Graph concepts

Graphs are made up by vertices (nodes) and edges (links). An edge connects two vertices, or a vertex with itself – loop.

- AC, AC - multiple edges
- BB – loop

The shape of the graph does not matter, only the way the nodes are connected to each other.

Simple graph - does not have loops and multiple edges.

Further reading:
http://www.utm.edu/departments/math/graph/glossary.html
Chapters I-III (p 1-19) of “The structure and function of complex networks”

Symmetrical and directed graphs

Two distinct types of edges: symmetrical and directed (also called arcs). Two different graph frameworks: graph, digraph. In the digraph framework a symmetrical edge means the superposition of two opposite directed edges.

Node degrees

Node degree: the number of edges connected to the node. \( k_i = 4 \)

In directed networks we can define an in-degree and out-degree. \( k_i^\text{in} = 2 \quad k_i^\text{out} = 1 \quad k_i = 3 \)

Average degree \( \langle k \rangle = \frac{1}{N} \sum k_i \)

\( \langle k^\text{in} \rangle = \frac{1}{N} \sum k_i^\text{in} \quad \langle k^\text{out} \rangle = \frac{1}{N} \sum k_i^\text{out} \quad \langle k \rangle = \langle k^\text{in} \rangle = \langle k^\text{out} \rangle \)

What is the relation between the number of edges in a (non-directed) graph and the sum of node degrees? How about in a directed graph?

Statistics of node degrees

Average degree \( \langle k \rangle = \frac{1}{N} \sum k_i = \frac{2E}{N} \)

\( \langle k^\text{in} \rangle = \langle k^\text{out} \rangle = \frac{E}{N} \)

The degree distribution \( P(k) \) gives the fraction of nodes that have \( k \) edges.

Similarly \( P(k^\text{in}) / P(k^\text{out}) \) gives the fraction of nodes that have in-degree \( k^\text{in} \) / out-degree \( k^\text{out} \).
Paths and circuits

Adjacent nodes (vertices) – there is an edge joining them.

In the digraph framework the adjacency only defined in the direction of the arrow.

Path: a sequence of nodes in which each node is adjacent to the next one. Edges can be part of a path only once.

A path in a directed network needs to follow the direction of the edges, thus a symmetrical edge in the digraph framework can be used once in one direction and once in the opposite.

Circuit: a path that starts and ends at the same vertex.

Connected and disconnected graphs

Connected graph: any two vertices can be joined by a path. A disconnected graph is made up by connected components.

Bridge: if we erase it, the graph becomes disconnected.

Connectivity of directed graphs

Strongly connected directed graph: has a path from each node to every other node. Strongly connected components can be identified, but not every node is part of a nontrivial strongly connected component.

In-component: nodes that can reach the scc, out-component: nodes that can be reached from the scc.

Exercises

1. Draw a graph or digraph with 4 nodes such that each node has degree 1 / 2 / 3. Try to use a variety of edges: symmetrical, directed, multiple edges, loops.

2. You have N nodes and need to build a connected graph from them. Each time you add an edge you must pay $1. What is the minimum amount of money needed to build the graph?

3. You are constructing a disconnected graph from N nodes. For each edge you add you receive $1. You are not allowed to use directed edges, loops or multiple edges, and you must stop before the graph becomes connected. What is the most money you can make?
Exercise: the bridges of Konigsberg

Walking problem –
traverse each bridge
do not recross any bridge
return to the starting point

Euler circuit: return to the starting
point by traveling each edge of the
graph once and only once.

Is there a solution to the Konigsberg bridge walking problem?

Euler’s theorem:
(a) If a graph has any vertices of odd degree, it cannot have an Euler circuit.
(b) If a graph is connected and every vertex has an even degree, it has at
least one Euler circuit.

Euler circuits in directed graphs

If a digraph is strongly connected and
the in-degree of each node is equal to
its out-degree, then there is an Euler
circuit

Otherwise there is no Euler
circuit. Basically, this is
because in a circuit we need
to enter any node as many
times as we leave it.

Common subgraphs

Subgraph: a subset of nodes of the original graph
and of edges connecting them. Does not have to contain
all the edges of a node included in the subgraph.

Trees: contain no circuits; N nodes and N-1 edges.

Cycles: circuits where nodes are not revisited;
N nodes and N edges

Cliques: completely connected subgraphs; N nodes
and N(N-1)/2 edges

Special directed subgraphs

Bi-fan
Feed-forward loop: two nonintersecting
directed paths between a start and endpoint

Bi-parallel: two nonintersecting paths of identical length
between a start and endpoint

Feed-back loop: a directed cycle

Network topology and dynamics

Network motifs can illustrate regulatory relationships

- linear pathway
- T-shaped motif
- feedforward loop
- feedback loop
- triangle
- branched loop

Further, dynamical details are needed to describe how multiple inputs on a node are integrated - additive action (e.g. same product for two chemical reactions) - synergy (e.g. transcriptional regulation)

Distances between nodes

The distance between two nodes is defined as the number of edges along the shortest path connecting them.

If the two nodes are disconnected, the distance is infinity.

