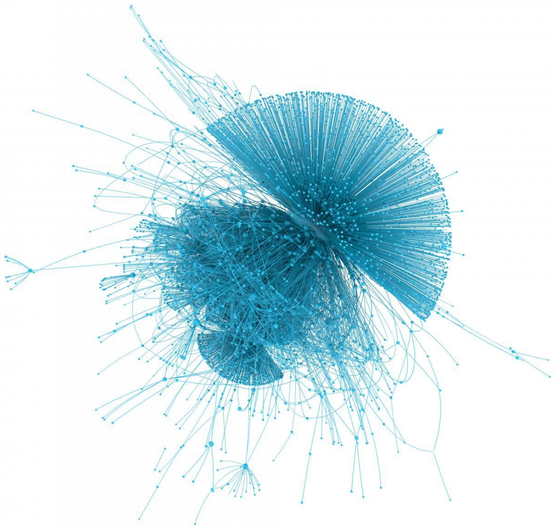
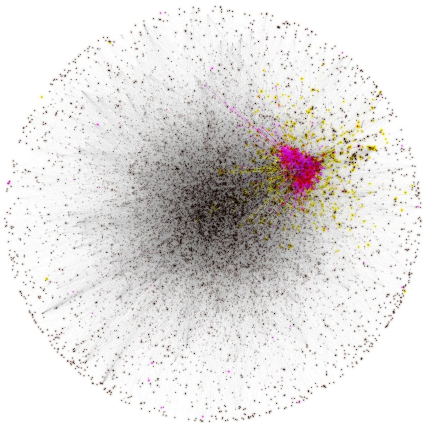
Graph theory



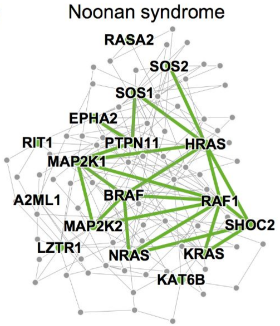


Images from [Barabási Lab]

