 \\ 2 & 1 \end{bmatrix} - 1 \times \text{Det} \begin{bmatrix} 2 & 2 \\ 0 & 1 \end{bmatrix} \\ &\quad + 0 \times \text{Det} \begin{bmatrix} 2 & -1 \\ 0 & 2 \end{bmatrix} \\ &= 3(-5) - 1(2) + 0 \\ &= -17. \end{aligned}$$

Review of Matrices (continued)

Eigenvalues of a square matrix: If A is an $n \times n$ matrix, the **eigenvalues** of A are the solutions to the **characteristic** equation

$$\text{Det}(A - \lambda I_n) = 0$$

where λ is a variable.

Example: Suppose

$$A = \begin{bmatrix} 1 & 3 \\ 2 & 2 \end{bmatrix}.$$

Note that

$$\lambda I_2 = \begin{bmatrix} \lambda & 0 \\ 0 & \lambda \end{bmatrix}.$$

So,

$$A - \lambda I_2 = \begin{bmatrix} 1 - \lambda & 3 \\ 2 & 2 - \lambda \end{bmatrix}.$$

Hence,

$$\text{Det}(A - \lambda I_2) = (2 - \lambda)(1 - \lambda) - 6 = \lambda^2 - 3\lambda - 4.$$

Review of Matrices (continued)

Example (continued):

So, the characteristic equation for A is given by

$$\lambda^2 - 3\lambda - 4 = 0$$

- The solutions to this equation are: $\lambda = 4$ and $\lambda = -1$.
- These are the **eigenvalues** of the matrix A .
- The largest eigenvalue (in this case, $\lambda = 4$) is called the **principal** eigenvalue.
- For each eigenvalue λ of A , there is a 2×1 matrix (vector) \mathbf{x} such that $A\mathbf{x} = \lambda\mathbf{x}$. Such a vector is called an **eigenvector** of the eigenvalue λ . (This vector can be computed efficiently.)
- For the above matrix A , for the principal eigenvalue $\lambda = 4$, an eigenvector \mathbf{x} is given by

$$\mathbf{x} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}.$$

Review of Matrices (continued)

Matrices and linear equations:

Example:

$$\begin{aligned}3x_1 - 2x_2 + x_3 &= 7 \\x_1 - 3x_2 - 2x_3 &= 0 \\2x_1 + 3x_2 + 3x_3 &= 5\end{aligned}$$

Suppose

$$A = \begin{bmatrix} 3 & -2 & 1 \\ 1 & -3 & -2 \\ 2 & 3 & 3 \end{bmatrix} \quad X = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \quad \text{and} \quad B = \begin{bmatrix} 7 \\ 0 \\ 5 \end{bmatrix}.$$

Then the above set of equations can be written as

$$\begin{bmatrix} 3 & -2 & 1 \\ 1 & -3 & -2 \\ 2 & 3 & 3 \end{bmatrix} * \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 7 \\ 0 \\ 5 \end{bmatrix} \quad \text{or} \quad AX = B.$$