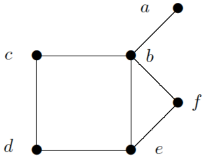
Universitat Pompeu Fabra Barcelona

Lecture 4 · Graph Theory III

Networks, Crowds and Markets

Warm Up

Given the following graph:



Determine:

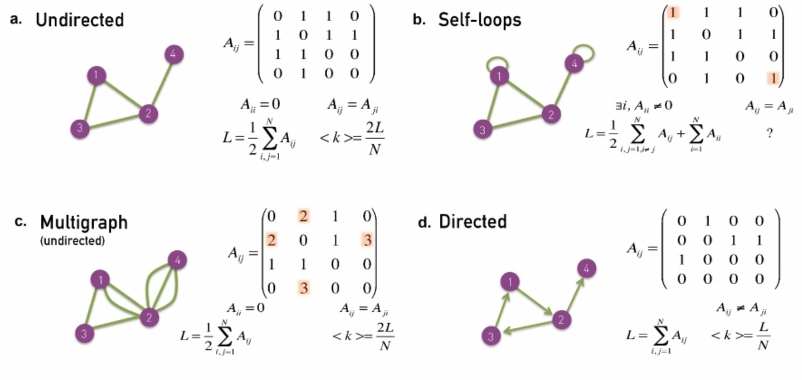
a) the average degree of the graph.

b) the degree distribution.

c) its adjacency matrix.

Average Degree

In a simple graph, the vector $A\mathbf{1}$ gives the degrees. Thus $\frac{1}{N}\mathbf{1}^\top A\mathbf{1} = \frac{1}{N}\sum_{i,j=1}^{N} A_{ij}$ gives the average degree. How about other cases?



Barabási denotes the average degree by $\langle k \rangle$. We do not follow this convention.

Today's Lecture

1. Distances in a graph
   1.1 Path / Shortest path / Distance
   1.2 Breadth First Search
   1.3 Diameter / Local Diameter
   1.4 Eccentricity
2. Connectivity
   2.1 Definition
   2.2 Bridge
3. Trees
4. Regular Graphs

Distances in graphs

Path

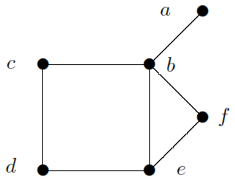
Definition

A **walk** in an undirected graph $G = (V, E)$ is a sequence of vertices $(v_0, \ldots, v_\ell)$ s.t. each consecutive pair $v_{i-1}v_i$ is an edge in E.

A **path** in an undirected graph $G = (V, E)$ is a sequence of distinct vertices $(v_0, \ldots, v_\ell)$ s.t. each consecutive pair $v_{i-1}v_i$ is an edge in E.

Length: the number of edges in the sequence, here ℓ.

In a weighted graph we could take the sum of the weights of the edges on the path; path weight.

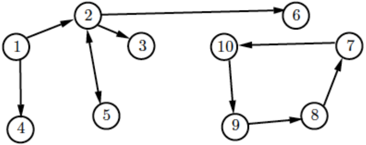


Examples: $p_1 = abfe$, $p_2 = cdef$, $p_3 = defba$

Directed Path

Definition

A **directed path** in a directed graph G is a finite sequence of edges $e_i = (x_i, y_i)$ such that for each $i \geq 1$, $x_{i+1} = y_i$.

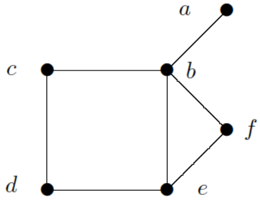


Examples:
$p_1 = \overline{126}$, $p_2 = \overline{523}$

Shortest Path

Definition

A **shortest path** in a graph G between two nodes u, v is any path that connects u with v having minimum length.



abe is the shortest path between a and e

abcd and *abed* are both shortest path between a and d

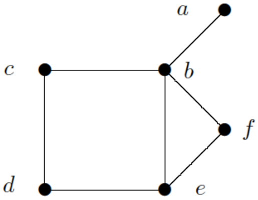
In `NetworkX`:

```
path = nx.shortest_path(G, source=1, target=3)
print("Shortest path:", path)
length = nx.shortest_path_length(G, source=1, target=3)
print("Path length:", length)
```

Distance

Definition

The **distance** $d_G(u, v)$ in a graph G between two nodes u, v is the length of any shortest path connecting u with v or infinite if there is no such path.



$$d(a, e) = 2$$

Properties of the graph distance

Let $G = (V, E)$ be a graph. The distance $d(u, v)$ between nodes u, v has the following properties:

- $d(u, v) \geq 0$, and $d(u, v) = 0 \iff u = v$.
- $d(u, v) = d(v, u)$ (symmetry).
- $d(u, v) \leq d(u, w) + d(w, v)$ (triangle inequality).
- If there is no path between u and v, then $d(u, v) = \infty$.

Connectedness: G is **connected** if $d(u, v) < \infty$ for every pair $u, v \in V$.