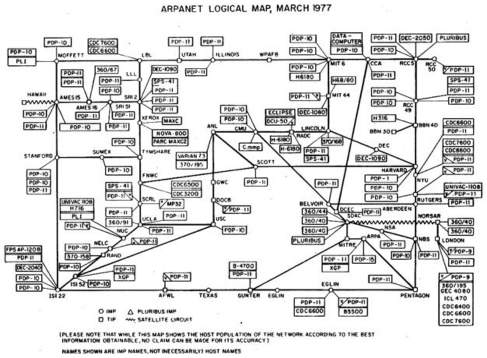
Graphs are everywhere

Underground networks

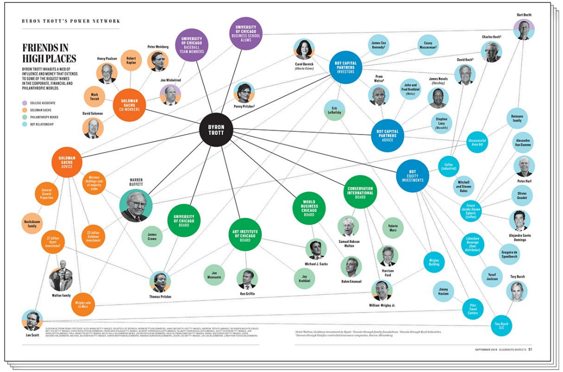
Social networks



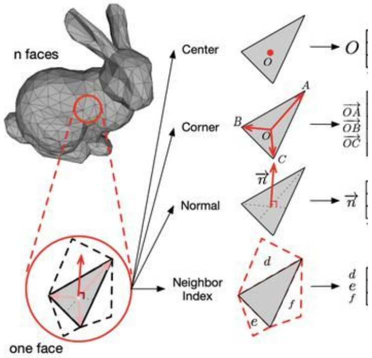
Disease pathways



Internet connection



Financials

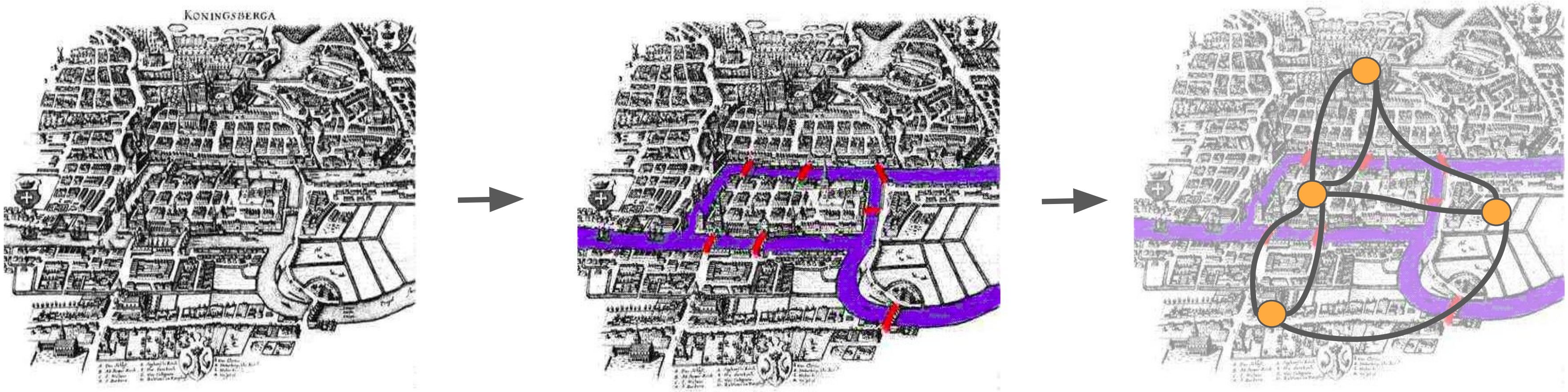


3D Shapes

Where graph theory comes?

Seven Bridges of Königsberg problem, resolved and demonstrated in Euler’s paper

Solutio problematis ad geometriam situs pertinentis (1736)



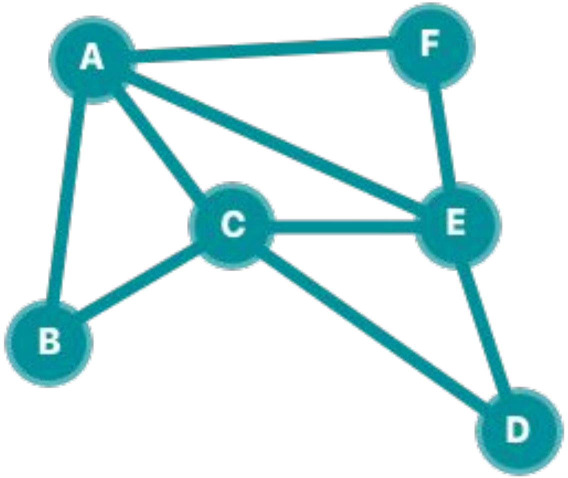
What is a graph?

A structure defined as a pair ***G = (V, E)***.

Where:

- ➔ ***V*** is a set called vertices
- ➔ ***E*** is a set of pairs called edges

For vertices *x* and *y* of an edge {x, y}, they are called endpoints of the edge. Also, a vertex doesn't have to join an edge.

