Complex Networks, CSYS/MATH 303
University of Vermont, Spring 2014
Assignment 1 • code name: "Dun Dun Duuuunnnn!"

Dispersed: Tuesday, January 14, 2014.
Due: By start of lecture, 2:30 pm, Thursday, January 30, 2014.
Some useful reminders:
Instructor: Peter Dodds
Office: Farrell Hall, second floor, Trinity Campus
E-mail: peter.dodds@uvm.edu
Office hours: 3:45 pm to 4:15 pm, Tuesday, and 12:45 pm to 2:15 pm, Wednesday
Course website: http://www.uvm.edu/~pdodds/teaching/courses/2014-01UVM-303

All parts are worth 3 points unless marked otherwise. Please show all your working clearly and list the names of others with whom you collaborated.

Graduate students are requested to use LaTeX (or related TeX variant).

- For this initial assignment, you are to explore a number of data sets, performing some key measurements.
- Data is available in two compressed formats:
  - Matlab + text (tgz): `http://www.uvm.edu/~pdodds/teaching/courses/2014-01UVM-303/data/303complexnetworks-data-package.tgz`
  - Matlab + text (zip): `http://www.uvm.edu/~pdodds/teaching/courses/2014-01UVM-303/data/303complexnetworks-data-package.zip`

  and can also be found on the course website (helpfully) under data.
- The main matlab file containing everything is networkdata_combined.mat.
- For directed networks, the ijth entry of the adjacency matrix represents the weight of the link from node i to node j. Adjacency matrices for undirected networks are symmetric.
- For all questions below, treat each network unless otherwise instructed.
- For this assignment, convert all weights on links to 1, if the network is weighted.
- Data will be in Matlab struct format and you are strongly encouraged to use Matlab for your basic analyses. The supplied text versions may be of use for visualization using gml.

- The Matlab command spy will give you a quick plot of a sparse adjacency matrix.
- Real data sets used here are taken from Mark Newman's compilation (and linked-to sites) at `http://www-personal.umich.edu/~mejn/netdata/`.

1. Record in a table the following basic characteristics:
   - N, the number of nodes;
   - m, the total number of links;
   - Whether the network is undirected or directed based on the symmetry of the adjacency matrix;
   - $\langle k \rangle$, the average degree ($\langle k_{\rm in} \rangle$ and $\langle k_{\rm out} \rangle$ if the network is directed);
   - The maximum degree $k^{\max}$ (for both out-degree and in-degree if the network is directed);
   - The minimum degree $k^{\min}$ (for both out-degree and in-degree if the network is directed).
2. (3+3)
   (a) Plot the degree distribution P_k as a function of k. In the case that P_k versus k is uninformative, also produce plots that are clarifying. For example, $\log_{10} P_k$ versus $\log_{10} k$.
   (Note: Always use base 10.)
   (b) See if you can characterize the distributions you find (e.g., exponential, power law, etc.).
3. Measure the clustering coefficient C_2 where
$$C_2 = \frac{3 \times \#\text{triangles}}{\#\text{triples}}.$$
   For directed networks, transform them into undirected ones first.
   (Hint: to compute triangles, consider the trace of A^3.)
4. Measure the degree-degree assortativity. This is the standard Pearson correlation coefficient and the focus is on links, and then the nodes at the end of each link.
   For undirected networks, we need to think about how we choose the ordering of an edge's two degrees when we perform the correlation. Which degree goes first? Or should we include both orderings? How about randomly choosing the ordering? Does it matter?

For directed networks, various correlations are possible (in-in, in-out, etc.). For this question, measure the correlation of the in-degree of the source node and the out-degree of the destination node for each link.

5. Produce plots of the adjacency matrices using Matlab's spy command.

6. Using a network visualization tool of your choice, produce plots of the networks (if possible, depending on size).

   For the smaller ones, please label the nodes numerically.