1. Recaps of graph theory

Teacher: Arnaud Casteigts

In this first lesson, we review some background on graph theory. Some definitions are given in a intuitive way, prioritizing simplicity over formality. These definitions will be used in the subsequent lessons when we focus on temporal graphs.

1.1 Basic definitions

1.1.1 Undirected graphs

An **undirected graph** $G = (V, E)$ (or simply a graph) is made of a set of **vertices** V and a set of **edges** E connecting these vertices, where each edge is defined by a *pair* of vertices. For example, the following graph corresponds to $V = \{a, b, c, d\}$ and $E = \{\{a, b\}, \{b, c\}, \{a, c\}, \{c, d\}\}$. We write n for $|V|$ and m for $|E|$. Here, $n = 4$ and $m = 4$.

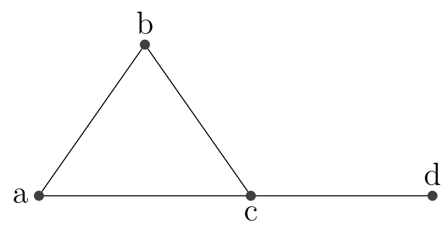


Figure 1: A graph G_1.

In general, a graph may also have **loops** (edge from a vertex to itself), or **multiple edges** (several edges between the same pair of vertices). Unless otherwise mentioned, we will consider only **simple graphs**, i.e. graphs without loops nor multiple edges, like G_1. The rest of the definitions below consider simple graphs only.

The **degree** of a vertex u, noted $d(u)$ is the number of edges incident to u, namely $d(v) = |\{e \in E \mid v \in e\}|$. Here, $d(a) = 2$ and $d(c) = 3$. The **neighbors** of u are all the vertices that share an edge with u. Thus, the number of neighbors of a vertex corresponds to its degree. If we have an edge $e = \{u, v\}$, then u and v are called the **endpoints** of e.

A **path** from u to v is a sequence of vertices $\langle u_1, u_2, ..., u_k\rangle$ such that $u = u_1$, $v = u_k$, and for all $i < k$, $\{u_i, u_{i+1}\} \in E$. For example, $\langle a, b, c, d\rangle$ is a valid path in G_1, and so is $\{a, c, b, a, c, d\}$. But $\langle a, c, b, d\rangle$ is not a path. The **length** of a path is the number of *edges* that it uses. So, the above two examples have length 3 and 5, respectively. A **cycle** is a path from a vertex to itself, for example $\langle a, b, c, a\rangle$. A path (or cycle) is called simple if it does not visit a vertex twice (except for the starting point, for the cycle).

A graph is **connected** if a path exists between every pair of vertices. Otherwise, it can be partitionned into a set of **connected components**. A graph is **complete** if an edge exists between every pair of vertices. Connectivity and completeness should not be mistaken; for example, G_1 is connected, but is not complete. Finally, the **distance** between two vertices is the length of the shortest path between them (∞ if no paths exist), and the **diameter** of the graph is the largest distance over all pairs of vertices (G_1 has diameter 2, which is the distance between a and d, or the distance between b and d).

1.1.2 Directed graphs

A **directed graph** $G = (V, A)$ (also called a digraph) is made of a set of vertices V and a set of **arcs** (or directed edges), where each arc is defined by an *ordered pair* of vertices. For example, the graph on fig. 2 corresponds to $V = \{a, b, c, d\}$ and $A = \{(b, a), (a, c), (b, c), (c, b), (c, d)\}$.

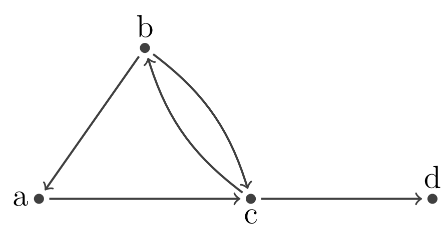


Figure 2: A directed graph G_2.

Simple digraphs are defined analogously to simple graphs (no loop nor multiple arcs). However, two arcs can exist between the same pair of vertices if they have different directions (they are not the same arc!), like in G_2 between b and c.

