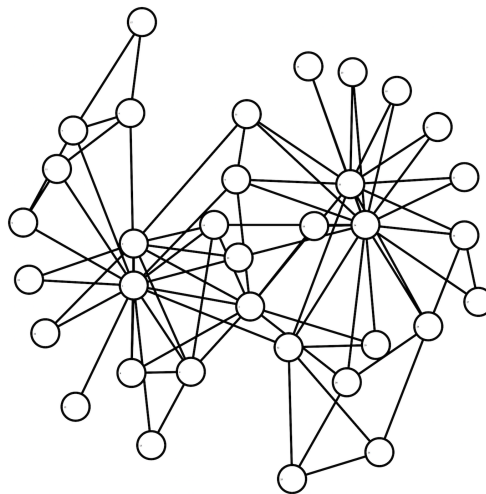


Graph definitions & terminology

Many real-world systems are naturally modelled as *networks*, including social, information, biological, technological, lexical, and other systems. A network is represented by a mathematical *graph* consisting of points called *nodes*, which are connected by lines called *edges*.



Note that the choice of graph used to represent a network is a design decision. In some cases, there is a unique, unambiguous representation, whereas in other cases, there are multiple possible representations.



network



another network



same graph

Simple undirected graph

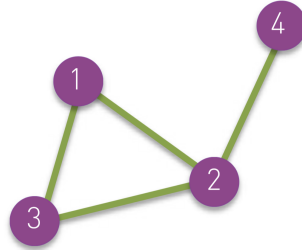
Consider a friendship network of classmates represented by a *simple undirected graph* $G(V, E)$, where V is the set of nodes (i.e. students) and E is the set of edges connecting these nodes (i.e. friendships). Denote the number of nodes by $n = |V|$ and the number of edges by $m = |E|$.

For example,

$$V = \{1, 2, 3, 4\},$$

$$E = \{\{1, 2\}, \{1, 3\}, \{2, 3\}, \{2, 4\}\},$$

where $n = 4$ and $m = 4$.

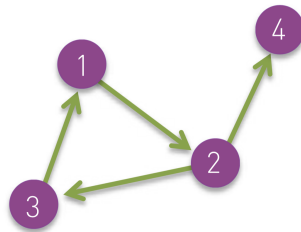


Simple directed graph

In the case of an *Twitter/X* follower network, the corresponding graph is a *simple directed graph*, where the edges point from the source nodes (i.e. followers) to the target nodes (i.e. followees).

For example,

$$E = \{(1, 2), (2, 3), (2, 4), (3, 1)\}.$$

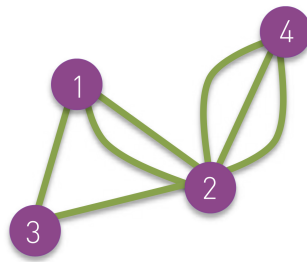


Undirected multigraph

In the case of a transportation network (e.g., roads between cities), there can be multiple edges between two nodes (i.e. a *multigraph*).

For example,

$$E = \{\{1, 2\}, \{1, 2\}, \{1, 3\}, \{2, 3\}, \{2, 4\}, \{2, 4\}, \{2, 4\}\}.$$

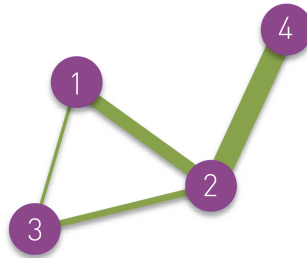


Undirected weighted graph

In the case of a communication network (e.g., calls or texts between classmates), each edge $\{i, j\}$ is assigned a positive weight $w_{ij} \in \mathbb{R}^+$ (i.e. a *weighted graph*).

For example,

$$E = \{(\{1, 2\}, 2.0), (\{1, 3\}, 0.5), (\{2, 3\}, 1.0), (\{2, 4\}, 3.0)\}.$$

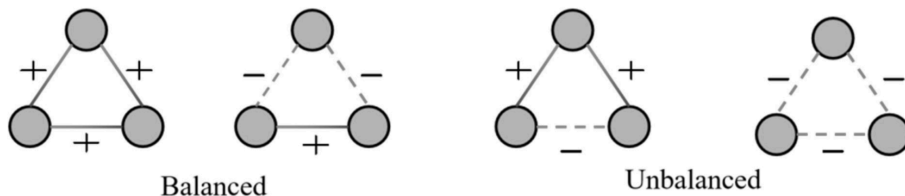


Undirected signed graph

Note that weights in a weighted graph must be non-negative $w_{ij} \in \mathbb{R}^+$. If we allow for arbitrary weights $w_{ij} \in \mathbb{R}$, we get a *signed graph* (e.g., alliances and rivalries in international relations). Most commonly, $w_{ij} \in \{1, -1\}$.

