

## Exercise 2 - Theory

```
In [ ]: import networkx as nx
```

### Problem 1

*Discuss briefly advantages and disadvantages of representing a network by an edgelist and an adjacency matrix.*

An advantage of the edgelist is the efficient storage requirements if the graph is sparse but if the graph is dense then it becomes inefficient to store.

Advantages of the adjacency matrix are the easy to find neighbors of a given node and to find paths of length  $k$  one can calculate the  $k$ -th power of the matrix. Another advantage is the more efficient storage requirements of dense graphs compared to the edgelist. On the other hand the naive implementation for sparse graphs is very inefficient to store.

### Problem 2

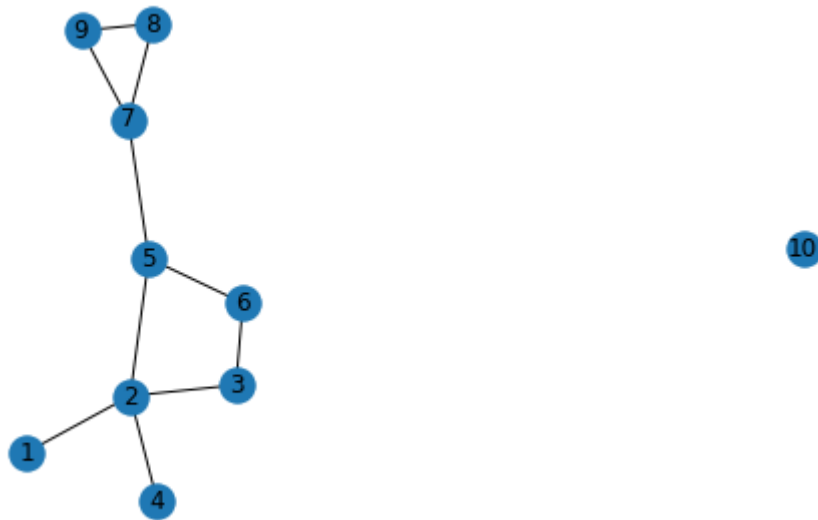
*Imagine that your social network has a subnetwork where 14 of your friends together with you are all friends with each other. What is such a subnetwork called? How many edges are contained in the subnetwork?*

For  $N = 15$  nodes we just calculate

$$\sum_{n=1}^{N-1} n = \frac{N(N-1)}{2} = \frac{15 \cdot 14}{2} = 105$$

### Problem 3

```
In [ ]: G = nx.Graph()
G.add_nodes_from(range(1,11))
G.add_edges_from([(1,2), (2,3), (2,4), (2,5), (3,6), (5,6), (5,7), (7,8), (8,9), (9,10)])
nx.draw(G, with_labels=True)
```



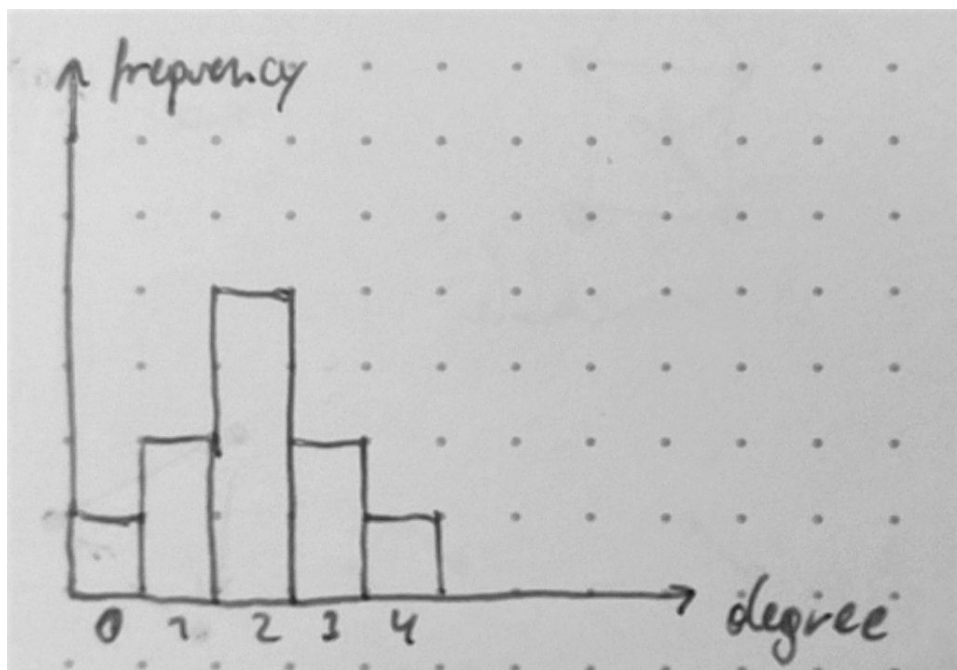
a) Write down the adjacency matrix.

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

b) Write down the edge list.

(1,2), (2,3), (2,4), (2,5), (3,6), (5,6), (5,7), (7,8), (7,9), (8,9)

c) Draw the degree distribution by hand.



d) Calculate the clustering coefficient, diameter and density.

The clustering coefficient of node  $n$  is  $C_n := \frac{e_n}{\max_n}$ , where  $e_n$  is the number of edges between the neighbors of node  $n$  and  $\max_n = \frac{\deg(n)(\deg(n)-1)}{2}$  is the maximum number of edges between these neighbors. The only nonzero clustering coefficient of our graph are  $C_7 = \frac{1}{3}$ ,  $C_8 = 1$  and  $C_9 = 1$ . Thus the average clustering coefficient is  $\langle C \rangle = \frac{7}{30}$  if we suppose that the clustering coefficient of node 10 is zero.

The diameter is the longest of the shortest paths between each node. Since node 10 is isolated the distance to another node is infinite. If we disregard node 10 the distance is clearly  $d = 4$  which is for example the distance  $d_{1,8}$ .

The density is  $D = \frac{2L}{N(N-1)} = \frac{2}{9}$  for the number of nodes  $N = 10$  and number of edges  $L = 10$ .

e) Find the number of  $d=3$  paths between 2 and 3.

There are 4 paths from 2 to its neighbors and back then to 3, also the paths 2->3->6->3 and 2->5->6->3 have length 3. This makes 6 such paths.

f) Which node pair has the most  $d=3$  paths?

Guess: (2, 5) since they are neighbors and they have the highest degrees.

## Problem 4

Consider a bipartite network with  $N_1$  and  $N_2$  nodes in the two sets.

a) What is the maximum number of edges the network can have?

At most there can be  $N_2$  edges from each node in the  $N_1$ -group to the other which makes  $N_1 \cdot N_2$  edges. Since the graph is undirected we already counted all edges.

b) How many edges cannot occur compared with a non-bipartite network of size  $N = N_1 + N_2$ ?

All the edges between  $N_1$ -nodes and between  $N_2$ -nodes are missing, which comes to  $\frac{N_1(N_1-1)}{2} + \frac{N_2(N_2-1)}{2}$ . Alternatively it can be calculated by  $\frac{(N_1+N_2)(N_1+N_2-1)}{2} - N_1N_2$ .