02-680 Module 4

Essentials of Mathematics and Statistics

Aidan Jan

September 11, 2025

Graphs

Graphs are powerful tools for modeling structured **relationships** between entities. A graph consists of:

- **Nodes** (or Vertices): The entities in the graph
- **Edges** (or Links): The relationships between pairs of nodes.

A graph is also called a **network**.

Undirected Graphs

- In **undirected graphs**, edges represent mutual relationships. If node A is connected to node B, then node B is also connected to node A.

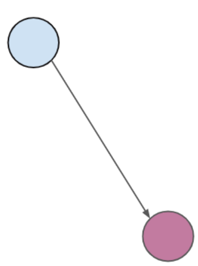
Many real-world relationships are mutual. This representation is suitable for systems where all connections are peer-to-peer.

Directed Graphical Models

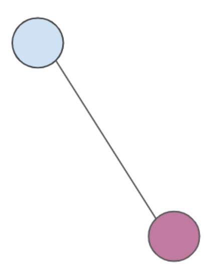
In a social network, each **node** represents a person, and each **directed edge** represents a relationship such as following, influencing, or messaging. These directed connections capture how information, behavior, or influence propagates across the network.

Directed Graph vs. Undirected Graph

In a directed graph, an edge is an **ordered pair** of vertices ("an edge from u to v") and in an undirected graph, an edge is an unordered pair of vertices ("an edge between u and v").



directed graph



undirected graph

A **graph** is a mathematical structure used to model pairwise relationships. We typically write a graph as a tuple $G = \langle V, E \rangle$.

- V is a set of nodes or vertices
- E is a set of edges, each connecting a pair of nodes.
  - $E \subseteq V \times V$ if G is **directed**
  - $E \subseteq \{\{u, v\} \,:\, u, v \in V\}$ if G is **undirected**
  - Note that one edge can loop around the same node. (e.g., $u = v$)

Nodes in a graph are typically drawn as circles ad edges are represented as lines. In directed graphs, each edge is drawn with an arrow indicating its orientation ("which way it goes").

Parents and Children

In a directed graph, there exists the notion of parent nodes and child nodes. If an arrow connects two variables X and Y (in either direction) we say that X and Y are adjacent. If there is an arrow from X to Y, then X is a parent of Y, and Y is a child of X. A **directed path** between two variables is a set of arrows all pointing in the same direction linking one variable to the other such as:

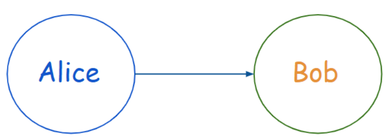
$$X \longrightarrow \cdots \longrightarrow Y$$

Some Examples of Graphs

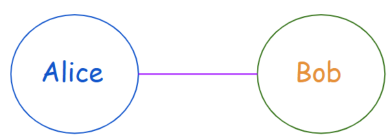
To understand a **directed graph**, think about the difference between:

- Twitter follows: $A \to B$ (directed)
- Facebook friendships: $A - B$ (undirected)

In Twitter, Alice can follow Bob without Bob following back — this is a one-way (directed) edge.



In Facebook, friendship is mutual — it exists only if both agree, representing an undirected edge.



Path in a Directed Graph

Some examples of things that can be represented as **graphs**: train/flight maps, cell interactions, social networks, etc. A **path** is a sequence of nodes:

$$\langle v_1, v_2, v_3, \ldots, v_k \rangle$$

$$v_1 \to v_2 \to v_3 \to v_4 \dashrightarrow v_k$$

- $\langle v_i, v_{i+1} \rangle \in E$ if G is **directed**
- $\{v_1, v_{i+1}\} \in E$ if G is **undirected**

We say that the **path** has length k, and that a path is from v_1 to v_k.

Route Graph

In the case of a route graph (e.g., for navigation or delivery), each edge might be labeled with a street name (a string), but in other types of graphs the labels could be numerical values (such as distance, time, or cost) - which we usually call **weights**.

- A **path label** could be the concatenation of street names (e.g. "Main St." →"Broadway" →"5th Ave.")
- A **path weight** could be the **sum** of distnaces or travel time along the path. Edge labels can carry qualitative (string) or quantitative (numeric) information, and we interpret paths differently depending on the type of label.

Graph Function

We can also **label** or **weight edges** in some scenarios. What does this mean?

- In many types of graphs (especially weighted or labeled graphs) we want to assign a value or label to each edge.
- These labels can be:
  - Strings (e.g., street names, course codes, etc.)
  - Numbers (e.g., cost, distance, weight, time)

Formally, edges with string labels are typically represented by a function:

$$\ell \,:\, E \to \Sigma$$

(or maybe edges with weight labels function $w \,:\, E \to \mathbb{R}$)

Instead of assigning labels manually, we define a **function** that maps each edge to a value.

Function	**Purpose**	**Output Type**
ℓ	Edge label	A string (from Σ^*)
w	Edge weight	A real number ($\mathbb{R}$)

Neighborhood of a Node: Undirected Graph

Let $G = \langle V, E \rangle$ be an **undirected graph**, and let $u \in V$ be a **node**. The neighborhood of u is the set $v \in V \,:\, \{u, v\} \in E$.

- That is, the set of all neighbors of u

$N_G(v)$ is the set of all nodes u

- If $\{u, v\} \in E$, it means there is an edge connecting u and v, so node u is considered a neighbor of node v.

Degree: Undirected Graph

The **degree** of a node u is an undirected graph G is the size of the neighborhood of u in G, that is, the number of nodes adjacent to u.

Aside: Indicator Function

The **indicator function** is a simple mathematical tool used to express whether a condition is **true** or **false**, in a numeraical form. Given an arbitrary set X, the indicator function of a subset A of X is the function:

$$1_A \; : \; X \mapsto \{0, 1\}$$

defined by

$$1_A(X) = \begin{cases} 1 & \text{if } x \in A \\ 0 & \text{if } x \notin A \end{cases}$$

Complete Graphs: LECTURE ENDED HERE