HW2

Network Analytics, MSc BA, 2016_17

Individual part due on **25 Nov, 7PM** (worth 3% of your final grade).

Group part due on **6 Dec 2015 10AM** (worth 8% of your final grade).

There are three parts, an individual part (to get you started quickly) and a group part. Both have to be submitted. In addition there are some practice problems that you don't submit, but I recommend solving.

Programming: You are allowed to consult on the programming part with your colleagues and on the web but the final code has to be written and debugged entirely on your own.

In the following, you have to search for the appropriate functions in NetworkX yourselves.

FOR READING: Barbasi 3.1 to 3.5, 3.8, 4.1 to 4.3, 4.12 EK: 3.1, 4.1, 14.3, 20.1

INDIVIDUAL PART (due 25 Nov 7 PM)

Submission Instructions: *Submit individual and group parts separately via the HUB. The file names for the individual part should be as follows:* ***LastName_FirstName_HW#****.xxx (either .word or .pdf or .xls along with the code).*

1. Load the data HW2_tsp.txt and calculate the distance matrix (use an approximation like here which is quite accurate for short distances; or use packages like haversine or geopy). Submit the distance matrix and your code.

GROUP PART (due 6 Dec 10AM)

Submission Instructions: *The file names should be as follows:* ***Group#_HW#****.xxx (either .word or .pdf or .xls along with the code).*

2. Calculate 4 centrality measures, (with at least one using betweenness, one using flow and one, eigenvalue concepts; use only functions built into NetworkX) on the data from HW2_ problem2.txt (same as the data from HW1 problem 2). The data is of an organization where the nodes represent people, some of whom have a leadership role. Identify the two most important nodes according to each criteria. Write a few lines on what the measure is trying to capture and whether it is appropriate in the context.

3. The data HW2_tsp.txt contains latitude and longitude data of 38 cities in a country in Africa (Djibouti).

   a. Plot the latitude and longitude as a scatter plot using a drawing package (some options are: matplotlib basemap toolkit (the most advanced, but also the most difficult to use), geopy, gmplot, plotly …). It should look roughly like this.

   DO EITHER (b) OR (c) --- no need to do both

   b. Use the or-tools traveling salesman routine to find a tour of the 38 cities. If it is not finding a tour fast enough, try a subset of 10 cities (then 15 and so on).

   c. [Challenge problem, worth 10% bonus points--- so your grade for HW2 can be (8.8/8)] The most powerful integer programming solver is Gurobi (along with CPLEX). They give a free one year license if you download and install from a University IP address. Use the most powerful computer you have in your group (cores and memory).

      Connect with the Python interface of Gurobi and find an optimal tour using the sub-tour elimination integer programming formulation we did in class. You cannot generate all the sub-tour elimination constraints at once (too many). Instead,

      i. Start with a problem with only the degree =2 constraints.
      ii. Check the resulting solution for sub-tours. If there is a sub-tour, add the sub-tour elimination constraints corresponding to a cut defined by the sub-tour (i.e. you are eliminating that sub-tour in the next round).
      iii. Repeat till you find a tour (in which case, it is optimal). Make sure you do not discard the solution from the previous round, but start from what you had found.

      This method of generating constraints on the fly is suitable when the number of constraints are exponential in the problem size. The dual version of this is called column generation. Compare the optimal solution with the or-tools solution.

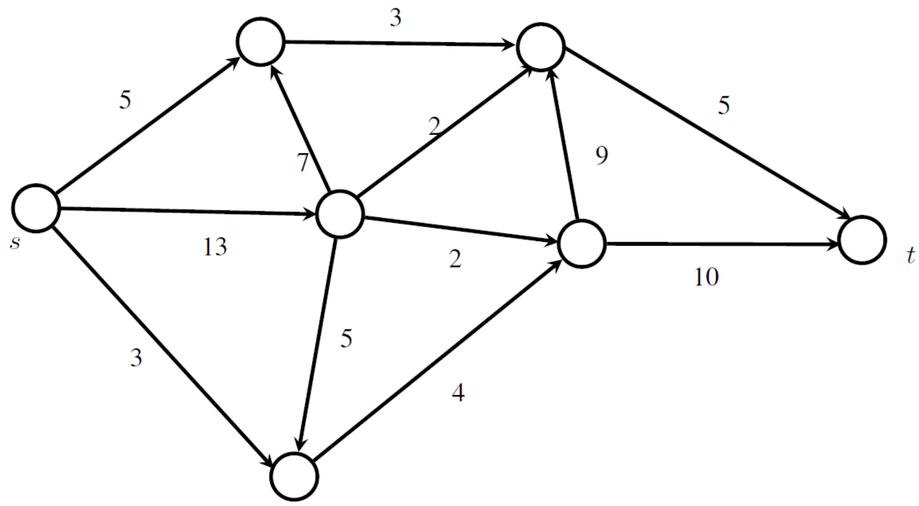
The optimal tour is here.

d. Plot the resulting tour on the scatter plot.

PRACTICE PROBLEMS (Not for submission, but I strongly recommend you spend time on them)

1. Show that the greedy algorithm works in finding the optimal (min or max) weighted spanning tree. i.e., you have to show (a) it terminates in polynomial time (b) the tree that it finds is an optimal tree.

2. a. Formulate the problem of sending the maximum amount of flow on the following network (the edges have capacities as shown) as a linear program



b. Formulate the dual of the linear program as well.
c. Solve using a linear programming solver (say Excel)

3. (somewhat challenging) With reference to problem 2

Define a s-t *Cut* as a partition of V into disjoint sets S, T (i.e., $S \cup T = V$ and $S \cap T = \emptyset$) with s in S and t in T.
Define *capacity of the cut*, Cap(S, T) = sum of the capacities of edges from S to T
Define *Flow across the cut*, Flow(S,T): net flow out of S = sum of flows out of S - sum of flows into S

a. From the dual values (i.e. the shadow prices) of the linear program corresponding to the optimal extreme point (basic feasible) solution find a minimum-capacity cut. (the cut itself would be easy to find by inspection, but you have to come up with a general way of extracting the cut from the optimal dual variables).
b. Is Max-flow always less than Minimum capacity of a cut between s and t?
c. Are they equal by linear programming theory?