For example,

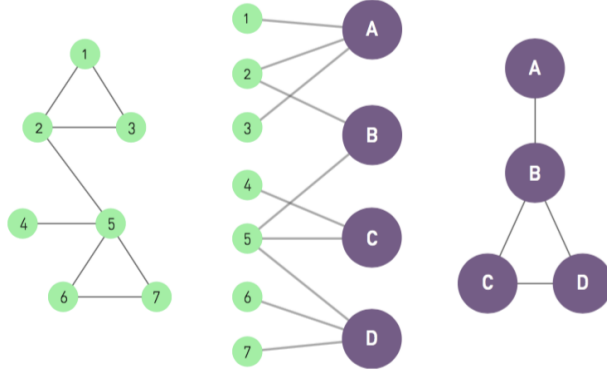
$$E = \{(\{1, 2\}, 1), (\{1, 3\}, -1), (\{2, 3\}, -1), (\{2, 4\}, 1)\}.$$



Bipartite & collaboration graphs

In a *bipartite graph*, nodes are divided into two disjoint sets V_1 and V_2 , such that $V = V_1 \cup V_2$. Edges $E \subseteq V_1 \times V_2$ can exist only between the nodes of different sets (e.g.,

users and products in recommender systems). A projection to only one set of nodes V_1 or V_2 is called a *collaboration graph*, where we connect two nodes if they share a neighbor in the original graph.



Neighborhood, degree & distance

Let Γ_i denote the *neighborhood* of node i in a simple undirected graph. Note that, in an undirected multigraph, Γ_i is a multiset.

$$\Gamma_i = \{j \mid \{i, j\} \in E\}$$

The number of neighbors of node i is called the *degree* $k_i = |\Gamma_i|$. If we denote the average degree by $\langle k \rangle = \frac{1}{n} \sum_i k_i$, then

$$n\langle k \rangle = 2m.$$

In the case of a directed graph, we differentiate between the node's *predecessors* Γ_i^{in} and *successors* Γ_i^{out} defined as

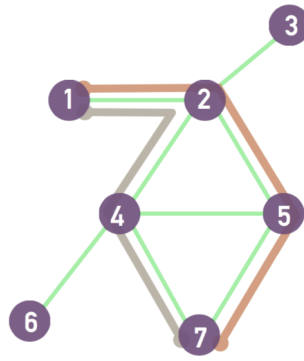
$$\Gamma_i^{in} = \{j \mid (j, i) \in E\} \quad \& \quad \Gamma_i^{out} = \{j \mid (i, j) \in E\}.$$

Similarly, the number of predecessors is called the *in-degree* $k_i^{in} = |\Gamma_i^{in}|$ and the number of successors is called the *out-degree* $k_i^{out} = |\Gamma_i^{out}|$.

$$k_i = k_i^{in} + k_i^{out}$$

Finally, the *distance* d_{ij} between the nodes i and j is the minimum number of hops (i.e. edges) required to get from i to j (or vice versa).

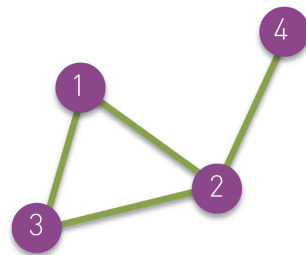
In the example below, $d_{17} = 3$.



If i and j are neighbors in an unweighted graph, then $d_{ij} = 1$. For a weighted graph, we usually adopt the convention $d_{ij} = w_{ij}^{-1}$.

Edge list representation

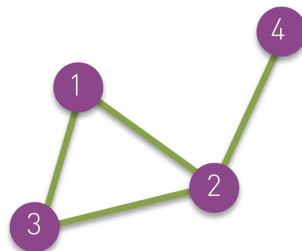
In the examples above, the graph is defined by its *edge list* E . This is the standard representation of graphs in the *PyTorch Geometric* library for graph machine learning, as well as in the *Pajek* format for storing graphs in a file (i.e. `.net` format).



```
*vertices 4
1 "1"
2 "2"
3 "3"
4 "4"
*edges
1 2
1 3
2 3
2 4
```

Adjacency list representation

Alternatively, the graph can be defined by an *adjacency list*, which stores Γ_i for each node i . This is the representation used in the *NetworkX* library for network analysis.



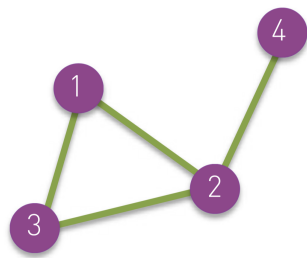
```
1: [2, 3]
2: [1, 3, 4]
3: [1, 2]
4: [2]
```

Adjacency matrix representation

Finally, the graph may also be represented by an $n \times n$ *adjacency matrix* A , with entries $A_{ij} > 0$ whenever nodes i and j are connected, and $A_{ij} = 0$ otherwise. Such a

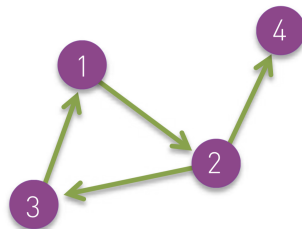
representation is useful for writing mathematical expressions in a concise form.

In the case of an undirected graph, the matrix is symmetric $A = A^T$.



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

In the case of a directed graph, $A \neq A^T$, where the entry A_{ij} denotes an edge either from i to j or from j to i (depending on the convention).



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