Computational neurodynamics

Exercise Sheet 3 (Unassessed)
Small-world networks

All the files for these exercises can be found online at

https://www.github.com/pmediano/ComputationalNeurodynamics

To install the Brain Connectivity Toolbox go into your virtual environment and run

```
pip install bctpy
```

Question 1.

a) Start up Python and build a `NetworkWattsStrogatz` with 12 nodes, a neighbourhood size of 4 and a rewiring probability of 0.1. You can visualise the network you have built with `PlotConnectivity`. Inspect the code inside these functions and make sure you understand how it works.

b) Using these two functions convince yourself that the Watts-Strogatz procedure produces a ring lattice for rewiring probability p=0 and a random network for p=1.

b) Create and plot networks for a series of intermediate values of p. Using the function `SmallWorldIndex`, compute the small-world indices of each network.

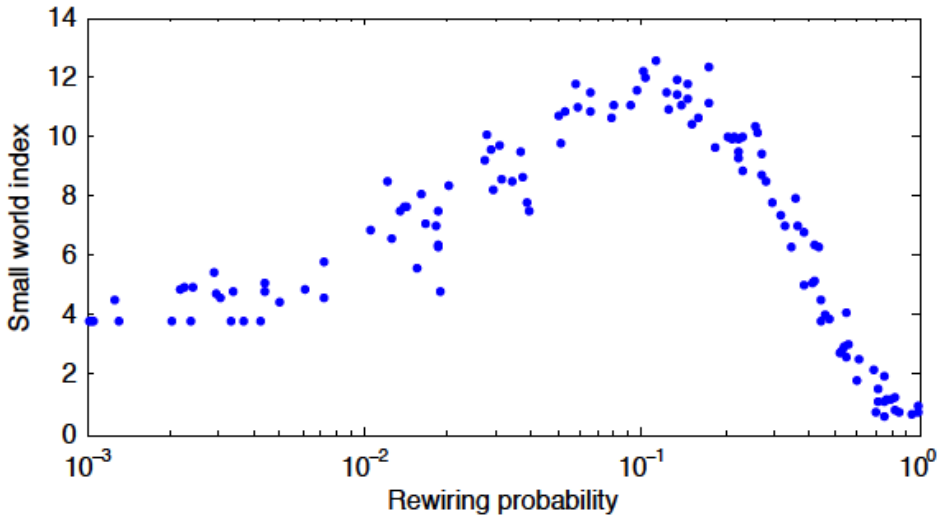
Question 2.

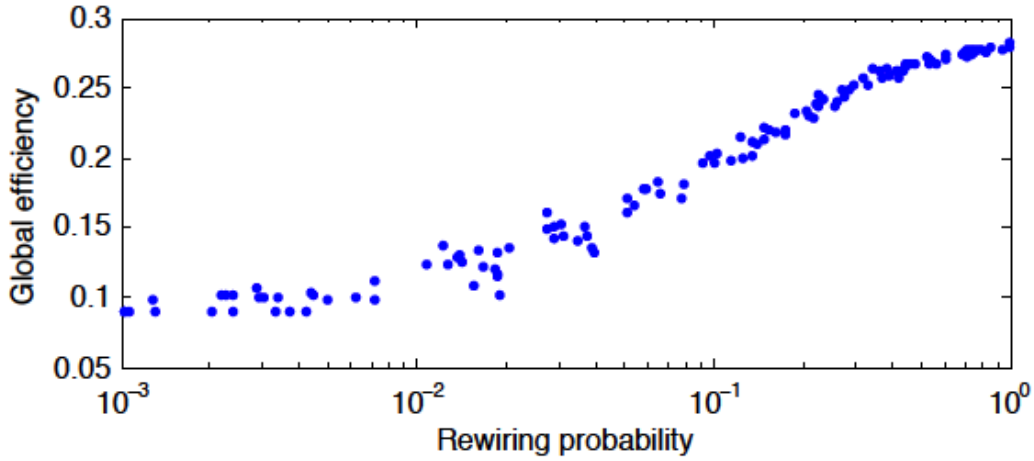
Using the functions above, along with the lecture notes, write a Python script to calculate and plot the local and global efficiency of Watts-Strogatz networks for different values of p. You should replicate the graphs from the lectures pasted below.

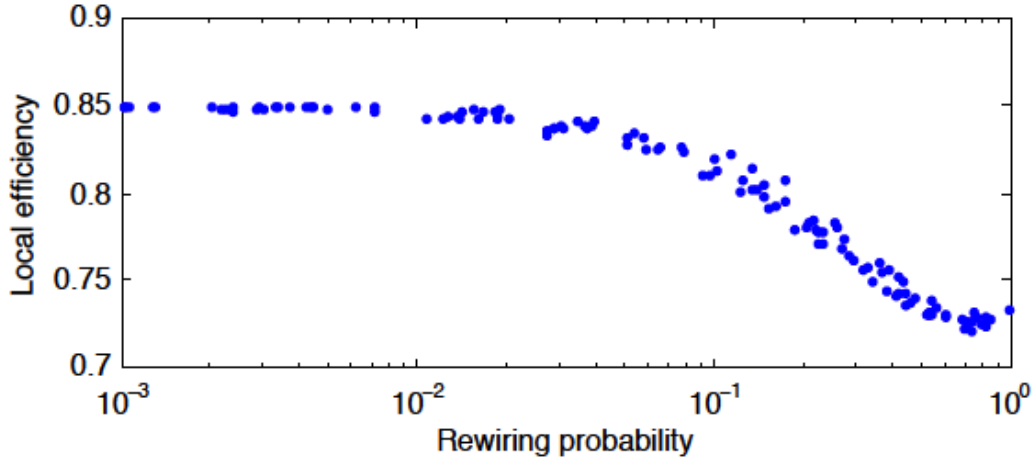
Note: More information about the Brain Connectivity Toolbox in

https://sites.google.com/site/bctnet/

As of 31.X.2015, there is a discrepancy in the local efficiency implementation between the original Matlab BCT and BCTPY. Don't be alarmed if there are some minor differences between BCTPY and the plots in the lecture slides, that were obtained in Matlab. A bug report has been submitted and this should be fixed soon.


Small world index
14
12
10
8
6
4
2
0
10^{-3}
10^{-2}
10^{-1}
10^{0}
Rewiring probability


Global efficiency
0.3
0.25
0.2
0.15
0.1
0.05
10^{-3}
10^{-2}
10^{-1}
10^{0}
Rewiring probability


Local efficiency
0.9
0.85
0.8
0.75
0.7
10^{-3}
10^{-2}
10^{-1}
10^{0}
Rewiring probability