CSE416A

Homework 1

M. Neumann

Due: TUE 25 SEPT 2018 10AM

This homework consists of 3 problems.

Submission Instructions

- **written work**
  - needs to be submitted electronically in *pdf format* via GRADESCOPE
  - start every problem on a *new page*
  - we prefer *typed submissions*, e.g., using LaTeX (if we cannot read your handwriting we cannot give you credit)
- **code**
  - needs to be submitted in form ofa submission to the corresponding GRADESCOPE programming assignment (intructions can be found on the course webpage)
  - make sure to not change the *file name(s)* and follow the *formatting instructions* (otherwise we cannot grade your submission)
- **group work** (up to 2 students)
  - make a **goup submission** via GRADESCOPE (one submission per team) for both written work and code submission

Piazza

If you haven't done so already, sign up to Piazza (`https://piazza.com/wustl/fall2018/cse416a`). All course and homework related announcements will be made there. Ask **all questions** on Piazza using the appropriate tags.

Grading Results and Regrades

Grades will be uploaded to Canvas and detailed grading comments will be provided via GRADESCOPE . You will be notified viaGRADESCOPE when the grades are published. All regrade requests need to made via GRADESCOPE **within one week** of this announcement.

PROBLEM 1: Complete graphs and shortest paths (20%)

[Pen-and-paper] A *complete graph* $G = (V, E)$ with $|V| = n$ nodes and $|E| = m$ edges is defined as the graph with the maximum number of edges. Assume that G is *undirected* and has no *self loops*.

(a) What is the number of edges m for a complete graph (as a function of n)?

(b) What is the average node degree (as a function of n)?

(c) How could you use a complete graph to model social interactions between people? Think about a suitable type of graph and interpret the meaning of its data.

(d) Let P denote a shortest path between the nodes u and v (that are at a distance greater than 2 from each other) in a network $\tilde{G}$, and w be another node on this path P. $\tilde{G}$ is not necessarily a complete graph. If Q denotes the segment of P between u and w, including both u and w (see Figure 1 for an illustration), is it true that Q is a shortest path between the nodes u and w? If yes, give a short proof. If no, explain with an example.

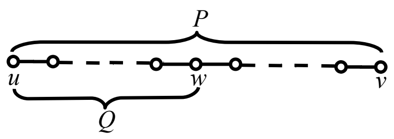

Figure 1: Problem 1.

PROBLEM 2: Average node distance (30%)

When we think about a single aggregate measure to summarize the distances between the nodes in a given graph, there are two natural quantities that come to mind. One is the *diameter*, which we define to be the maximum distance between any pair of nodes in the graph. Another is the *average distance*, which – as the term suggests – is the average distance over all pairs of nodes in the graph.

(a) **[Pen-and-paper]** Perform BREATH-FIRST SEACH (BFS) to compute all distances from node CASE to any other node in the Arpanet (cf. [NCM] Figure 2.3). Show your work to receive maximum credit. Verify that you get the same result by applying DIJKSTRA'S ALGORITHM discussed in-class. No need to include this in your submission. What is the differnece between BFS and DIJKSTRA in terms of graph types?

(b) **[Implementation]** Represent the Arpanet using an edge list and run BFS or DIJKSTRA to verify your computations programatically. You may use NetworkX functions. Add the edge list and the output of your program to your written submission.

(c) **[Plot]** Now, compute all pairwise distances between all nodes in the Arpanet and plot the *distance distribution* (`matplotlib.pyplot.hist` will be useful). Your program should be a general purpose program – do not hardcode the distances. You will use this program again in a future assignment. Add the plot to your written submission. HINT: Be careful to not *double count* the edges!

PROBLEM 3: Who is the most central actor? (50%)

See `hw1_problem3.ipynb` for instructions.

(a) **Load Data and Compute LCC**

(b) **Degree centrality**

(c) **Betweenness centrality**

(d) **Closeness centrality**