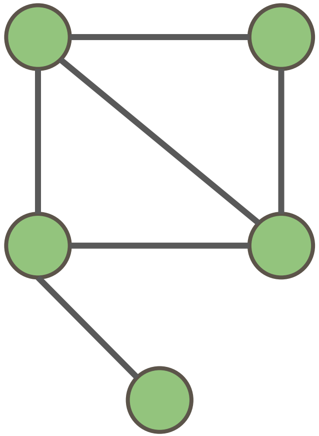


V: {A,B,C,D,E,F}

E: {AB,AC,AE,AF,BC,CD,CE,DE,EF}

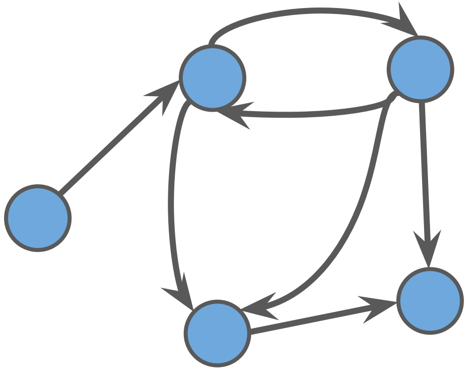
Type of graphs

Undirected graph



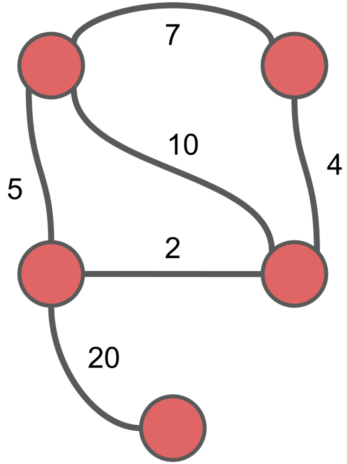
$e = (u, v)$

Directed graph



$e = (u, v)$

Weighted graph

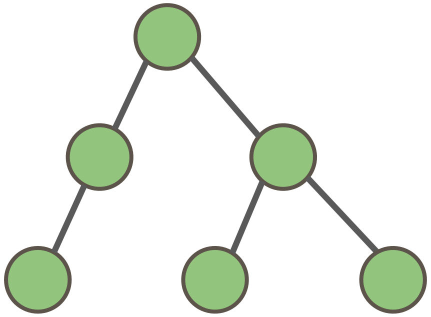


$e = (u, v, w)$

Newman, M. E. (2003) [The structure and function of complex networks]

Special graphs

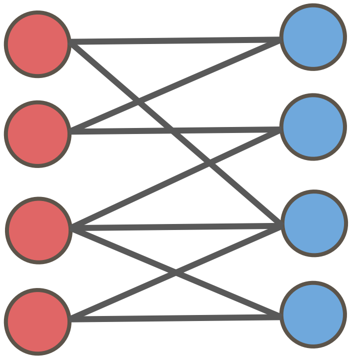
Tree



N = number of nodes
nodes = N
edges = N - 1

Undirected graph with no cycles
Other trees: rooted trees, binary trees…

Bipartite

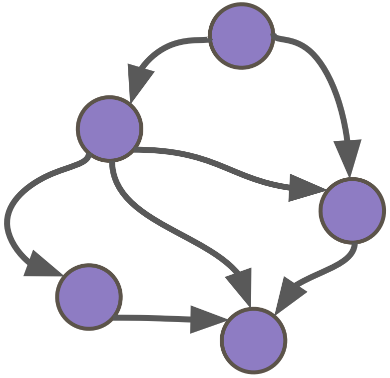


2 independent groups

There are not odd cycles

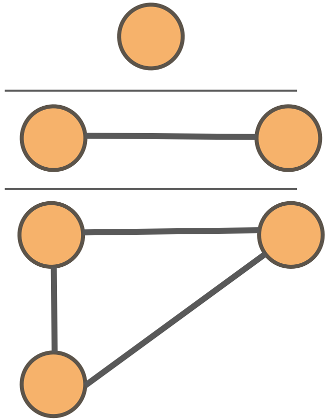
DAG

(directed acyclic graph)



Directed graph with no cycles

Complete

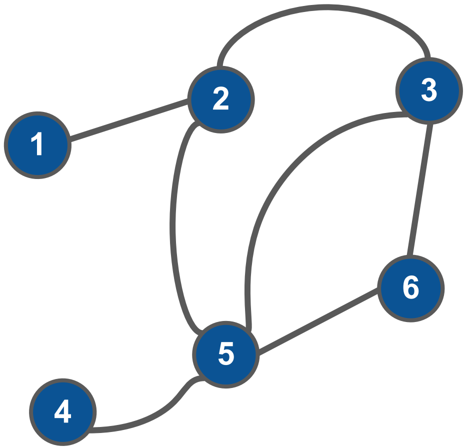


N = number of nodes
nodes = N
edges = (N · (N-1)) / 2

Denoted as K_n
Maximum of edges that a graph can have

And more graphs such as r-regular, null graph, eulerian path, Hamiltonian graph, star...

Graph representation



Adjacency matrix

- ➔ Binary matrix of *n x n*, where *n* is the number of nodes.
- ➔ Space inefficient.
- ➔ Iterating has high cost for big graphs.

$$m_{i,j} = \begin{cases} 1 & v_i \sim v_j \\ 0 & otherwise \end{cases}$$

	1	2	3	4	5	6
1	0	1	0	0	0	0
2	1	0	1	0	1	0
3	0	1	0	0	1	1
4	0	0	0	0	1	0
5	0	1	1	1	0	0
6	0	0	1	0	0	0

Adjacency list

- ➔ Greate for sparse graphs.
- ➔ Iterating is efficient.
- ➔ It is a more complex data structure.

