Homework 2

EE232E - Graphs and Network Flows

Due April 25,2017

One can use the igraph library (http://igraph.sourceforge.net/) to generate all kinds of networks and measure various properties of a given network. The library has R, Python, Ruby and C interfaces. You can choose one of these languages to program and generate the following networks, although R is preferred.

Submission: Please submit a zip file containing your codes and report to "ee232e.spring2017@gmail.com". The zip file should be named as "HW1_UID1_UID2_..._UIDn.zip" where UIDx are student ID numbers of team members. If you had any questions you can send an email to the same address.

1. Random walk on random networks

   (a) Create undirected random networks with 1000 nodes, and the probability p for drawing an edge between any pair of nodes equal to 0.01.

   (b) Let a random walker start from a randomly selected node (no damping). We use t to denote the number of steps that the walker has taken. Measure the average distance $\langle s(t)\rangle$ of the walker from his starting point at step t. Also measure the standard deviation $\sigma^2(t) = \langle (s(t) - \langle s(t)\rangle)^2\rangle$ of this distance. Plot $\langle s(t)\rangle$ v.s. t and $\sigma^2(t)$ v.s. t. Here, the average $\langle\cdot\rangle$ is over all possible starting nodes and different runs of the random walk (or different walkers). You can measure the distance of two nodes by finding the shortest path between them.

(c) We know that a random walker in d dimensional has average (signed) distance $\langle s(t) \rangle = 0$ and $\sqrt{\langle s(t)^2 \rangle} = \sigma \propto \sqrt{t}$. Compare this with the result on a random network. Do they have similar relations? Qualitatively explain why.

(d) Repeat (b) for undirected random networks with 100 and 10000 nodes. Compare the results and explain qualitatively. Does the diameter of the network play a role?

(e) Measure the degree distribution of the nodes reached at the end of the random walk on the 1000-node random network. How does it compare with the degree distribution of graph?

2. Random walk on networks with fat-tailed degree distribution

   (a) Use barabasi.game to generate a network with 1000 nodes and degree distribution proportional to x^{-3}.

   (b) Let a random walker start from a randomly selected node. Measure and plot $\langle s(t) \rangle$ v.s. t and $\sigma^2(t)$ v.s. t.

   (c) Are these results similar to results of random walks in d dimensional space? Explain why.

   (d) Repeat (b) for fat-tailed networks with 100 and 10000 nodes. Compare the results and explain qualitatively. Does the diameter of the network play a role?

   (e) Measure the degree distribution of the nodes reached at the end of the random walk on the 1000-node fat-tailed network. How does it compare with the degree distribution of the graph?

3. PageRank

   The PageRank algorithm, as used by the Google search engine, exploits the linkage structure of the web to compute global "importance" scores that can be used to influence the ranking of search results. Here, we use the random walk to simulate PageRank.

   (a) For random walks on the network created in 1(a), measure the probability that the walker visits each node. Is this probability related to the degree of the nodes?

(b) Create a directed random network with 1000 nodes, where the probability p for drawing an edge between any pair of nodes is 0.01. Measure the probability that the walker visits each node. Is this probability related to the degree of the nodes?

(c) In all previous questions, we used a damping parameter equal to $d = 1$, which means no teleportation (because teleportation probability is equal to $1 - d = 0$). Now, we use a damping parameter $d = 0.85$. For random walks on the network created in 1(a), measure the probability that the walker visits each node. Is this probability related to the degree of the node?

4. Personalized PageRank

While the use of PageRank has proven very effective, the web's rapid growth in size and diversity drives an increasing demand for greater flexibility in ranking. Ideally, each user should be able to define their own notion of importance for each individual query.

(a) Create a directed random network with 1000 nodes, where the probability p for drawing an edge between any pair of nodes is 0.01. Use the random walk with damping parameter 0.85 to simulate the PageRank of the nodes.

(b) Suppose you have your own notion of importance. Your interest to a node is proportional to the the node's PageRank, because you totally rely upon Google to decide which website to visit (assume that these nodes represent websites). Again use the random walk to simulate this personalized PageRank. Here the teleportation probability to each node is proportional to its PageRank (As opposed to the regular PageRank, where teleportation probability to all nodes are the same and equal to $\frac{1}{N}$). The damping parameter is equal to $d = 0.85$. Compare the results with (a).

(c) More or less, (b) is what happens in the real world. However, this is against the original assumption of normal PageRank, where we assume that people's interest in all nodes are the same. Can you take into account the effect of this self-enforcement and adjust the PageRank equation?

Useful commands:

`(1) sample_gnp, netrw, get.shortest.paths, diameter; (2) barabasi.game, netrw, get.shortest.paths, diameter; (3,4) sample_gnp, netrw, degree`

If you are unsure, please just use the default values for the command parameters.

You can use the `netrw` package to implement random walks and personalized PageRank. The `netrw` package source code for different platforms are uploaded in the handouts section on course website. You can search online on how to install R packages from source code for your specific operating system. If you are using RStudio you can just install the package from archive in the package manager. But this package needs R version before `R 3.0.0`. You may want to use `random_walk` instead and implement teleportation yourself in 3(c) and 4.