Average path length of a connected graph

\[
\bar{l} = \frac{1}{N_{\text{pairs}}} \sum_{ij} l_{ij}
\]

where \( N_{\text{pairs}} \) is the number of node pairs

Graph efficiency

To avoid infinities, one can define a graph efficiency (\( \eta \)) as the average inverse distance

\[
\eta = \frac{1}{N_{\text{pairs}}} \sum_{ij} l_{ij}^{-1}
\]

\( N_{\text{pairs}} \) is the number of node pairs

Q: What is the average distance and efficiency of the graph on the left?

Betweenness centrality (load)

- Find all the shortest paths between nodes \( i \) and \( j \) - \( C_{ij}(k) \)
- Determine how many of these pass through node \( k \) - \( C_{ik}(j) \)
- The betweenness centrality of node \( k \) is

\[
B_k = \sum_{i,j \neq k} \frac{C_{ij}(k)}{C_{ij}'}
\]

L. C. Freeman, Sociometry 40, 35 (1977)
Ex: Calculate the betweenness centrality of the nodes in this graph.
Do not count being the starting or ending point of a path \((k \neq i, k \neq j)\). 

\[ E_k = \sum_{i,j} \frac{C(i,j)}{C_{g}(i,j)} \]

\( C(i,j) \) - nr. of shortest paths btw. \( i,j \)
\( C_{g}(i,j) \) - nr. of these paths that contain \( k \)

**Weighted networks**

In some applications it is necessary to quantify edges with weights, corresponding, e.g., to a traversal cost or a geographical distance.

Then the shortest path between two nodes is redefined in terms of weight, e.g. "the path with lowest cost", or "most efficient path".

To find the distance between two nodes in a weighted network, calculate the sum of edge weights on each path (this is the path weight), then select the path with lowest weight

\[ l_i = w_{i1} + w_{i2} + \ldots + w_{ij} \]

where \( i, k, l, m, j \) is the path with minimum weight
\( w_{ij} \) is the weight of edge \( ij \)

**Bipartite graphs**

Group structure (e.g. in social networks) can be incorporated into a bipartite graph.

A bipartite graph has two types of nodes: group nodes (black, numbered) and member nodes (white, lettered). Edges are possible only between different types of nodes: membership in group.

An alternative representation connects all members in a given group - each group becomes a completely connected subgraph.

Local order and clustering

Clique (completely connected subgraphs)

\[ k = N - 1, \quad n = \frac{N(N - 1)}{2} \]

How close the neighborhood of a node is to a clique?

Edges among first neighbors of node i

\[ C_i = \frac{n_i}{k(k-1)/2}, \quad k \neq 0,1 \quad \text{or} \quad 1 \]

Average clustering coefficient

\[ C = \frac{1}{N} \sum_{i \neq i} C_i \]

Ex. 1
N nodes are connected by N edges such that they form a cycle. How does the maximum distance between nodes (the diameter) depend on N? How about the average distance?

Ex. 2
On the ring lattice from above every second neighbor is connected by an edge. What is the clustering coefficient of the nodes?

Ex. 3
Construct a square lattice (grid) L edges long. How does the maximum distance between nodes depend on L?

Regular lattices, ex. 1, 1D lattice (ring)

\[ k = 4 \quad \text{for inside nodes} \]

\[ C = \frac{1}{2} \quad \text{for inside nodes} \]

\[ 1 + \sum_{i=0}^{4} n = l \Rightarrow l_{\text{max}} = \frac{N}{4} \quad \langle l \rangle = \frac{\sum_{i=0}^{4} l}{N} = \frac{N}{8} \]

The average path-length varies as \( \langle l \rangle \propto N \)

Constant degree, constant clustering coefficient.

Clustering coefficient of 1D lattice

The origin (black) node is connected to k (=4) nodes on each side. Total possible edges among neighbors: 2k(2k-1)/2

Enumerate edges that are actually there:

\[ n = \sum_{k=1}^{k} \left( \frac{2k}{2k-1} \right) \]

First neighbors on lattice second neighbors edge "length" kth neighbors

\[ C = \frac{2 \sum_{i=0}^{k} (2k - i)}{2k(2k-1)} = \frac{3 k - 1}{2k - 1} \]
Regular lattices, ex. 2, 2D lattice

\[k = 6 \text{ for inside nodes}\]
\[C = \frac{6}{15} \text{ for inside nodes}\]
\[l + \sum_{l=0}^{N} 6l = N \Rightarrow l_{\text{max}} \propto N^{0.5}\]
\[<l> \geq L \propto N^{1/2}\]

In general, the average distance varies as \(<l> \propto N^{1/D}\)
where D is the dimensionality of the lattice. Constant degree (coordination number), constant clustering coefficient.

Regular lattices, ex. 3, the Cayley tree

\[k = 3 \text{ for inside nodes}\]
\[k = 1 \text{ for surface nodes}\]
\[C = 0\]
\[l + j \sum_{j=1}^{N} 2^{j-1} \approx N \Rightarrow l_{\text{max}} \propto \frac{\log N}{\log <k>}\]
\[<l> \geq \frac{\log N}{\log <k>}\]

Distances vary logarithmically with N. Constant degree, no clustering.