1 ⇒ { 2 }

2 ⇒ { 1, 3, 5}

3 ⇒ { 2, 5, 6}

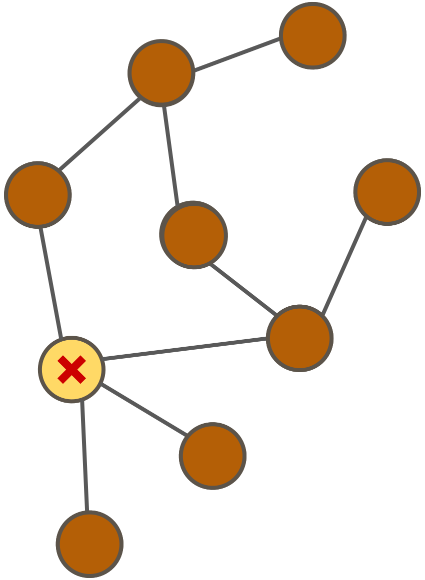
4 ⇒ { 5 }

5 ⇒ { 2, 3, 4, 6}

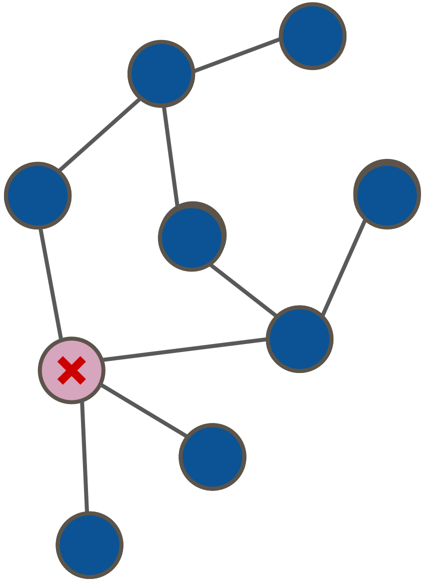
6 ⇒ { 3, 5 }

Some algorithms

BFS (Breadth First Search)



DFS (Depth First Search)

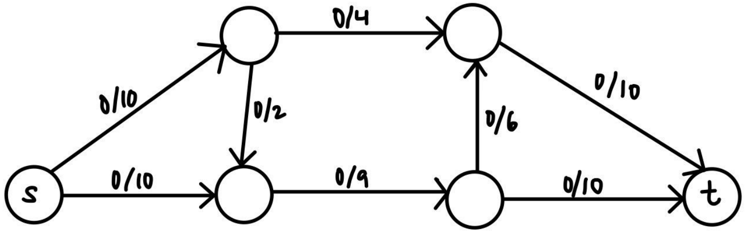


Dijkstra

Search for the shortest path between two nodes.

E. W. Dijstrak - A note on two problems in connexion with graphs

Network flow algorithms



Ford–Fulkerson / Edmonds–Karp

Find the maximum flow between two points.

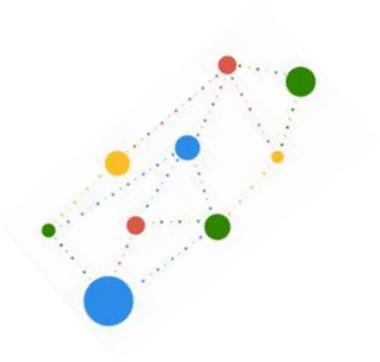
Ford, L. R., Fulkerson, D. R. - Maximal flow through a network

J. Edmonds, R. Karp - Theoretical Improvements in Algorithmic Efficiency for Network Flow Problems

PageRank

Finding the most relevant nodes on a graph.

L. Page, S. Brin, R. Motwani, T. Winograd - The PageRank citation ranking: bringing order to the web



Useful toolkits

[NetworkX](#) [igraph](#) [SNAP](#)

[Neo4j](#) [GraphX | Apache Spark](#) [Titan](#) [Amazon Neptune](#) [Azure Cosmos DB](#)

Hands-on

https://colab.research.google.com/github/dmartmillan/graph-theory-bggagora/blob/main/graph_theory_walkthrough.ipynb

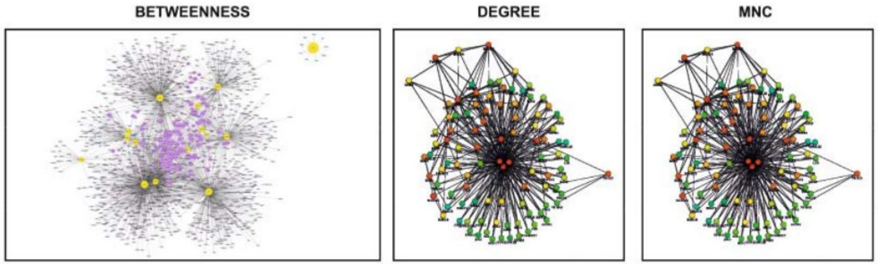
Tools for protein-protein interaction (PPIN)

