CocoNuTs
Complex Networks
@networksvox
Everything is connected

Complex Networks, CSYS/MATH 303
University of Vermont, Spring 2018
Assignment 1 • code name: Basically made of glass

Dispersed: Thursday, January 18, 2018.
Due: Friday, February 2, by 11:59 pm, 2018.
Some useful reminders:
Deliverator: Peter Dodds
Office: Farrell Hall, second floor, Trinity Campus
E-mail: peter.dodds@uvm.edu
Office hours: 10:05 am to 12:00 pm, Tuesday and Thursday
Course website: http://www.uvm.edu/pdodds/teaching/courses/2018-01UVM-303

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

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

Email submission: PDF only! Please name your file as follows (where the number is to be padded by a 0 if less than 10 and names are all lowercase):
`CSYS303assignment%02d$firstname-$lastname.pdf` as in
`CSYS303assignment06michael-palin.pdf`

Please submit your project's current draft in pdf format via email. Please use this file name format (all lowercase after CSYS):
`CSYS303project-$firstname-$lastname-YYYY-MM-DD.pdf` as in
`CSYS303project-lisa-simpson-1989-12-17.pdf`

- We will explore real networks throughout the course performing some key measurements introduced in Principles of Complex Systems.
- Please note that while Matlab files are available, you are encouraged to use Python (along with, for example, NetworkX or graph-tools).
- Data is available in two compressed formats:
  - Matlab + text (tgz): `http://www.uvm.edu/pdodds/teaching/courses/2018-01UVM-303/data/303complexnetworks-data-package.tgz`
  - Matlab + text (zip): `http://www.uvm.edu/pdodds/teaching/courses/2018-01UVM-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 as undirected unless otherwise instructed.
- For this assignment, convert all weights on links to 1, if the network is weighted.
- You do not have to use Matlab for your basic analyses. Python would be a preferred route for many.
- 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 ($\langle k_{\text{in}} \rangle$ and $\langle k_{\text{out}} \rangle$ 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.

One approach is to compute C_2 as

$$C_2 = \frac{3 \times \frac{1}{6}\mathrm{Tr}A^3}{\frac{1}{2}\left(\sum_{ij}[A^2]_{ij} - \mathrm{Tr}A^2\right)}.$$

Note: avoiding computing A^3 is important and can be done.