Extension: In weighted graphs with nonnegative edge weights, $d(u, v)$ is the minimum total weight of any path between u and v.

Diameter of a graph

Diameter and the Moore bound

Definition

The diameter $\mathrm{diam}(G)$ of a graph $G = (V, E)$ is the maximum distance between two vertices:

$$\mathrm{diam}(G) = \max_{u,v \in V} d(u, v).$$

Diameter and the Moore bound

Definition

The diameter $\mathrm{diam}(G)$ of a graph $G = (V, E)$ is the maximum distance between two vertices:

$$\mathrm{diam}(G) = \max_{u,v \in V} d(u, v).$$

Theorem (Moore Bound)

$$N \le 1 + \Delta \cdot \sum_{k=0}^{D-1} (\Delta - 1)^k.$$

Diameter and the Moore bound

Definition

The diameter $\mathrm{diam}(G)$ of a graph $G = (V, E)$ is the maximum distance between two vertices:

$$\mathrm{diam}(G) = \max_{u,v \in V} d(u, v).$$

Theorem (Moore Bound)

$$N \ \leq \ 1 + \Delta \cdot \sum_{k=0}^{D-1} (\Delta - 1)^k.$$

Why is this useful?

It shows that, if degrees are bounded, you cannot have both a very small diameter and a very N.

Small-world networks keep diameters small using **hubs** or **shortcuts**.

Proof: Moore Bound

Fix a vertex v. Since the diameter is D, every vertex lies within distance D of v. To get the maximal N given the constraints:

- Distance 0: only v.
- Distance 1: at most Δ neighbors of v.
- Distance 2: each neighbor adds at most $\Delta - 1$ new vertices, giving at most $\Delta(\Delta - 1)$.
- Distance i: at most $\Delta(\Delta - 1)^{i-1}$ vertices.

Summing up to distance D,

$$N \leq 1 + \Delta \sum_{i=0}^{D-1} (\Delta - 1)^i.$$

When $\Delta = 2$, the graph is a path or cycle, giving $N \leq 2D + 1$. $\square$

Breath–First Search (BFS)

Definition

The Breadth–First Search explores a graph starting from a root node v:

- Discover all neighbors of v first.
- Then, in order, discover neighbors of those neighbors, and so on.

BFS naturally finds shortest paths from the root to all other vertices.

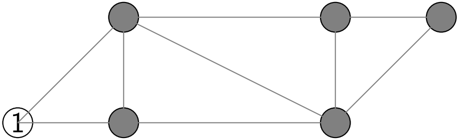
Breath–First Search (BFS)

Definition

The Breadth–First Search explores a graph starting from a root node v:

- Discover all neighbors of v first.
- Then, in order, discover neighbors of those neighbors, and so on.

BFS naturally finds shortest paths from the root to all other vertices.



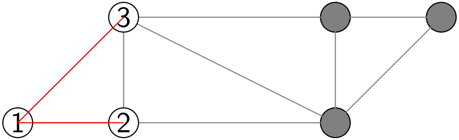
Breath–First Search (BFS)

Definition

The Breadth–First Search explores a graph starting from a root node v:

- Discover all neighbors of v first.
- Then, in order, discover neighbors of those neighbors, and so on.

BFS naturally finds shortest paths from the root to all other vertices.



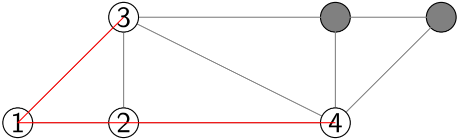
Breath–First Search (BFS)

Definition

The Breadth–First Search explores a graph starting from a root node v:

- Discover all neighbors of v first.
- Then, in order, discover neighbors of those neighbors, and so on.

BFS naturally finds shortest paths from the root to all other vertices.



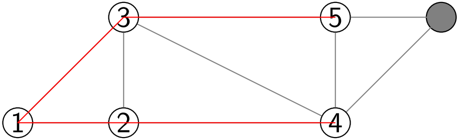
Breath–First Search (BFS)

Definition

The Breadth–First Search explores a graph starting from a root node v:

- Discover all neighbors of v first.
- Then, in order, discover neighbors of those neighbors, and so on.

BFS naturally finds shortest paths from the root to all other vertices.



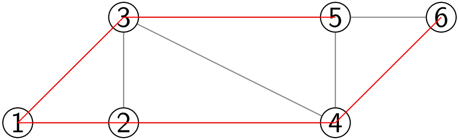
Breath–First Search (BFS)

Definition

The Breadth–First Search explores a graph starting from a root node v:

- Discover all neighbors of v first.
- Then, in order, discover neighbors of those neighbors, and so on.

BFS naturally finds shortest paths from the root to all other vertices.