Tools for protein-protein interaction network analysis in cancer research

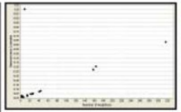
Rebeca Sanz-Pamplona, Antoni Berenguer, Xavier Sole, David Cordero, Marta Crous-Bou, Jordi Serra-Musach, Elisabet Guinó, Miguel Ángel Pujana & Víctor Moreno

Clinical and Translational Oncology **14**, 3–14 (2012) | Cite this article

https://doi.org/10.1007/s12094-012-0755-9



Betweenness	Gene Symbol
0.41975901	AURKA
0.35056479	TP53
0.34594459	EGFR
0.15654983	CTNNB1
0.14714474	SMAD4
0.12950845	BRAF
0.07258603	KRAS
0.03136116	PIK3CA
0.02229573	TGFBR2
0.0221333	APC



Name	Score	Rank
AURKA	462.00000	1
EGFR	365.00000	2
TP53	358.00000	3
BRAF	253.00000	4
CTNNB1	166.00000	5
SMAD4	149.00000	6
KRAS	76.00000	7
APC	41.00000	8
PIK3CA	40.00000	9
TGFBR2	34.00000	10

Name	Score	Rank
AURKA	241.00000	1
BRAF	198.00000	2
EGFR	114.00000	3
CTNNB1	59.00000	4
TP53	55.00000	5
APC	29.00000	6
SMAD4	26.00000	7
KRAS	16.00000	8
TGFBR2	14.00000	9
MAD2L1	8.00000	10

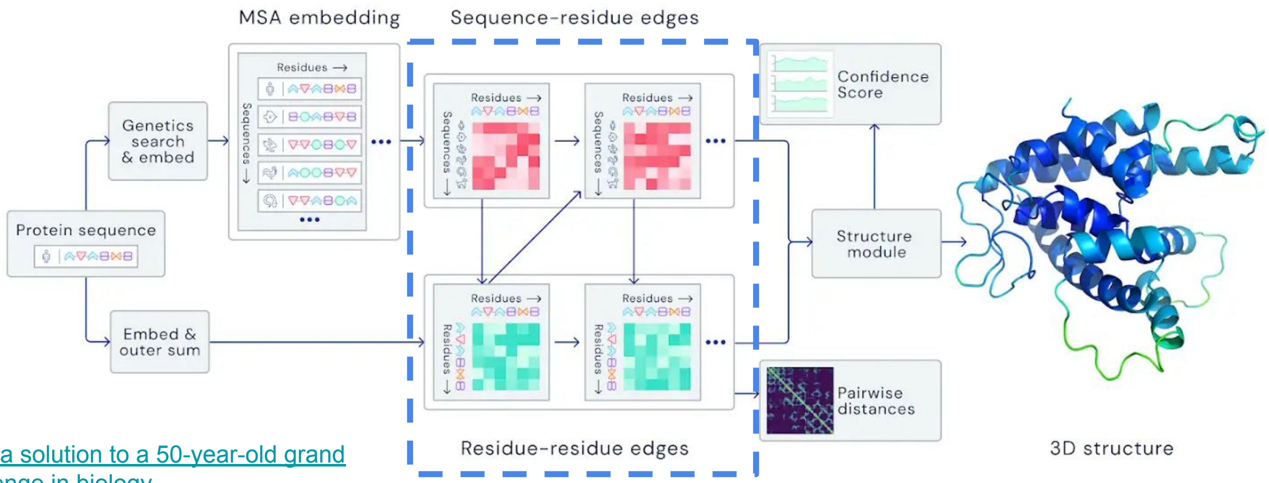
Machine Learning with graphs

Geometric Deep Learning: Grids, Groups, Graphs, Geodesics and Gauges

arXiv:2104.13478

Protein Folding - AlphaFold

- ➔ Nodes: amino acids in a protein structure
- ➔ Edges: proximity between amino acids



Spatial graph

DeepMind - AlphaFold: a solution to a 50-year-old grand challenge in biology

Other applications for ML Graphs

- Recommendations system (e. g. social media: person to follow, ecommerce: some product to buy, recommendation of movies or series...)

  Graph Convolutional Neural Networks for Web-Scale Recommender Systems, KDD 2018

- Drug side effects

  Modeling Polypharmacy Side Effects with Graph Convolutional Networks, Bioinformatics 2018

- Prediction of road network or paths

  Traffic prediction with advanced Graph Neural Networks, DeepMind

- Physical simulation

  Learning to Simulate Complex Physics with Graph Networks, ICML 2020

- Molecule generation

  Graph Convolutional Policy Network for Goal-Directed Molecular Graph Generation, NeurIPS 2018

Thanks for your attention!