Statistical Analysis of Networks
Statistics 218
Professor: Mark S. Handcock

Homework 1

Due date on the Bruinlearn *Assignments* page

You will need the `networkdata` and `sna` packages. To install the packages from inside `R`:

```
install.packages("network")
install.packages("sna")
install.packages("networkdata",repos="http://www.stat.ucla.edu/~handcock")
```

1) *Centrality:* Here we consider the same three network data sets from Homework 1 in the `networkdata` package

```
library(networkdata)
data(butland_ppi)
data(addhealth9)
data(tribes)
```

Useful other packages are:

```
library(sna)
library(network)
```

You can check the nature of the networks, via, `help(tribes)`, etc. These networks need to be processed a bit for the analysis to match those in Lecture 6. For the Add Health 9 network, consider only the (known) boys (via `addhealth9$X[,"female"]==0`). Next consider the undirected versions of the networks (For the Add Health 9 network we choose to say there is a tie if either boy nominates the other as a friend. For the tribes take the positive relations, as in Lecture 6). Next consider only the largest component (see, e.g., `component.dist` in `sna`).

a) For each network, compute the eigenvalue centrality of each node (see `evcent` in `sna`). For each network, summarize the centralities using node-level graphical and numerical summaries (see `centrality` in `sna`). *Hint:* The `mode` option in `evcent` and `centralization` is important.

b) For each network, compute the closeness centrality of each node. For each network, summarize the centralities using node-level graphical and numerical summaries (see `closeness` in `sna`).

c) For each network, compute the betweenness centrality of each node. For each network, summarize the centralities using node-level graphical and numerical summaries (see `betweenness` in `sna`).

d) For each network, compare the four centrality measures (degree,eigenvector, closeness and betweenness). Are they measuring the same thing?

e) Reconstruct the network-level centralizations on the page "Empirical study: Comparing centralization of different networks" of Lecture 6.

	degree	closeness	betweenness	eigenvector
ppi	0.13	0.26	0.31	0.33
addhealth	0.07	0.17	0.22	0.43
tribes	0.35	0.50	0.51	0.36

f) For each network, find a measure of the vertex-connectivity and edge-connectivity of the giant component. Find a minimal set of vertices and a minimal set of edges that you could remove to disconnect the giant component.

2) *Connectivity:* Here we consider the Florentine marriage data from the `network` package, as we did in the class exercise.

```
library(networkdata)
data(florentine)
```

a) Find measures of the vertex-connectivity and edge-connectivity of the large component. Find a minimal set of vertices and a minimal set of edges that you could remove to disconnect the giant component.

3) *Degree distributions:* Degree distributions summarize the densities of ties of the population of nodes. In this question we explore the interactions between proteins of the yeast *S. cerevisiae*. The nodes are types of proteins in the yeast and a directed tie is said to exist if a protein binds to the target protein in a i"wet lab" experiment set up to test just this. Not all protein combinations are tested. Here we will consider a series of "mapping" experiments conducted in 2008 that covered approximately 20% of all yeast binary interactions (Yu et. al Science (2008))

a) Go to the home page of the "Yeast Interactome Project":
`http://interactome.dfci.harvard.edu/S_cerevisiae/`
From there download the interactions from CCSB-YI11. These comprise 1809 interactions among 1278 proteins. Construct a `network` object from this edge-list.

b) Fit degree distribution models using the `degreenet` package. Fit the classical discrete Pareto/Zipf law model discussed in the books:

$$P(K = k; \nu) = \frac{k^{-\nu}}{\zeta(\nu)} \qquad \nu \geq 1, k = 1, 2, \ldots$$

Fit the Yule, Waring, Poisson, and Conway-Maxwell-Poisson models. Note: The corresponding functions are `adpmle`, `ayulemle`, `awarmle`, `apoimle`, `acmpmle`. See the help pages.

c) How can we compare the fits of the different models? Use the `lldpall()`, etc, functions to compute the corrected AIC and the BIC for the models. Summarize the fits of the models in a table. Which models fit best? Briefly comment on the models as a whole in terms of their fit.

4) *Components:* Continuing with the CCSB-YI11 network: Consider undirected the version of it where two proteins are linked if either of them binds with the other. This network has 1809 edges.

a) Compute the component distribution of the network. How large is the largest component? Does the network have a "giant" component? Note: Consider the `component.dist` function in the `sna` package.

b) Plot the subgraph comprised of the largest component.

c) Compute the (pairwise) matrix of geodesic distances between the proteins. Create a summary tabulation of the distances. What proportion of nodes-pairs are reachable (from each other)? What is the mean geodesic distance for reachable pairs? How many isolates are there in the network?