Pre-class work

Create a Google document or a Gist and record your work and all exercises. **Make sure your document is shared** so that it can be assessed, and **be ready to paste a link to your document into a class poll**.

Percolation

Following the mathematical analysis in the textbook, n is defined as the number of nodes in a network, and q is defined as the probability that a random node from the network is not part of the largest connected component (LCC) of the network.

The derivation in the textbook shows that q is a solution to the equation

$$q = e^{\langle k \rangle (q-1)} \tag{1}$$

where $\langle k \rangle$ is the average degree of the network.

Question: Given the information above, what is the theoretical estimate for the number of nodes in the LCC, expressed in terms of the known variables, n, q, and $\langle k \rangle$?

Task: Plot how the size of the LCC depends on the average degree $\langle k \rangle$ by using the theoretical result in (1). This equation does not have a nice analytical solution, so we use a numerical root finder in Scipy to determine the value of q that solves the equation for a given $\langle k \rangle$. A root finder computes a numerical solution to an equation of the form $f(x) = 0$, so we need to rewrite (1) as

$$q - e^{\langle k \rangle (q-1)} = 0 \tag{2}$$

We give the expression on the left-hand side to the root finding function. Use the code below to compute q for different values of $\langle k \rangle$ in the range $[1, 10]$. Note that $\langle k \rangle$ will not necessarily be an integer since it is an average degree.

Use the value of q to determine the theoretical estimate for the size of the LCC in a network with average degree $\langle k \rangle$. Plot your results of how the size of the LCC depends on $\langle k \rangle$.

```
def calculate_q(k):
    '''
    Use a numerical root finder to determine q from the equation
    q = exp(k*(q-1)).
    '''
    from scipy.optimize import root
    return root(lambda q: q - np.exp(k * (q - 1)), 0).x[0]
```