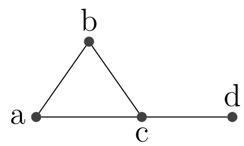
The **in-degree** of a vertex u, noted $d^-(u)$ is the number of incoming arcs to u. Its **out-degree** $d^+(u)$ is the number of outgoing arcs from u. For example, in G_2, we have $d^-(b) = 1$ and $d^+(b) = 2$. **In-neighbors** and **out-neighbors** are defined analogously. They are also called **predecessors** and **successors**.

A **path** from u to v in a digraph is defined analogously as a path in an undirected graph, except that it must satisfy $(u_i, u_{i+1}) \in A$ (so direction matters!). Observe that in digraphs, the reachability relation is not symmetric, e.g. in G_2, there is a path from a to d, but not from d to a.

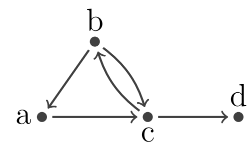
A digraph is **strongly connected** if there exists a path from every vertex to every other vertex. It is **weakly connected** if a path would exist if we ignored the direction of the arcs. For example, G_2 is weakly connected, but it is not strongly connected.

1.1.3 Adjacency matrix

Graphs can be represented in various ways. A common one is as an **adjacency matrix**, which is a $n \times n$ matrix that encodes the edges (resp. non-edges) as 1 (resp. 0).



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



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

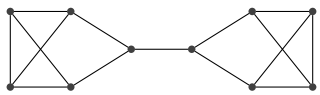
Adjacency matrices are not memory efficient, especially when the graph is sparse (has few edges). But they are convenient for algebraic manipulation of graphs. For example, if M is the adjacency matrix of some graph, then M^k describes the number of paths of length k between any pair of vertices in that graph (you can try it!).

1.1.4 Some classes of graphs

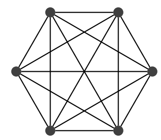
It is often useful to consider specific families (or classes) of graphs. Some of the most common are:

- Regular: all vertices have the same degree.
- Complete: all pairs of vertices share an edge.
- Cycle: consists of a cycle.
- Tree: contains no cycles.
- Grid: see the picture.
- Bipartite: V can be split in two parts, such that all the edges are between these parts.
- Planar: can be drawn on the plane without crossing edges.

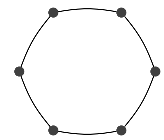
These classes of graphs are illustrated in fig. 3e.



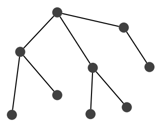
(a) A 3-regular graph.



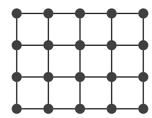
(b) A complete graph.



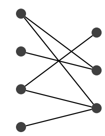
(c) A cycle graph.



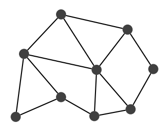
(d) A tree.



(e) A grid graph.



(f) A bipartite graph.



(g) A planar graph.

1.2 Exercises

All the graphs we consider in these exercises are undirected and simple.

1.2.1 Regular graphs

- Build a 3-regular graph with 4 vertices, 5 vertices, 6 vertices, and 7 vertices.
- What do you think? Explain why!
- Now same question for 4-regular graphs.

1.2.2 Playing with degrees

1. Find a graph (with $n > 1$), where every vertex has a different degree.
2. A sequence of degree is realizable if we can find a graph whose degrees are these numbers (for example, a triangle graph corresponds to the sequence $2, 2, 2$). Which of the following sequences are realizable?

    - 3, 3, 2, 1, 1
    - 3, 3, 1, 1
    - 3, 3, 2, 2
    - 4, 2, 1, 1, 1, 1
    - 5, 3, 2, 1, 1, 1

- 5, 4, 3, 1, 1, 1, 1

3. Find two different graphs corresponding to the sequence 3, 2, 2, 2, 1.

1.2.3 The six friends

Two persons either know each other (they share an edge), or they don't (there share no edges). We say that three persons know each other if their edges form a triangle (cycle of size 3). They don't known each other if this triangle is *empty* (no edges exist among them).

Consider the following declaration: "In a group of five persons, there is *always* three persons who know each other or three persons who do not know each other."

1. Is this declaration true?

2. What about 6 persons instead of 5?