ICPSR: Network Analysis I
Assignment 4

by James D. Wilson and Melanie Baybay (University of San Francisco)

Answer the following in the assignment4.R file and submit via github (Completed Assignments link). These will be due on Wednesday, July 12th by the time of class.

1. Read Section 3.5 of Statistical Analysis of Networks in R and work through the code in R Studio. Pay close attention to their approach of identifying subgraphs / groups of nodes.

2. Go to `https://snap.stanford.edu/data/index.html` and identify a network that interests you from their collection of datasets. Load the data set into R studio and convert it to an 'igraph' object. Then answer the following questions:

   (a) What do the vertices and edges represent in this example?

   (b) How many vertices and edges are there in the data set?

   (c) Is the network weighted? If so, what do the weights represent?

   (d) Is the network directed or undirected?

   (e) How much storage will it require to store the network using sparse representation? How about as an adjacency matrix?

   (f) What kinds of questions would you like to investigate for this application?

3. Extract 3 induced subgraphs, each with 200 random nodes, from the dataset that you've looked at in (2). Visualize each of these networks using circular, Kamada-Kawai and Fructerman Reingold layouts (total of 9 plots). Consider what attributes in your chosen network could be used to recolor or resize the vertices. Are there any interesting network properties that become apparent when coloring or resizing the vertices?

4. Suppose that we represent the nodes of G arbitrarily by the sequence of numbers $[n] = \{1, \ldots, n\}$. Does the re-ordering of these nodes in a graph G affect the *global* structural properties of its associated adjacency matrix? (By *global*, I mean sub-graph counts, or totals on the graph) Why or why not?

5. One problem that we will look into in much more depth later is the problem of *community detection*. In a static network, community detection identifies a partition of the nodes of a graph into disjoint collections of vertices that are far more connected within their own community than they are to vertices in other communities. Imagine that you ran community detection on a graph G with 75 nodes and identified 3 equally sized communities that were well separated. Subsequently, you re-ordered the rows and columns of the adjacency matrix $\mathbf{A}$ so that the communities are presented in order. If you were to look at a heat map of $\mathbf{A}$, what do you expect the matrix to look like once it has been re-ordered according to its communities? (Note: there is a specific structure that should be apparent whenever communities are well separated.)

6. Often used to demonstrate statistical classification problems, the Iris flower data set is a classic multivariate data set from R.A. Fischer's 1936 paper, *The Use of Multiple Measurements in Taxonomic Problems*. It includes three iris flower species, with 50 samples of each and measurements for sepal length, sepal width, petal length, and petal width. Consider how this data set can be modeled as a network and what the network represents. You can load this data to your `R` session using the command: `data(iris)`. First, generate two graph models for this data where the nodes represent the 150 flower

observations (one of which is an association graph of your choice and the other of which is a Gaussian graphical model). For each graph, perform the following tasks:

(a) Generate and plot an adjacency matrix for the iris data set.

(b) Generate its network and color the vertices based on species name.

(c) Visualize the network using a layout (i.e. circular, kamada-kawai, fructerman reingold) that best represents its inherent community structure.

(d) Based on these plots and basic summary statistics of the two networks you've generated, comment on the similarities and differences of your two means of creating a graph.