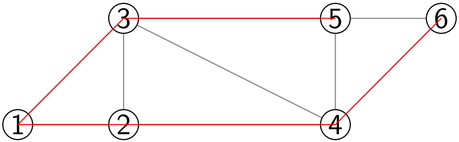
Breath–First Search (BFS)

Definition

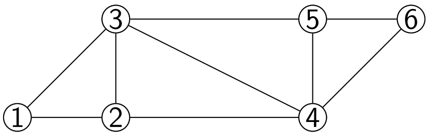
The Breadth–First Search explores a graph starting from a root node v:

- Discover all neighbors of v first.
- Then, in order, discover neighbors of those neighbors, and so on.

BFS naturally finds shortest paths from the root to all other vertices.



Diameter



$$\mathrm{diam}(G) = 3$$

Definition

The **diameter** $\mathrm{diam}(G)$ of a graph G is the max distance between two pair of nodes of G

$$\mathrm{diam}(G) \; = \; \max_{u,v \in V} d(u, v).$$

Graph Eccentricity

Definition

The **eccentricity of a vertex** u is: $\varepsilon(u) = \max_{v \in V} d(u, v)$.

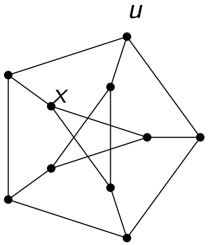
The **eccentricity of a graph** is: $\mathrm{ecc}(G) = \min_{u \in V} \varepsilon(u)$.

(every minimizer is in some sense central)

- Captures the distance from the "most central" vertex to the farthest node.
- Always satisfies $\mathrm{ecc}(G) \leq \mathrm{diam}(G)$ (indeed $\mathrm{diam}(G) = \max_{u \in V} \varepsilon(u)$).
- **Applications:**
  - In communication networks: optimal placement of a hub or server.
  - In social networks: identifying the most central or influential actors.
  - In epidemics: best/worst nodes to start monitoring or intervention.

Exercise 1

Given the following graph:



a) Determine the shortest path and length between nodes u and x.

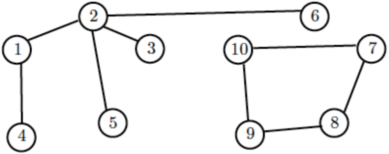
b) Determine its diameter.

c) Determine the local diameter of node z

d) Which is the minimum number of edges that we have to add to the graph so that its eccentricity is 1?

Connectivity

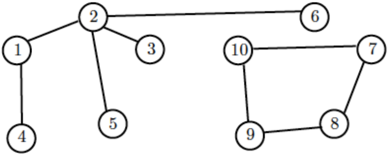
Recall: Connectivity



Definition

Two nodes u, v are connected in G if there exists a path connecting both nodes. A graph is connected if every two nodes are connected.

Recall: Connectivity



Definition

Two nodes u, v are connected in G if there exists a path connecting both nodes. A graph is connected if every two nodes are connected.

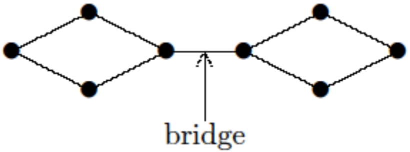
Definition

The connected components G are the maximal connected subgraphs.

Bridge

Definition

A **bridge** in a graph G is any edge such that when removing it from the graph, the number of connected components is increased.



(More) special graphs

Special Graphs: Trees

Definition

A tree is a connected and acyclic graph

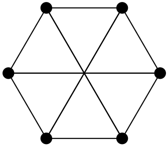
Paths P_n, Stars S_n are trees

- In a tree, every pair of vertices is joined by a unique path.
- The number of edges is one less than the number of vertices.
- Every connected graph contains a tree as a subgraph with the same vertex set (spanning tree).

Trees appear everywhere

Internet routing (spanning trees avoid cycles); **Data structures** in computer science (binary search trees, decision trees); **Phylogenetics** in biology (evolutionary trees); **Epidemics** and **rumor spreading** (branching processes); **Hierarchical clustering** (data science).

Special Graphs: Regular Graphs



Definition

A graph is called r-**regular** if every node has exactly degree r.

- Example: a cycle C_n is 2-regular.
- Example: a complete graph K_n is $(n-1)$-regular.

For an r-regular graph with N vertices:

$$L = \frac{Nr}{2}.$$

(try to prove it; see also Exercise C below)

Exercise A: Complements

a) Show that the complement of an r–regular graph on N nodes is $(N - r - 1)$–regular.

b) Is the complement of a bipartite graph always bipartite? Give a counterexample if not.

c) Give an example of a 4-node graph G such that both G and its complement are connected.

Exercise B: Connectivity

a) Show that if a graph G is not connected, then its complement $\overline{G}$ must be connected.

b) Consider a tree T (connected, acyclic). Is its complement always connected? Give an example.

Exercise C: Adjacency matrix

Let A be the adjacency matrix of a graph G.

a) Show that G is r–regular if and only if

$$A\mathbf{1}_N = r\mathbf{1}_N,$$

where $\mathbf{1}_N$ is the all-ones vector in $\mathbb{R}^N$.

b) Suppose

$$I + A + A^2 + A^3 = J,$$

where J is the all-ones matrix. What does this imply about the diameter of G?